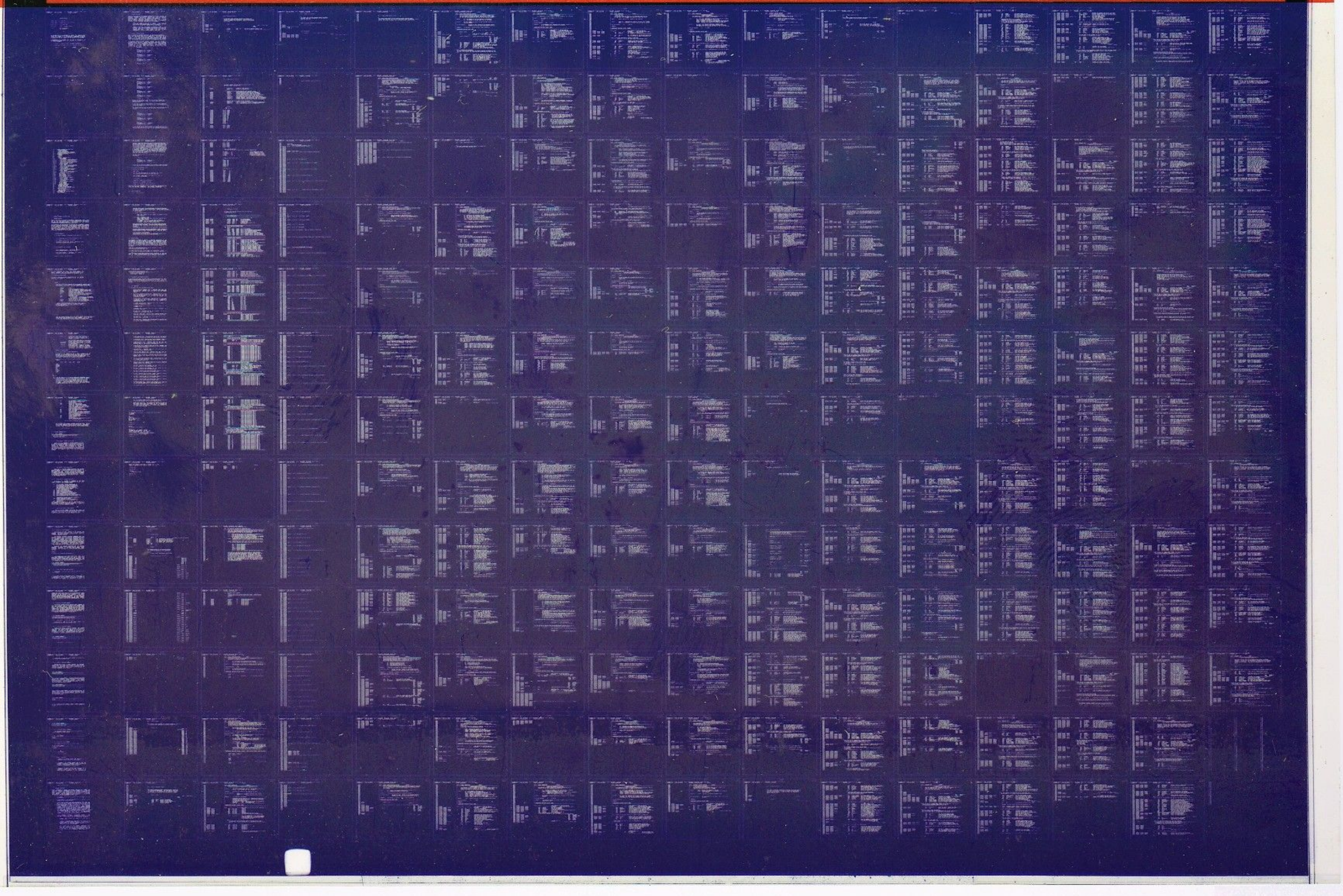
DHV11

DHV-11 FUNC TST PT2
CVDHBA0

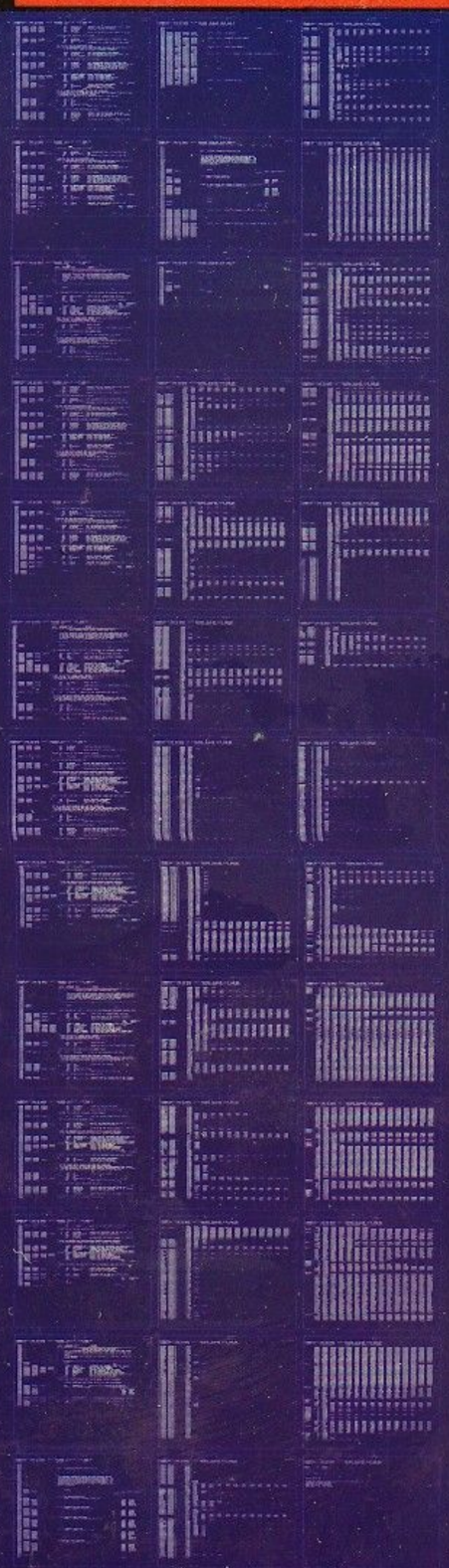
AH-T655A-MC
FICHE 1 OF 2



DHV11

DHV-11 FUNC TST PT2
CVDHBA0

AH-T655A-MC
FICHE 2 OF 2



```
.REM &
```

IDENTIFICATION

PRODUCT CODE: AC-T654A-MC

PRODUCT NAME: CVDHBA0 DHV-11 FUNC TST PART2

PRODUCT DATE: 31 OCTOBER 1983

MAINTAINER: EDSHE - DIAGNOSTICS GROUP

AUTHOR: BERT KLEINSCHMIDT
TONY GRIMSHAW

***** MODIFICATION HISTORY *****

ORIGINAL RELEASE: 31-OCT-83 BERT KLEINSCHMIDT

TABLE OF CONTENTS

1.0 GENERAL PROGRAM CONSIDERATIONS

1.1 PROGRAM ABSTRACT

CVDHB IS PART ONE OF THE DHV-11 FUNCTIONAL VERIFICATION TEST. THIS PART OF THE TEST VERIFIES THAT THE MAJOR COMMUNICATION FUNCTIONS OF THE BOARD ARE FUNCTIONING CORRECTLY. THIS PROGRAM DOES NOT PERFORM EXTENSIVE DATA TRANSMISSION AND RECEPTION TESTS.

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

1.2 SYSTEM REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE DHV FVT:

- O LSI-11 PROCESSOR WITH AT LEAST 32 KBYTES OF RAM.
- O DHV11 BOARDS INSTALLED ON THE Q-BUS.
- O APPROPRIATE PROGRAM LOAD DEVICE SUPPORTING XXDP+ MEDIA OR A DOWN-LINE LOADING SYSTEM.

1.3 RELATED DOCUMENTS AND STANDARDS

- O DHV-11 HARDWARE MANUAL - THIS MANUAL DESCRIBES THE FUNCTIONS AND USES OF THE DHV-11 DEVICE.
- O XXDP+ USER'S MANUAL - DESCRIBES THE RUNNING OF DIAGNOSTICS UNDER THE XXDP+ MONITOR.

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE LSI-11 PROCESSOR, THE Q-BUS, THE SYSTEM MEMORY, THE CONSOLE TERMINAL, AND THE LOAD MEDIA ARE ASSUMED TO HAVE BEEN TESTED AND FOUND WORKING BEFORE THIS PROGRAM IS RUN.

2.0 OPERATING INSTRUCTIONS

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

2.1 COMMANDS

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

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

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".
MORE INFORMATION CAN BE FOUND WITHIN THE SECTION LABELLED EXTENDED COMMAND SYNTAX

2.2 SWITCHES

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

SWITCH	EFFECT

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

EXAMPLE OF SWITCH USAGE:

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

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

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

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

2.3 FLAGS

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

FLAG	EFFECT

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

*SEE THE ERROR INFORMATION SECTION OF THIS DOCUMENT.

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

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

2.4 EXTENDED COMMAND SYNTAX

2.4.1 START COMMAND -

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

2.4.1.1 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.), SEPERATED BY COLONS, THAT SPECIFY THE TESTS TO BE EXECUTED. 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 "EFFECT OF START COMMAND" SECTION.

2.4.1.2 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).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 "EFFECT OF START COMMAND" SECTION.

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

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

HOE	HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED.
LOE	LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR.
IER	INHIBIT ERROR REPORTING.
IBE	INHIBIT BASIC ERROR REPORTS.
IXE	INHIBIT EXTENDED ERROR REPORTS.
PRI	DIRECT ALL MESSAGES TO A LINE PRINTER.
PNT	PRINT NUMBER OF TEST BEING EXECUTED.
BOE	BELL ON ERROR (NOT RELATED TO BELL PROMPTING).
UAM	RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION (ILLEGAL FOR THIS DIAGNOSTIC).
ISR	INHIBIT STATISTICAL REPORTS.
IDU	INHIBIT DROPPING OF UNITS BY DIAGNOSTIC. (HAS NO EFFECT IN THIS DIAGNOSTIC.)
LOT	LOOP ON TEST.

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

2.4.1.4 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 "EFFECT OF START COMMAND" SECTION.

2.4.1.5 EFFECT OF START COMMAND -

THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, THE INITIALIZATION QUESTIONS, AND THEN THE DIAGNOSTIC COMMENCES TESTING.

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION "# UNITS (D) ?" TO WHICH THE OPERATOR SHOULD REPLY WITH THE NUMBER OF UNITS TO BE TESTED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES ARE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE COMPLETE UNIT.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. FOR THE ACTUAL HARDWARE P-TABLE QUESTIONS SEE THE "HARDWARE PARAMETERS" SECTION.

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE OPERATING PARAMETERS OF THE DIAGNOSTIC PROGRAM. THESE QUESTIONS ARE DESCRIBED IN THE "SOFTWARE PARAMETERS" SECTION.

EXAMPLE:

```
STA/TESTS:1:3-4:/PASS:3/FLAGS:IER:HOE=1
```

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, WITH EACH PASS CONSISTING OF TESTS 1,3, AND 4. 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.

2.4.2 RESTART COMMAND -

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

2.4.2.1 TESTS, PASS, AND FLAGS SWITCHES -

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

2.4.2.2 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.

2.4.2.3 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 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, OR C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

2.4.3 CONTINUE COMMAND -

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

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

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

2.4.3.2 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.

2.4.4 PROCEED COMMAND -

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

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

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

2.4.4.2 EFFECT OF PROCEED COMMAND -

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

2.4.5 ADD COMMAND -

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

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

2.4.7 DROP COMMAND -

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

2.4.8 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.

2.4.9 PRINT COMMAND -

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

2.4.9.1 EFFECT OF PRINT COMMAND - ERROR SUMMARY REPORTING IS NOT IMPLEMENTED IN THIS DIAGNOSTIC, SO THIS COMMAND HAS NO EFFECT.

2.4.10 DISPLAY COMMAND -

****

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

****

2.4.10.1 EFFECT OF DISPLAY COMMAND -

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

2.4.11 FLAGS COMMAND -

****

FLA(GS)

****

2.4.11.1 EFFECT OF FLAGS COMMAND -

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

2.4.12 ZFLAGS COMMAND -

****

ZFL(AGS)

****

2.4.13 ZFLAGS COMMAND -

ALL FLAGS ARE CLEARED.

2.4.14 CONTROL CHARACTERS -

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

Z A CONTROL/Z (Z) ENTERED DURING ONE OF THE TWO OPERATOR DIALOGUES-- HARDWARE P-TABLE DIALOGUE OR SOFTWARE P-TABLE DIALOGUE CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

O 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 CONTROL/O IS TYPED, WHICH RESTORES

NORMAL TELETYPE OUTPUT.

2.5 HARDWARE QUESTIONS

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

1. CSR ADDRESS - THIS QUESTION REQUESTS THE CSR ADDRESS OF THE SPECIFIED DHV11.

2. VECTOR ADDRESS - THIS QUESTION REQUESTS THE INTERRUPT VECTOR ADDRESS OF THE SPECIFIED DHV11.

3. ACTIVE LINES BIT MAP - THIS QUESTION REQUESTS AN OCTAL BIT MAP OF THE SERIAL COMMUNICATION LINES ON THE DHV11 WHICH ARE BEING SELECTED FOR TESTING. IF THE BIT IN THE BIT MAP IS SET WHICH CORRESPONDS TO A PARTICULAR LINE (I.E. BIT 3 FOR LINE 3) THAT LINE WILL BE TESTED BY THE FVT. WITH STAGGERED LOOPBACK A PAIR OF LINES WITH THE SPECIFIED TRANSMIT LINE AND ANOTHER RECEIVE LINE WILL BE TESTED. THEREFORE, TO GUARANTEE THAT BOTH THE TRANSMITTER AND RECEIVER OF A SPECIFIED LINE ARE TESTED WHEN USING THE STAGGERED LOOPBACK CONNECTOR, BOTH THE INTENDED LINE AND ITS MATE MUST BE SELECTED (IE. TO TEST LINE 1, SELECT BOTH LINE 1 AND LINE 3). IN NONSTAGGERED TESTING, A BIT IN THE ACTIVE LINES BIT MAP SELECTS THE TRANSMITTER AND RECEIVER FOR THE SAME LINE.

4. TYPE OF LOOPBACK (1=INTERNAL, 2=STAGGERED, 3=H325) - THIS QUESTION REQUESTS THE TYPE OF LOOPBACK TO BE USED IN TESTING THE DHV11. THE FOLLOWING TYPES OF LOOPBACK ARE SUPPORTED:

   - INTERNAL - ONLY INTERNAL UART LOOPBACK IS TO BE USED IN TESTING THE DHV.

   - STAGGERED - STAGGERED BERG CONNECTOR(S) ARE INSTALLED ON THE BERG CONNECTOR SOCKETS OF THE DHV11. FOR THE CIRCUIT CONNECTIONS OF THE STAGGERED LOOPBACK CONNECTOR SEE THE HARDWARE SECTION OF THIS DOCUMENT.

   - H325 - SINGLE LINE, 25 PIN LOOPBACK CONNECTORS (TYPE H325) ARE INSTALLED ON THE LINES TO BE TESTED. THESE CONNECTORS CAN BE INSTALLED ON THE DISTRIBUTION PANEL OR ON THE END OF THE TERMINAL OR MODEM CABLE. THE H325 CONNECTORS MUST HAVE THE REMOVABLE JUMPERS INSTALLED.

5. BR LEVEL - THIS QUESTIONS REQUESTS THE INTERRUPT BR LEVEL OF THE DHV11.

2.6 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y". THE FOLLOWING SOFTWARE P-TABLE QUESTIONS ARE ASKED BY THE PROGRAM IF THE OPERATOR INDICATES THAT THE SOFTWARE PARAMETERS ARE TO BE CHANGED:

1. REPORT UNIT NUMBER AS EACH UNIT IS TESTED - THIS QUESTION ASKS WHETHER THE PROGRAM SHOULD REPORT THE NUMBER OF THE UNIT WHICH IT IS TESTING AS IT BEGINS TO TEST EACH UNIT.

2. NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE - THIS QUESTION ASKS FOR THE NUMBER OF DATA ERRORS WHICH SHOULD BE REPORTED INDIVIDUALLY BY THIS PROGRAM FOR EACH LINE FOR EACH TRANSMISSION TEST. ERRORS WHICH ARE NOT REPORTED INDIVIDUALLY ARE REPORTED IN SUMMARY ERROR REPORTS.

2.7 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0<CR>
Q-FACTOR (O) 0 ?  1<CR>

UNIT 2
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  1<CR>
Q-FACTOR (O) 1 ?  0<CR>

UNIT 3
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  2<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 4
CSR ADDRESS (O) ?  160000<CR>
```

```
                    SUB-DEVICE # (0) ?  3<CR>
                    Q-FACTOR (0) 0 ?  <CR>

                    UNIT 5
                    CSR ADDRESS (0) ?  160000<CR>
                    SUB-DEVICE # (0) ?  4<CR>
                    Q-FACTOR (0) 0 ?  <CR>

                    UNIT 6
                    CSR ADDRESS (0) ?  160000<CR>
                    SUB-DEVICE # (0) ?  5<CR>
                    Q-FACTOR (0) 0 ?  <CR>

                    UNIT 7
                    CSR ADDRESS (0) ?  160000<CR>
                    SUB-DEVICE # (0) ?  6<CR>
                    Q-FACTOR (0) 0 ?  1<CR>

                    UNIT 8
                    CSR ADDRESS (0)  160000<CR>
                    SUB-DEVICE # (0) ?  7<CR>

                    Q-FACTOR (0) 1 ?  <CR>
```

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

```
        # UNITS (D) ?  8<CR>

        UNIT 1
        CSR ADDRESS (0) ?  160000<CR>
        SUB-DEVICE # (0) ?  0,1<CR>
        Q-FACTOR (0) 0 ?  1,0<CR>

        UNIT 3
        CSR ADDRESS (0) ?  160000<CR>
        SUB-DEVICE # (0) ?  2-5<CR>
        Q-FACTOR (0) 0 ?  0<CR>

        UNIT 7
        CSR ADDRESS (0) ?  160000<CR>
        SUB-DEVICE # (0) ?  6,7<CR>
        Q-FACTOR (0) 0 ?  1<CR>
```

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES

ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES 2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 160000 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0-7<CR>
Q-FACTOR (O) 0 ?  0,1,0,,,,1,1<CR>
```

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

2.8 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK AND THE QUESTION IS ASKED) QUESTIONS
3. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
6. ANSWER ALL THE HARDWARE QUESTIONS
7. ANSWER THE "CHANGE SW" QUESTION WITH "N"

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. FOR DEFAULT INFORMATION SEE THE SECTIONS WITHIN THIS DOCUMENT ON FLAGS,AND HARDWARE QUESTIONS.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

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

THE GENERAL ERROR MESSAGE IS OF THE FORM:

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

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

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

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

3.2 ERROR MESSAGES

THIS PROGRAM IS INTENDED TO PROVIDE A GO/NO-GO INDICATION OF THE FUNCTIONALITY OF DHV-11 BOARDS. TO EXECUTE THE PROGRAM IN THIS MODE THE OPERATOR CAN RUN WITH THE INHIBIT BASIC ERROR REPORTING SWITCH. IN THIS MODE THE PROGRAM PRINTS ERROR MESSAGES WHICH CONTAIN THE ERROR MESSAGE HEADER DESCRIBED ABOVE, PLUS THE NAME OF THE FAILING TEST. FOR A LIST OF THE TEST NAMES IN THIS PROGRAM SEE THE TEST SUMMARIES SECTION OF THIS DOCUMENT. AN EXAMPLE OF SUCH AN ERROR MESSAGE IS THE FOLLOWING:

```
CVDHB DVC FTL ERR 01603 ON UNIT 02 TST 015 SUB 000 PC: 015244
DEVICE REGISTER WORD READ/WRITE TEST
```

THIS ERROR INDICATES THAT A FATAL ERROR WAS ENCOUNTERED WITHIN THE TEST WHICH TESTS THE READ/WRITE CAPABILITY OF THE DHV-11 REGISTERS.

IF THE OPERATOR REQUIRES MORE EXTENSIVE ERROR REPORTING HE CAN RUN WITH ALL ERROR REPORTING ENABLED BY NOT USING THE INHIBIT REPORTING SWITCHES. THE ABOVE ERROR MESSAGE WOULD THEN BECOME THE FOLLOWING:

```
CVDHB DVC FTL ERR 01603 ON UNIT 02 TST 015 SUB 000 PC: 015244
```

```
DEVICE REGISTER WORD READ/WRITE TEST
  BAD BIT(S) IN DEVICE TBUFFAD1 REGISTER FOR LINE  7 (D).
    EXPECTED DATA: 000000 (O).
    ACTUAL DATA:   000023 (O).
```

4.0 PERFORMANCE AND PROGRESS REPORTS

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

5.0 TEST SUMMARIES

THE FOLLOWING TESTS ARE INCLUDED WITHIN CVDHB:

1. DEVICE REGISTER ADDRESS TEST - VERIFIES THAT THE UUT REGISTERS WILL RESPOND WITHT THE PROPER Q-BUS HANDSHAKING WHEN ACCESSED. VERIFIES THAT THE UUT IS AT THE PROPER ADDRESS.

2. NO TX.DATA.VALID/NO TX.ACTION TEST - VERIFIES THAT IF A DATA WORD IS WRITTEN WITHOUT THE TX.DATA.VALID BIT SET, NO TX.ACTION IS GENERATED. THIS TEST DOES NOT REQUIRE THAT CHARACTERS ARE TXED.

3. TX.DATA.VALID / TX.ACTION TEST - VERIFIES THAT IF A DATA WORD IS WRITTEN WITH THE TX.DATA.VALID BIT SET, IT GENERATES A CORRESPONDING TX.ACTION. THIS TEST DOES NOT REQUIRE THAT CHARACTERS ARE TXED.

4. TX.ENABLE INACTIVE TEST - VERIFIES THAT IF THE TX.ENABLE BIT IS CLEAR NO TRANSMISSION OCCURS.

5. TX.ENABLE ACTIVE TEST - VERIFIES THAT TX OCCURS IF THE TX.ENABLE IS SET.

6. INTERRUPTS TEST - VERIFIES THAT THE TX AND RX INTERRUPTS ARE FUNCTIONING CORRECTLY.

7. BR LEVEL TEST - VERIFIES THAT THE UUT GENERATES TX AND RX INTERRUPTS AT THE CORRECT BR LEVEL.

8. DIAG FIELD (BMP) TEST - VERIFIES THAT A REQUEST FOR BMP CODE REPORTING IS ANSWERED BY THE UUT WITHIN THE SPECIFIED TIME.

9. DMA.START TEST - VERIFIES THAT EACH DMA.START BIT WILL INITIATE A DMA TX ON A LINE.

10. DMA.ABORT TEST - VERIFIES THAT THE DMA.ABORT BIT ON EACH LINE WILL STOP A DMA TRANSMISSION AND RETURN A TX.ACTION AND THAT THE DMA CAN THEN BE RESTARTED.

11. O.AUTO INACTIVE TEST - VERIFIES THAT THE UUT WILL NOT RESPOND TO INCOMING XON AND XOFF CHARACTERS WHEN O.AUTO IS INACTIVE.

12. O.AUTO ACTIVE TEST - VERIFIES THAT THE UUT RESPOND CORRECTLY TO INCOMING XON AND XOFF CHARACTERS WHEN O.AUTO IS ACTIVE.

13. I.AUTO INACTIVE TEST - VERIFIES THAT THE UUT WILL NOT GENERATE AND TX XON OR XOFF CHARACTERS IN RESPONSE TO THE FIFO CONDITIONS IF THE I.AUTO BIT IS INACTIVE.

14. I.AUTO ACTIVE TEST - VERIFIES THAT THE UUT WILL GENERATE AND TX XON AND XOFF CHARACTERS IN RESPONSE TO THE FIFO CONDITIONS IF THE I.AUTO BIT IS ACTIVE.

15. FIFO DATA TEST - VERIFIES THAT THE FIFO WILL HOLD 256 CHARACTERS WITHOUT CORRUPTING DATA.

16. FIFO 3/4 LEVEL INACTIVE TEST - VERIFIES THAT THE FIFO 3/4 ALARM DOES NOT BECOME ACTIVE UNTIL THE FIFO BECOMES 3/4 FULL.

17. FIFO 3/4 LEVEL ACTIVE TEST - VERIFIES THAT THE FIFO 3/4 ALARM BECOMES ACTIVE, AND REMAINS ACTIVE, WHEN THE FIFO IS MORE THAN 3/4 FULL.

18. FIFO 3/4 LEVEL ACTIVE/INACTIVE TEST - VERIFIES THAT THE FIFO 3/4 ALARM, ONCE ACTIVATED, REMAINS ACTIVE UNTIL THE FIFO LEVEL IS REDUCED BELOW 1/2.

19. FIFO 1/2 LEVEL TEST - VERIFIES THAT FIFO 1/2 LEVEL INDICATOR BECOMES ACTIVE AND REMAINS ACTIVE AS THE FIFO LEVEL IS REDUCED BELOW THE 1/2 FULL POINT.

20. DTR TEST - VERIFIES THAT CHANGING THE UUT LNCTRL DTR BIT AFFECTS THE STATE OF THE DTR CONTROL LINE.

21. RTS TEST - VERIFIES THAT CHANGING THE UUT LNCTRL RTS BIT AFFECTS THE STATE OF THE RTS CONTROL LINE.

22. DSR TEST - VERIFIES THAT DSR STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK DTR CONTROL LINE.

23. RI TEST - VERIFIES THAT RI STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK DTR CONTROL LINE.

24. CTS TEST - VERIFIES THAT CTS STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK RTS CONTROL LINE.

25. DCD TEST - VERIFIES THAT DCD STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK RTS CONTROL LINE.

26. DTR INTERACTIONS TEST - VERIFIES THAT CHANGING THE STATE OF THE DTR CONTROL SIGNAL ON ANY LINE DOES NOT AFFECT THE STATE OF ANY STATUS SIGNALS THAT IT IS NOT LOOPED BACK TO.

27. RTS INTERACTIONS TEST - VERIFIES THAT CHANGING THE STATE OF

THE RTS CONTROL SIGNAL ON ANY LINE DOES NOT AFFECT THE STATE OF ANY STATUS SIGNALS THAT IT IS NOT LOOPED BACK TO.

28. REPORT BMP CODES TEST - THIS PSEUDO TEST REPORTS THE FIRST 32 BMP CODES WHICH WERE DISCOVERED IN THE FIFO DURING THE EXECUTION OF THE OTHER TESTS. THIS AVOIDS THE INTERRUPTION OF OTHER TESTS BY THESE CODES, IF THEY ARE NOT CRITICAL TO THE TESTS BEING PERFORMED.

6.0 EXAMPLE ERROR FREE PASS

THE FOLLOWING IS AN EXAMPLE OF AN ERROR FREE PASS DIALOGUE:

```
.R CVDHBA0
CVDHBA0.BIC

DRS
CVDHB-A-0
DHV-11 FUNC TST PART2
UNIT IS DHV-11
RESTART ADDR: 147670
DR>STA

CHANGE HW (L)  ? Y

# UNITS (D)  ? 2

UNIT 0
CSR ADDRESS:  (O)  160020 ? ^Z

UNIT 1
CSR ADDRESS:  (O)  160020 ? 160040
INTERRUPT VECTOR ADDRESS:  (O) 300 ? 320
ACTIVE LINE BIT MAP:  (O)  377 ? <CR>
TYPE OF LOOPBACK (1=INTERNAL OR NONE,2=STAGGERD,
                  3=25 PIN CONNECTOR):  (O)  2 ? 1
INTERRUPT BR LEVEL:  (O)  4? <CR>

CHANGE SE (L)  ? Y
```

```
        REPORT UNIT NUMBER AS EACH UNIT IS TESTED:  (L) Y ? <CR>
        NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE:  (D)  0 ? 4

         TESTING UNIT :  0

         TESTING UNIT :  1

        CVDHB EOP    1
            0 CUMULATIVE ERRORS

         TESTING UNIT :  0
        ^C
        DR> EXIT

        .
```

&

```
                        .LIST SEQ,LOC,BIN,MEB

                        .NLIST CND



                        .SBTTL  PROGRAM HEADER


                                .MCALL  SVC
000000'                         SVC                             ; INITIALIZE SUPERVISOR MACROS
        000001          SVCINS= 1       ; LIST INSTRUCTIONS, SHIFTED RIGHT
        000001          SVCTST= 1       ; LIST TEST TAGS, SHIFTED RIGHT
        000001          SVCSUB= 1       ; LIST SUBTEST TAGS, SHIFTED RIGHT
        000001          SVCGBL= 1       ; LIST GLOBAL TAGS, SHIFTED RIGHT
        000001          SVCTAG= 1       ; LIST OTHER TAGS, SHIFTED RIGHT
                        .ENABL  ABS
                        .ENABL  AMA
        002000          .       =       2000

002000                          BGNMOD

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

002000                  POINTER BGNRPT,BGNSW,BGNSFT,BGNDU,ERRTBL


002000                  HEADER  CVDHB,A,0,70,0,PRI07
002000
002000  103
002001  126
002002  104
002003  110
002004  102
002005  000
002006  000
002007  000
002010
002010  101
002011
002011  060
002012
002012  000000
002014
002014  000070
002016
002016  037706
002020
002020  040244
002022
002022  002216
002024
002024  002230

                L$NAME::
                        .ASCII  /C/
                        .ASCII  /V/
                        .ASCII  /D/
                        .ASCII  /H/
                        .ASCII  /B/
                        .BYTE   0
                        .BYTE   0
                        .BYTE   0
                L$REV::
                        .ASCII  /A/
                L$DEPO::
                        .ASCII  /0/
                L$UNIT::
                        .WORD   0
                L$TIML::
                        .WORD   70
                L$HPCP::
                        .WORD   L$HARD
                L$SPCP::
                        .WORD   L$SOFT
                L$HPTP::
                        .WORD   L$HW
                L$SPTP::
                        .WORD   L$SW
```

```
1118  002026
1119  002026  040504
1120  002030
1121  002030  000000
1122  002032
1123  002032  000000
1124  002034
1125  002034  000000
1126  002036
1127  002036  000000
1128  002040
1129  002040  002124
1130  002042
1131  002042  000340
1132  002044
1133  002044  000000
1134  002046
1135  002046  000000
1136  002050
1137  002050     003
1138  002051     003
1139  002052
1140  002052  000000
1141  002054  000000
1142  002056
1143  002056  000000
1144  002060
1145  002060  004122
1146  002062
1147  002062  020114
1148  002064
1149  002064  000000
1150  002066
1151  002066  000000
1152  002070
1153  002070  000000
1154  002072
1155  002072  021006
1156  002074
1157  002074  000000
1158  002076
1159  002076  004132
1160  002100
1161  002100  104035
1162  002102
1163  002102  004052
1164  002104
1165  002104  020130
1166  002106
1167  002106  020770
1168  002110
1169  002110  020766
1170  002112
1171  002112  020122
1172  002114
1173  002114  000000
```

```
L$LADP::
        .WORD   L$LAST
L$STA::
        .WORD   0
L$CO::
        .WORD   0
L$DTYP::
        .WORD   0
L$APT::
        .WORD   0
L$DTP::
        .WORD   L$DISPATCH
L$PRIO::
        .WORD   PRI07
L$ENVI::
        .WORD   0
L$EXP1::
        .WORD   0
L$MREV::
        .BYTE   C$REVISION
        .BYTE   C$EDIT
L$EF::
        .WORD   0
        .WORD   0
L$SPC::
        .WORD   0
L$DEVP::
        .WORD   L$DVTYP
L$REPP::
        .WORD   L$RPT
L$EXP4::
        .WORD   0
L$EXP5::
        .WORD   0
L$AUT::
        .WORD   0
L$DUT::
        .WORD   L$DU
L$LUN::
        .WORD   0
L$DESP::
        .WORD   L$DESC
L$LOAD::
        EMT     E$LOAD
L$ETP::
        .WORD   L$ERRTBL
L$ICP::
        .WORD   L$INIT
L$CCP::
        .WORD   L$CLEAN
L$ACP::
        .WORD   L$AUTO
L$PRT::
        .WORD   L$PROT
L$TEST::
        .WORD   0
```

```
 1174  002116
 1175  002116  000000
 1176  002120
 1177  002120  000000
 1178
```

```
L$DLY::
        .WORD   0
L$HIME::
        .WORD   0
```

```
                                .SBTTL  DISPATCH TABLE

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

                                        DISPATCH 28
002122                                                          .WORD   28
002122  000034                                          L$DISPATCH::
002124                                                          .WORD   T1
002124  021124                                                 .WORD   T2
002126  021414                                                 .WORD   T3
002130  021632                                                 .WORD   T4
002132  022074                                                 .WORD   T5
002134  022402                                                 .WORD   T6
002136  022746                                                 .WORD   T7
002140  024000                                                 .WORD   T8
002142  024670                                                 .WORD   T9
002144  025154                                                 .WORD   T10
002146  025550                                                 .WORD   T11
002150  026176                                                 .WORD   T12
002152  026744                                                 .WORD   T13
002154  027512                                                 .WORD   T14
002156  030114                                                 .WORD   T15
002160  030536                                                 .WORD   T16
002162  031034                                                 .WORD   T17
002164  031344                                                 .WORD   T18
002166  032044                                                 .WORD   T19
002170  032542                                                 .WORD   T20
002172  033214                                                 .WORD   T21
002174  033730                                                 .WORD   T22
002176  034444                                                 .WORD   T23
002200  035074                                                 .WORD   T24
002202  035524                                                 .WORD   T25
002204  036154                                                 .WORD   T26
002206  036604                                                 .WORD   T27
002210  037214                                                 .WORD   T28
002212  037624
```

```
1218
1219
1220                                    .SBTTL  DEFAULT HARDWARE P-TABLE
1221
1222                                    ;++
1223                                    ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
1224                                    ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
1225                                    ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
1226                                    ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
1227                                    ;--
1228
1229  002214                                    BGNHW   DFPTBL
1230  002214  000004                                                                   .WORD   L10000-L$HW/2
1231  002216                                                                    L$HW::
1232  002216                                                                    DFPTBL::
1233
1234  002216  160020                           .WORD   160020  ;DEFAULT CSR ADDRESS
1235  002220  000300                           .WORD   300     ;DEFAULT VECTOR ADDRESS
1236  002222  177777                           .WORD   177777  ;DEFAULT ACTIVE LINES BIT MAP
1237  002224     002                           .BYTE   2       ;DEFAULT LOOPBACK MODE
1238  002225     004                           .BYTE   4       ;DEFAULT BR LEVEL
1239
1240  002226                                    ENDHW
1241  002226                                                                    L10000:
```

```
1242
1243
1244                            .SBTTL  SOFTWARE P-TABLE
1245
1246                            ;++
1247                            ; THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
1248                            ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE
1249                            ; SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
1250                            ; AT RUN TIME.
1251                            ;--
1252
1253  002226                            BGNSW   SFPTBL
1254  002226  000002                                                   .WORD   L10001-L$SW/2
1255  002230                                                    L$SW::
1256  002230                                                    SFPTBL::
1257
1258  002230  000020                    OPTION::        .WORD   20      ;BIT MAP OF PROGRAM CONTROL FLAGS
1259  002232  000000                    NDERPT::        .WORD   0       ;DEFAULT NUMBER OF INDIVIDUAL DATA ERRORS TO RPT
1260
1261  002234                            ENDSW
1262  002234                                                    L10001:
```

```
                                .SBTTL  GLOBAL EQUATES SECTION



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

000010                                  NUMLNS==10      ;NUMBER OF LINES ON DHV11 IS 8.
000377                                  MAPLNS==377     ;BIT MAP OF LINES ON DHV11.

                                ;****** DEVICE REGISTER OFFSETS FROM THE CSR'S ADDRESS ******
000000                                  CSR0==0         ;CSR REGISTER OFFSET FROM THE CSR ADDRESS
000002                                  RBUF0==2        ;RECEIVE REGISTER OFFSET FROM THE CSR ADDRESS
000002                                  TXCHR0==2       ;TRANSMIT REGISTER OFFSET FROM THE CSR ADDRESS
000004                                  LPR0==4         ;LINE PARAMETER REGISTER OFFSET FROM THE CSR ADDRESS
000006                                  STAT0==6        ;STATUS REGISTER OFFSET FROM THE CSR ADDRESS
000010                                  LNCTR0==10      ;LINE CONTROL REGISTER OFFSET FROM THE CSR ADDRESS
000012                                  TXAD10==12      ;TRANSMIT ADDRESS 1 REGISTER OFFSET FROM THE CSR ADDRESS
000014                                  TXAD20==14      ;TRANSMIT ADDRESS 2 REGISTER OFFSET FROM THE CSR ADDRESS
000016                                  TXBFC0==16      ;TRANSMIT COUNT REGISTER OFFSET FROM THE CSR ADDRESS

                                ;****** EQUATES USED WITH RESPECT TO THE RX BUFFER ******
000020                                  RXBETX==16.     ;LEVEL OF RX BUFFER AT WHICH TO RE-ENABLE TRANSMISSION.
000030                                  RXBDTX==24.     ;LEVEL OF RX BUFFER AT WHICH TO DISABLE TRANSMISSION.
000100                                  RXBFUL==64.     ;TOTAL CHARACTER CAPACITY OF THE RX BUFFER.


002234                                          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
```

```
1319            000040                  BIT5==  BIT05
1320            000020                  BIT4==  BIT04
1321            000010                  BIT3==  BIT03
1322            000004                  BIT2==  BIT02
1323            000002                  BIT1==  BIT01
1324            000001                  BIT0==  BIT00
1325                                    ;
1326                                    ; EVENT FLAG DEFINITIONS
1327                                    ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
1328                                    ;
1329            000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
1330            000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
1331            000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
1332            000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
1333            000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
1334                                    ;
1335                                    ;
1336                                    ; PRIORITY LEVEL DEFINITIONS
1337                                    ;
1338            000340                  PRI07== 340
1339            000300                  PRI06== 300
1340            000240                  PRI05== 240
1341            000200                  PRI04== 200
1342            000140                  PRI03== 140
1343            000100                  PRI02== 100
1344            000040                  PRI01== 40
1345            000000                  PRI00== 0
1346                                    ;
1347                                    ;OPERATOR FLAG BITS
1348                                    ;
1349          000004                    EVL==         4
1350          000010                    LOT==        10
1351          000020                    ADR==        20
1352          000040                    IDU==        40
1353          000100                    ISR==       100
1354          000200                    UAM==       200
1355          000400                    BOE==       400
1356          001000                    PNT==      1000
1357          002000                    PRI==      2000
1358          004000                    IXE==      4000
1359          010000                    IBE==     10000
1360          020000                    IER==     20000
1361          040000                    LOE==     40000
1362          100000                    HOE==    100000
1363
```

```
                                .SBTTL  GLOBAL DATA SECTION

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

                                ;*****************************************************************************
                                ;       UNIT VARIBLE AREA
                                ;*****************************************************************************

002234  000300                          RXVECA:: .WORD  300     ;RX VECTOR ADDRESS.
002236  000304                          TXVECA:: .WORD  304     ;TX VECTOR ADDRESS.
002240  000377                          ACTLNS:: .WORD  377     ;ACTIVE LINE BIT MAP.
002242     000                          LOPBCK:: .BYTE  0       ;LOOPBACK MODE
002243     004                          BRLEVL:: .BYTE  4       ;INTERRUPT BUS REQUEST LEVEL
002244  000000                          UNITN:: .WORD   0       ;UNIT NUMBER.

                                ;*****************************************************************************
                                ;       DEVICE REGISTER ADDRESS TABLE
                                ;*****************************************************************************
002246                                  DRADRT::
002246  160000                          CSRA::  .WORD   160000  ;DHV-11 CSR ADDRESS
002250  160002                  TXCHA:: RBUFA:: .WORD   160002  ;DHV-11 RECEIVE/TRANSMIT BUFFER ADDRESS
002252  160004                          LPRA::  .WORD   160004  ;DHV-11 LINE PARAMETER REGISTER ADDRESS
002254  160006                          STATA:: .WORD   160006  ;DHV-11 STATUS REGISTER ADDRESS
002256  160010                          LNCTRA:: .WORD  160010  ;DHV-11 LINE CONTROL REGISTER ADDRESS
002260  160012                          TXAD1A:: .WORD  160012  ;DHV-11 TRANSMIT BUFFER 1 REGISTER ADDRESS
002262  160014                          TXAD2A:: .WORD  160014  ;DHV-11 TRANSMIT BUFFER 2 REGISTER ADDRESS
002264  160016                          TXBFCA:: .WORD  160016  ;DHV-11 TRANSMIT BUFFER COUNT REGISTER ADDRESS

                                ;*****************************************************************************
                                ;       ASSORTED GLOBAL VARIABLES:
                                ;*****************************************************************************
002266  000000                          BUFPTR:: .WORD  0       ;STORAGE FOR RECEIVE CHARACTER BUFFER POINTER.
002270  000000                          CTRLCF:: .WORD  0       ;STORAGE FOR THE CONTROL-C FLAG.
002272  000001                          TSTNUM:: .WORD  1       ;STORAGE FOR THE TEST NUMBER.
002274  000000                          IESTAT:: .WORD  0       ;STORAGE FOR STATES OF THE DUT INT ENABLE BITS.
002276  031463                          LGRP1M:: .WORD  31463   ;BIT MAP OF LINES IN LINE GROUP I.
002300  146314                          LGRP2M:: .WORD  146314  ;BIT MAP OF LINES IN LINE GROUP II.
002302  000000                          PASCNT:: .WORD  0       ;STO'G FOR PASS COUNT USED IN ROM VERSION# TST.
002304  000000                          RXINTC:: .WORD  0       ;STORAGE FOR RECEIVER INTERRUPT FLAGS.
002306  000000                          RXINTF:: .WORD  0       ;STORAGE FOR RECEIVER INTERRUPT FLAGS.
002310  000000                          TXINTC:: .WORD  0       ;STORAGE FOR TRANSMIT INTERRUPT COUNT.
002312  000000                          TXINTF:: .WORD  0       ;STORAGE FOR TRANSMIT INTERRUPT FLAGS.
002314  000000                          TP4VEC:: .WORD  0       ;STORAGE FOR THE NORMAL 004 TRAP VECTOR.
002316  000000                          TP4FLG:: .WORD  0       ;FLAGS SET WHEN AN EXPECTED 004 TRAP OCCURS.
002320  000000                          WORD1::  .WORD  0       ;LOCATION FOR PASSING INDIRECT PARAMETERS.
                                ;*****************************************************************************
                                ;       LINE TIME CLOCK VARIABLES AND STORAGE.
                                ;*****************************************************************************
002322  177546                          CLKCSR:: .WORD  177546  ;CSR ADDRESS OF THE LTC.
002324  000300                          CLKBRL:: .WORD  PR106   ;INTERRUPT PRIORITY LEVEL OF THE LTC.
002326  000100                          CLKVEC:: .WORD  100     ;INTERRUPT VECTOR ADDRESS OF THE LTC.
```

```
1420  002330  000074                    CLKHRZ:: .WORD  60.     ;INTERRUPT FREQUENCY OF THE LTC.
1421  002332  000000                    TIMER1:: .WORD  0       ;HARDWARE CLOCK COUNTER #1.
1422  002334  000000                    TIMER2:: .WORD  0       ;HARDWARE CLOCK COUNTER #2.
1423  002336  000170                    TIMER3:: .WORD  120.    ;HARDWARE BREAK COUNTER LOCATION.
1424  002340  000170                    BCOUNT:: .WORD  120.    ;BREAK COUNT VALUE IN CLOCK TICKS.
1425  002342  000021                    MSTICK:: .WORD  17.     ;NUMBER OF MILLI-SECONDS PER LTC TICK.
1426  002344  000062                    MSLCNT:: .WORD  62      ;LOOP COUNT (USED BY MSLOOP) TO DELAY 1 MS.
1427
1428                                ;*****************************************************************************
1429                                ;       MEMMORY MANAGEMENT VARIABLES AND FLAGS.
1430                                ;*****************************************************************************
1431  002346  177572                    MMSR0:: .WORD   177572  ;ADDRESS OF MEM MGT STATUS REGISTER #0.
1432  002350  000000                    MMPRES:: .WORD  0       ;MEM MGT PRESENT FLAG (0 IF MM NOT PRESENT).
1433  002352  000000                    MMENAB:: .WORD  0       ;MEM MGT ENABLED FLAG (0 IF MM NOT ENABLED).
1434
1435  002354                        PARATB::                    ;BASE OF MEM MGT PAR ADDRESS TABLE.
1436  002354  172340                        PAR0A:: .WORD   172340  ;ADDRESS OF MEM MGT PAR #0.
1437  002356  172342                        PAR1A:: .WORD   172342  ;ADDRESS OF MEM MGT PAR #1.
1438  002360  172344                        PAR2A:: .WORD   172344  ;ADDRESS OF MEM MGT PAR #2.
1439  002362  172346                        PAR3A:: .WORD   172346  ;ADDRESS OF MEM MGT PAR #3.
1440  002364  172350                        PAR4A:: .WORD   172350  ;ADDRESS OF MEM MGT PAR #4.
1441  002366  172352                        PAR5A:: .WORD   172352  ;ADDRESS OF MEM MGT PAR #5.
1442  002370  172354                        PAR6A:: .WORD   172354  ;ADDRESS OF MEM MGT PAR #6.
1443  002372  172356                        PAR7A:: .WORD   172356  ;ADDRESS OF MEM MGT PAR #7.
1444  002374                        PARATE::                    ;END OF PAR ADDRESS TABLE.
1445                                ;*****************************************************************************
1446                                ;       TABLE OF WORDS WITH CORRESPONDING BIT SET FOR GENERATION OF BIT MAPS.
1447                                ;*****************************************************************************
1448  002374  000001                BITTBL:: .WORD  1               ;BIT 0 SET.
1449  002376  000002                        .WORD   2               ;BIT 1 SET.
1450  002400  000004                        .WORD   4               ;BIT 2 SET.
1451  002402  000010                        .WORD   10              ;BIT 3 SET.
1452  002404  000020                        .WORD   20              ;BIT 4 SET.
1453  002406  000040                        .WORD   40              ;BIT 5 SET.
1454  002410  000100                        .WORD   100             ;BIT 6 SET.
1455  002412  000200                        .WORD   200             ;BIT 7 SET.
1456  002414  000400                        .WORD   400             ;BIT 8 SET.
1457  002416  001000                        .WORD   1000            ;BIT 9 SET.
1458  002420  002000                        .WORD   2000            ;BIT 10 SET.
1459  002422  004000                        .WORD   4000            ;BIT 11 SET.
1460  002424  010000                        .WORD   10000           ;BIT 12 SET.
1461  002426  020000                        .WORD   20000           ;BIT 13 SET.
1462  002430  040000                        .WORD   40000           ;BIT 14 SET.
1463  002432  100000                        .WORD   100000          ;BIT 15 SET.
1464
1465                                ;*****************************************************************************
1466                                ;*      GPR SAVE AREAS ZERO AND ONE.
1467                                ;*****************************************************************************
1468  002434                        GPRS0B::                    ;BASE OF GPR SAVE AREA NUMBER ZERO.
1469  002434  000000                        .WORD   0               ;WORD 1, STORAGE FOR R1.
1470  002436  000000                        .WORD   0               ;WORD 2, STORAGE FOR R2.
1471  002440  000000                        .WORD   0               ;WORD 3, STORAGE FOR R3.
1472  002442  000000                        .WORD   0               ;WORD 4, STORAGE FOR R4.
1473  002444  000000                        .WORD   0               ;WORD 5, STORAGE FOR R5.
1474                                ;*****************************************************************************
1475                                ;*      TRANSMISSION AND RECEPTION VARIABLES, POINTERS, AND FLAGS.
```

```
1476
1477  002446  000000                ERSMRF:: .WORD  0                ;'PRINT ERROR SUMMARY'' FLAGS.
1478                                ;*****************************************************************************
1479                                ;       STORAGE AREA FOR THE BMP CODE QUEUE.
1480                                ;*****************************************************************************
1481  002450  000000                BMPCQP::        .WORD   0        ;POINTER USED TO ACCESS THE NEXT CELL IN QUE.
1482  002452  000100                BMPCQB::        .BLKW   64.      ;STORAGE FOR 32 CELLS, TEST# PLUS BMP CODE.
1483  002652                        BMPCQE::                         ;LAST ADDRESS PLUS 2 OF THE BMP CODE QUEUE.
1484                                ;*****************************************************************************
1485                                ;       STORAGE AREA FOR THE CONTENTS OF THE DUT STAT REGISTER STATES.
1486                                ;*****************************************************************************
1487  002652                        STSTB::                          ;BASE OF DUT STAT STORAGE TABLE.
1488  002652  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 0.
1489  002654  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 1.
1490  002656  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 2.
1491  002660  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 3.
1492  002662  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 4.
1493  002664  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 5.
1494  002666  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 6.
1495  002670  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 7.
1496  002672  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 8.
1497  002674  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 9.
1498  002676  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 10.
1499  002700  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 11.
1500  002702  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 12.
1501  002704  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 13.
1502  002706  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 14.
1503  002710  000000                        .WORD   0                ;STORAGE FOR STAT REGISTER FOR LINE 15.
1504  002712                        STSTE::                          ;END OF DUT STAT STORAGE TABLE.
1505                                ;*****************************************************************************
1506                                ;       GENERAL TABLE AND BUFFER AREA--513 WORDS.
1507                                ;*****************************************************************************
1508  002712                        BUFBAS::                         ;BASE OF MEMORY BUFFER.
1509  002712  000200                ERLTBL::        .BLKW   128.     ;FIRST HALF OF GENERAL TABLE OR BUFFER.
1510  003312  000100                BUFMID::        .BLKW   64.      ;SECOND HALF OF GENERAL TABLE OR BUFFER.
1511  003512  000100                BUF3QT::        .BLKW   64.      ;LAST QUARTER OF THE BUFFER AREA.
1512  003712                        BUFEND::                         ;END OF GENERAL PURPOSE MEMORY BUFFER.
1513  003712  000020                ENDETB::        .BLKW   16.      ;BUFFER OVERFLOW SPACE.
1514                                ;*****************************************************************************
1515                                ;* TABLE FOR STORAGE OF RX/TX LINE NUMBER ASSOCIATIONS.
1516                                ;* THE ASSOCIATIONS ARE STORED AS LINE NUMBER TIMES 2 FOR USE AS OFFSETS
1517                                ;*    WHEN ACCESSING A TABLE OF WORDS.
1518                                ;*    NOTE: DO NOT WRITE A NON-ZERO VALUE INTO THE UPPER BYTE OF ANY ENTRY.
1519                                ;*****************************************************************************
1520  003752                        TXRXLB::                         ;BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
1521  003752  000000                        .WORD   0                ;TX/RX LINE OFFSET FOR RX/TX LINE 0.
1522  003754  000002                        .WORD   2.               ;TX/RX LINE OFFSET FOR RX/TX LINE 1.
1523  003756  000004                        .WORD   4.               ;TX/RX LINE OFFSET FOR RX/TX LINE 2.
1524  003760  000006                        .WORD   6.               ;TX/RX LINE OFFSET FOR RX/TX LINE 3.
1525  003762  000010                        .WORD   8.               ;TX/RX LINE OFFSET FOR RX/TX LINE 4.
1526  003764  000012                        .WORD   10.              ;TX/RX LINE OFFSET FOR RX/TX LINE 5.
1527  003766  000014                        .WORD   12.              ;TX/RX LINE OFFSET FOR RX/TX LINE 6.
1528  003770  000016                        .WORD   14.              ;TX/RX LINE OFFSET FOR RX/TX LINE 7.
1529  003772  000020                        .WORD   16.              ;TX/RX LINE OFFSET FOR RX/TX LINE 8.
1530  003774  000022                        .WORD   18.              ;TX/RX LINE OFFSET FOR RX/TX LINE 9.
1531  003776  000024                        .WORD   20.              ;TX/RX LINE OFFSET FOR RX/TX LINE 10.
```

```
1532  004000  000026                    .WORD   22.            ;TX/RX LINE OFFSET FOR RX/TX LINE 11.
1533  004002  000030                    .WORD   24.            ;TX/RX LINE OFFSET FOR RX/TX LINE 12.
1534  004004  000032                    .WORD   26.            ;TX/RX LINE OFFSET FOR RX/TX LINE 13.
1535  004006  000034                    .WORD   28.            ;TX/RX LINE OFFSET FOR RX/TX LINE 14.
1536  004010  000036                    .WORD   30.            ;TX/RX LINE OFFSET FOR RX/TX LINE 15.
1537  004012                    TXRXLE::                       ;END OF TX/RX LINE NUMBER ASSOCIATION TABLE.
1538                                    .EVEN                  ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
1539                            ;*******************************************************************************
1540                            ;* TABLE FOR STORAGE OF RX/TX LINE NUMBER ASSOCIATIONS.
1541                            ;* THE ASSOCIATIONS ARE STORED AS LINE NUMBERS WHICH CAN BE USED AS SUCH OR
1542                            ;*    AS OFFSETS WHEN ACCESSING A TABLE OF BYTES.
1543                            ;*******************************************************************************
1544  004012                    TXRLNB::                       ;BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
1545  004012  000                       .BYTE   0           ;TX/RX LINE FOR RX/TX LINE 0.
1546  004013  001                       .BYTE   1.          ;TX/RX LINE FOR RX/TX LINE 1.
1547  004014  002                       .BYTE   2           ;TX/RX LINE FOR RX/TX LINE 2.
1548  004015  003                       .BYTE   3.          ;TX/RX LINE FOR RX/TX LINE 3.
1549  004016  004                       .BYTE   4.          ;TX/RX LINE FOR RX/TX LINE 4.
1550  004017  005                       .BYTE   5.          ;TX/RX LINE FOR RX/TX LINE 5.
1551  004020  006                       .BYTE   6.          ;TX/RX LINE FOR RX/TX LINE 6.
1552  004021  007                       .BYTE   7.          ;TX/RX LINE FOR RX/TX LINE 7.
1553  004022  010                       .BYTE   8.          ;TX/RX LINE FOR RX/TX LINE 8.
1554  004023  011                       .BYTE   9.          ;TX/RX LINE FOR RX/TX LINE 9.
1555  004024  012                       .BYTE   10.         ;TX/RX LINE FOR RX/TX LINE 10.
1556  004025  013                       .BYTE   11.         ;TX/RX LINE FOR RX/TX LINE 11.
1557  004026  014                       .BYTE   12.         ;TX/RX LINE FOR RX/TX LINE 12.
1558  004027  015                       .BYTE   13.         ;TX/RX LINE FOR RX/TX LINE 13.
1559  004030  016                       .BYTE   14.         ;TX/RX LINE FOR RX/TX LINE 14.
1560  004031  017                       .BYTE   15.         ;TX/RX LINE FOR RX/TX LINE 15.
1561  004032                    TXRLNE::                       ;END OF TX/RX LINE NUMBER ASSOCIATION TABLE.
1562                                    .EVEN                  ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
1563                            ;*******************************************************************************
1564                            ;* TABLE OF TX/RX LINE NUMBER ASSOCIATIONS IN STAGGERED LOOPBACK.
1565                            ;* THE ASSOCIATIONS ARE STORED AS LINE NUMBER TIMES 2 FOR USE AS OFFSETS
1566                            ;*    WHEN ACCESSING A TABLE OF WORDS.
1567                            ;* THIS IS A TABLE OF DATA FOR READING ONLY.  USE TO LOAD THE ABOVE TABLE.
1568                            ;*    NOTE: MUST CONVERT FROM BYTES TO WORDS WHEN LOADING ABOVE TABLE.
1569                            ;*******************************************************************************
1570  004032                    STGTRB::                       ;BASE OF STAGGERED TX/RX LINE NUMBER TABLE.
1571  004032  004                       .BYTE   4.          ;TX/RX LINE OFFSET FOR RX/TX LINE 0.
1572  004033  006                       .BYTE   6.          ;TX/RX LINE OFFSET FOR RX/TX LINE 1.
1573  004034  000                       .BYTE   0           ;TX/RX LINE OFFSET FOR RX/TX LINE 2.
1574  004035  002                       .BYTE   2.          ;TX/RX LINE OFFSET FOR RX/TX LINE 3.
1575  004036  014                       .BYTE   12.         ;TX/RX LINE OFFSET FOR RX/TX LINE 4.
1576  004037  016                       .BYTE   14.         ;TX/RX LINE OFFSET FOR RX/TX LINE 5.
1577  004040  010                       .BYTE   8.          ;TX/RX LINE OFFSET FOR RX/TX LINE 6.
1578  004041  012                       .BYTE   10.         ;TX/RX LINE OFFSET FOR RX/TX LINE 7.
1579  004042  024                       .BYTE   20.         ;TX/RX LINE OFFSET FOR RX/TX LINE 8.
1580  004043  026                       .BYTE   22.         ;TX/RX LINE OFFSET FOR RX/TX LINE 9.
1581  004044  020                       .BYTE   16.         ;TX/RX LINE OFFSET FOR RX/TX LINE 10.
1582  004045  022                       .BYTE   18.         ;TX/RX LINE OFFSET FOR RX/TX LINE 11.
1583  004046  034                       .BYTE   28.         ;TX/RX LINE OFFSET FOR RX/TX LINE 12.
1584  004047  036                       .BYTE   30.         ;TX/RX LINE OFFSET FOR RX/TX LINE 13.
1585  004050  030                       .BYTE   24.         ;TX/RX LINE OFFSET FOR RX/TX LINE 14.
1586  004051  032                       .BYTE   26.         ;TX/RX LINE OFFSET FOR RX/TX LINE 15.
1587                                    .EVEN                  ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
```

```
1588  004052                                  ERRTBL
1589  004052                                                        L$ERRTBL::
1590  004052  000000                  ERRTYP::        .WORD   0
1591  004054  000000                  ERRNBR::        .WORD   0
1592  004056  000000                  ERRMSG::        .WORD   0
1593  004060  000000                  ERRBLK::        .WORD   0
1594
1595                                  .EVEN
```

```
                        .SBTTL  GPR HANDLING ROUTINES FOR SUBROUTINE CALLS.
                        ;*******************************************************************************
                        ;*      THERE ARE 4 ROUTINES AND MACRO DEFINITIONS USED FOR THE HANDLING OF
                        ;*      GPR VALUES DURING SUBROUTINE CALLS WITHIN THIS PROGRAM.  THE FOUR
                        ;*      ROUTINES/MACRO CALLS HAVE THE FOLLOWING NAMES:
                        ;*
                        ;*      SAVE - MACRO DEFINITION USED AT THE BEGINNING OF A SUBROUTINE TO
                        ;*             SAVE THE GPR CONTENTS FOR LATER RESTORATION.
                        ;*      PASS - MACRO DEFINITION USED AT THE END OF A SUBROUTINE TO RESTORE
                        ;*             THE PREVIOUSLY SAVED GPR CONTENTS AND TO LEAVE THE CONTENTS
                        ;*             OF THE SPECIFIED GPR(S) INTACT (NOT RESTORED).
                        ;*      PREG05 - SUBROUTINE WHICH IS CALLED FROM THE SAVE AND PASS MACRO
                        ;*             EXPANSIONS WHICH ACTUALLY PERFORMS THE ACTIONS ON THE GPRS.
                        ;*
                        ;*      DURING A SUBROUTINE WHICH USES THESE GPR SAVE ROUTINES THE VALUES
                        ;*      OF THE GPRS ARE STORED ON THE STACK IN THE FOLLOWING STACK FRAME:
                        ;*
                        ;*             SP    -> RET PC INTO PREG05 ROUTINE.
                        ;*             SP+2  -> GPR R0 CONTENTS.
                        ;*             SP+4  -> GPR R1 CONTENTS.
                        ;*             SP+6  -> GPR R2 CONTENTS.
                        ;*             SP+8  -> GPR R3 CONTENTS.
                        ;*             SP+10 -> GPR R4 CONTENTS.
                        ;*             SP+12 -> GPR R5 CONTENTS.
                        ;*             SP+14 -> RET PC INTO CALLER OF SUB'TNE WHICH CALLED PREG05.
                        ;*
                        ;*      EACH LEVEL OF SUB'TNE CALLING USES 8 WORDS OF STACK OVERHEAD.
                        ;*      THE SAVE AND PASS MACROS CAN ALSO BE USED IN "STRAIGHT LINE CODE"
                        ;*      TO SAVE AND RESTORE THE GPR VALUES.  IN ANY CASE, AFTER THE
                        ;*      ISSUING OF A PASS CALL THE GPRS WILL BE RESTORED TO THE VALUES
                        ;*      THEY HAD PRIOR TO THE LAST SAVE CALL (EXCEPT FOR THE EXCEPTED,
                        ;*      OR PASSED INTACT, GPRS SPECIFIED AS PARAMETERS TO THE PASS CALL)
                        ;*      AND THE SP WILL ALSO BE RESTORED TO ITS CONDITION BEFORE THE LAST
                        ;*      SAVE CALL.  THE PROGRAMMER MUST BE SURE THAT THE SP HAS THE SAME
                        ;*      VALUE WHEN THE PASS MACRO IS CALLED AS IT HAD IMMEDIATELY AFTER
                        ;*      THE SAVE MACRO WAS CALLED.
                        ;*******************************************************************************
```

```
                                .SBTTL GPR FRAME ACCESS EQUATES
                                ;+++
                                ;EQUATES THAT ALLOW ACCESS TO THE STACK FRAME.  THESE ARE THE
                                ;OFFSETS INTO THE STACK FOR REGISTERS SAVED DURING THE PREGOS
                                ;ROUTINE.
                                ;---

000036                          LPCSLT==        36      ;OFFSET FOR LAST RETURN PC.
000016                          PCSLOT==        16      ;OFFSET FOR RETURN PC.
000014                          R5SLOT==        14      ;OFFSET FOR R5.
000012                          R4SLOT==        12      ;OFFSET FOR R4.
000010                          R3SLOT==        10      ;OFFSET FOR R3.
000006                          R2SLOT==        6       ;OFFSET FOR R2.
000004                          R1SLOT==        4       ;OFFSET FOR R1.
000002                          R0SLOT==        2       ;OFFSET FOR R0.
```

```
1648                            .SBTTL  GLOBAL MACRO DEFINITION         - SAVE -
1649                            ;*********************************************************************************
1650                            ;*      THIS MACRO IS USED AT THE BEGINNING OF A SUBROUTINE TO SAVE THE
1651                            ;*      CONTENTS OF THE GPRS R0 THRU R5.
1652                            ;*
1653                            ;* INPUTS:      SP - UNCHANGED SINCE SUBROUTINE WAS ENTERED
1654                            ;*              R5SLOT - OFFSET TO STACK SLOT FOR R5 (EQUATED TO 14 OCTAL)
1655                            ;*
1656                            ;* OUTPUTS:     GPR SAVE AREA ON THE STACK IS LOADED WITH THE CONTENTS OF GPRS
1657                            ;*              TOP OF STACK - LOADED WITH THE RETURN ADDRESS INTO PREG05
1658                            ;*
1659                            ;* CALLING SEQUENCE:    SAVE
1660                            ;*
1661                            ;* COMMENTS:    NO ARGUMENTS ARE ALLOWED.
1662                            ;*              THE PASS MACRO SHOULD BE CALLED TO RESTORE THE GPR VALUES.
1663                            ;*
1664                            ;* SUBORDINATE ROUTINES CALLED: PREG05.
1665                            ;*********************************************************************************
1666
1667                                    .MACRO  SAVE
1668                                    .LIST
1669                                            JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
1670                                    .NLIST
1671                                    .ENDM   SAVE
```

```
                        .SBTTL  GLOBAL MACRO DEFINITION          - PASS -
                        ;*****************************************************************************
                        ;*      THIS MACRO IS USED IN CONJUNCTION WITH THE SAVE MACRO.  IT IS
                        ;*      CALLED AT END OF A SUBROUTINE TO PASS PARAMETERS IN GPRS BACK TO THE
                        ;*      CALLING ROUTINE BY ALTERING THE GPR SAVE AREA ON THE STACK AND THEN
                        ;*      RETURNING TO PREGO5 TO RESTORE THE GPRS TO THEIR SAVED VALUES.
                        ;*
                        ;* INPUTS:      ONLY ALLOWED ARGUMENTS ARE "R0" THRU "R5".
                        ;*              R0SLOT THRU R5SLOT MUST BE EQUATED TO THEIR RESPECTIVE GPR SAVE
                        ;*                      SLOT OFFSETS BEFORE CALLING THIS MACRO.
                        ;*
                        ;* OUTPUTS:     THE GPR VALUES ARE PUT IN THEIR RESPECTIVE SLOTS ON THE STACK.
                        ;*
                        ;* CALLING SEQUENCE:    PASS    R0,R1,...
                        ;*
                        ;* COMMENTS:    ANY COMBINATION OF GPR ARGUMENTS MAY BE LISTED IN ANY ORDER.
                        ;*              FOR EXAMPLE, THE FOLLOWING ARE LEGAL:
                        ;*                      PASS    R1
                        ;*                      PASS    R4,R0,R2
                        ;*              THE GPRS LISTED AS ARGUMENTS WILL BE PASSED INTACT TO THE
                        ;*              CALLING ROUTINE, ALL OTHER GPRS WILL BE RESTORED.
                        ;*              THE SP MUST BE AT ITS ORIGINAL VALUE WHEN PASS IS CALLED.
                        ;*
                        ;*              THE MACRO CALL
                        ;*                      PASS    R0,R3
                        ;*              EXPANDS INTO THE FOLLOWING ASSEMBLY CODE:
                        ;*                      MOV     R0,R0SLOT(SP)           ;PUT R0 IN STACK SLOT.
                        ;*                      MOV     R3,R3SLOT(SP)           ;PUT R3 IN STACK SLOT.
                        ;*                      JSR     PC,@(SP)+               ;RETURN TO PREGO5 SUBRT.
                        ;*              IN THIS EXAMPLE GPRS R1, R2, R4, AND R5 WILL BE RESTORED TO
                        ;*              THEIR VALUES CONTAINED IN THE STACK FRAME AND R0 AND R3
                        ;*              WILL BE LEFT AT THEIR VALUES PRIOR TO THIS PASS CALL.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: (PREGRT - LABEL WITHIN PREGO5, VALUE ON STACK.)
                        ;*****************************************************************************

                                .MACRO  PASS    A,B,C,D,E,F
                                .IRP    X,<A,B,C,D,E,F>
                                .IF     NB,X
                                .LIST
                                        MOV     X,X'SLOT(SP)            ;PUT X IN STACK SLOT.
                                .NLIST
                                .ENDC
                                .ENDM
                                .LIST
                                        JSR     PC,@(SP)+               ;RETURN TO PREGO5 SUBRT.
                                .NLIST
                                .ENDM   PASS
```

```
                        .SBTTL GLOBAL SUBROUTINE                - PREG05 -
                        ;*******************************************************************************
                        ;*      PRESERVE REGISTERS R0 THROUGH R5 FOR SUBROUTINE CALLS.
                        ;*
                        ;* INPUTS:      THE RETURN ADDRESS BACK INTO THE CALLING ROUTINE MUST BE IN
                        ;*              GPR R5.  (I.E.- MACROS USE "JSR  R5,PREG05".)
                        ;*
                        ;* OUTPUTS:     REGISTERS R0 THROUGH R5 ARE SAVED ON THE STACK.
                        ;*
                        ;*CALLING SEQUENCE:     SAVE            ;MACRO EXPANSION CALLS PREG05.
                        ;*                      [SUBROUTINE CODE]...
                        ;*                      PASS            ;MACRO EXPANSION RECALLS PREG05.
                        ;*
                        ;*COMMENTS:     THIS ROUTINE IS RE-ENTRANT.
                        ;*
                        ;*              PARAMETERS MAY BE PASSED OUT OF A SUBROUTINE BY MODIFYING THE
                        ;*              REGISTER SAVE AREA ON THE STACK.  USE THE PASS GPRN MACRO
                        ;*              TO RETURN GPR VALUES INTACT.
                        ;*              USE THE RNSLOT OFFSETS FROM THE SP TO PASS OTHER PARAMETERS.
                        ;*              [EXAMPLE:       MOV    VALUE,ROSLOT(SP)        ]
                        ;*              MAKE SURE THE SP IS AT ITS ORIGINAL VALUE WHEN YOU DO THIS.
                        ;*
                        ;*SUBORDINATE ROUTINES CALLED:  NONE.
                        ;*******************************************************************************

004062                  PREG05:                 ;R5 HAS BEEN LOADED ON THE STACK BY THE SUBROUTINE CALL
004062  010446                  MOV     R4,-(SP)        ;SAVE R4
004064  010346                  MOV     R3,-(SP)        ;SAVE R3
004066  010246                  MOV     R2,-(SP)        ;SAVE R2
004070  010146                  MOV     R1,-(SP)        ;SAVE R1
004072  010046                  MOV     R0,-(SP)        ;SAVE R0
004074  010546                  MOV     R5,-(SP)        ;PUSH RETURN PC ON TOP OF STACK
004076  016605  000014          MOV     R5SLOT(SP),R5   ;RESTORE R5 TO VALUE IT HAD BEFORE CALLS

004102  004736                  JSR     PC,@(SP)+       ;CALL THE SUBROUTINE AT THE RETURN ADDRESS
                                                        ;FROM THE PREG05 CALL, PUTTING THE PRESENT
                                                        ;PC ON THE STACK AS A RETURN ADDRESS INTO
                                                        ;THIS (PREG05) ROUTINE.

                                ;+++
                                ;THE FOLLOWING CODE IS EXECUTED WHEN THE CALLING ROUTINE DOES A
                                ;"RETURN" [JSR  PC,@(SP)+] USING THE PC DEPOSITED ON THE STACK ABOVE.
                                ;---

004104  012605          PREGRT:: MOV    (SP)+,R5        ;PUT RETURN PC IN R5.
004106  012600                  MOV     (SP)+,R0        ;RESTORE R0.
004110  012601                  MOV     (SP)+,R1        ;RESTORE R1.
004112  012602                  MOV     (SP)+,R2        ;RESTORE R2.
004114  012603                  MOV     (SP)+,R3        ;RESTORE R3.
004116  012604                  MOV     (SP)+,R4        ;RESTORE R4.

004120  000205                  RTS     R5              ;RETURN TO THE SUBROUTINE WHICH CALLED PREG05.
                                                          ;RESTORING R5 IN THE PROCESS.
```

```
1773                                    .SBTTL  GLOBAL TEXT SECTION
1774
1775
1776                                    ;++
1777                                    ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
1778                                    ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
1779                                    ; MORE THAN ONE TEST.
1780                                    ;--
1781
1782                                    ;
1783                                    ; NAMES OF DEVICES SUPPORTED BY PROGRAM
1784                                    ;
1785  004122                                    DEVTYP  <DHV-11>
1786  004122                                                                  L$DVTYP::
1787  004122  044104  026526  030461                                                  .ASCIZ  /DHV-11/
1788  004130     000
1789           004132                                                                 .EVEN
1790
1791
1792                                    ; TEST DESCRIPTION
1793                                    ;
1794  004132                                    DESCRIPT        <DHV-11 FUNCT TEST PART2>
1795  004132                                                                  L$DESC::
1796  004132  044104  026526  030461                                                  .ASCIZ  /DHV-11 FUNCT TE
1797  004140  043040  047125  052103
1798  004146  052040  051505  020124
1799  004154  040520  052122  000062
1800                                                                             .EVEN
1801                                    .EVEN
1802
1803
```

```
1804
1805
1806                            ;
1807                            ; FORMAT STATEMENTS USED IN PRINT CALLS
1808                            ;
1809
1810
1811
```

```
1812
1813                .NLIST BIN
1814                .SBTTL  GLOBAL MESSAGE AREA
1815                        ;********** FORMAT STATEMENTS ***********
1816  004162  MFUNIT:: .ASCIZ /%N%A TESTING UNIT :%D4%N/
1817  004170
1818  004176
1819  004204
1820  004212
1821  004213  EF0503:: .ASCIZ /%T%N/
1822  004220  EF0505:: .ASCIZ /%A    %D5%A ILLEGAL INTERRUPTS RECEIVED.%N/
1823  004226
1824  004234
1825  004242
1826  004250
1827  004256
1828  004264
1829  004272
1830  004273  EF1601:: .ASCIZ /%A  %T%A ABORTED %N/
1831  004300
1832  004306
1833  004314
1834  004317  EF3001:: .ASCIZ /%A    EXPECTED OR CORRECT VALUE: %03%N/
1835  004324
1836  004332
1837  004340
1838  004346
1839  004354
1840  004362
1841  004366  EF3002:: .ASCIZ /%A    ACTUAL OR MEASURED  VALUE: %03%N/
1842  004374
1843  004402
1844  004410
1845  004416
1846  004424
1847  004432
1848  004435  EF7801:: .ASCIZ /%T%A ON LINE %D2%A DECIMAL.%N/
1849  004442
1850  004450
1851  004456
1852  004464
1853  004472
1854  004473  EF8401:: .ASCIZ /%A  %T%A FOR LINE %D2%A(D) AFFECTS OTHER MODEM SIGNALS.%N/
1855  004500
1856  004506
1857  004514
1858  004522
1859  004530
1860  004536
1861  004544
1862  004552
1863  004560
1864  004565  EF8402:: .ASCII /%A        CHANGING %T%A FOR LINE %D2%A(D) AFFECTED /
1865  004572
1866  004600
1867  004606
```

```
1868  004614
1869  004622
1870  004630
1871  004636
1872  004644
1873  004650          .ASCIZ /%T%A FOR LINE %D2%A(D).%N/
1874  004656
1875  004664
1876  004672
1877  004700
1878  004702  EF9004:: .ASCIZ /%A    %T%A VALUE: %O3%N/
1879  004710
1880  004716
1881  004724
1882  004732  EF9005:: .ASCIZ /%A    %T%A VALUE: NONE%N/
1883  004740
1884  004746
1885  004754
1886  004762
1887  004763  EF9006:: .ASCIZ /%A  %T%A %D2%N/
1888  004770
1889  004776
1890  005002  EF9019:: .ASCIZ /%A  %T%A %O6%N/
1891  005010
1892  005016
1893  005021  EF9301:: .ASCIZ /%A %T%D2%A(D),  BMP CODE REPORTED :%O3%A(O)%N/
1894  005026
1895  005034
1896  005042
1897  005050
1898  005056
1899  005064
1900  005072
1901  005077  EF9302:: .ASCIZ /%A  OVERFLOW OCCURRED (MORE THAN 31 BMP CODES FOUND IN QUEUE)%N/
1902  005104
1903  005112
1904  005120
1905  005126
1906  005134
1907  005142
1908  005150
1909  005156
1910  005164
1911  005172
1912                ;************* MESSAGE AREA *************
1913  005177  EM0103:: .ASCIZ /DEVICE REGISTER ACCESS ERRORS/
1914  005204
1915  005212
1916  005220
1917  005226
1918  005234
1919  005235  EM0525:: .ASCIZ /  RX INTERRUPT(S) RECEIVED WITH RX INTERRUPTS DISABLED./
1920  005242
1921  005250
1922  005256
1923  005264
```

```
1924  005272
1925  005300
1926  005306
1927  005314
1928  005322
1929  005325  EM0526:: .ASCIZ /  TX INTERRUPT(S) RECEIVED WITH TX INTERRUPTS DISABLED./
1930  005332
1931  005340
1932  005346
1933  005354
1934  005362
1935  005370
1936  005376
1937  005404
1938  005412
1939  005415  EM1601:: .ASCIZ /TIMEOUT OCCURRED WAITING FOR MASTER RESET TO CLEAR/
1940  005422
1941  005430
1942  005436
1943  005444
1944  005452
1945  005460
1946  005466
1947  005474
1948  005500  EM2101:: .ASCIZ \NO TX_DATA_VALID/NO TX_ACTION TEST\
1949  005506
1950  005514
1951  005522
1952  005530
1953  005536
1954  005543  EM2102:: .ASCIZ /  TX_ACTION FOUND AFTER INVALID DATA WORD WRITTEN TO LINE: /
1955  005550
1956  005556
1957  005564
1958  005572
1959  005600
1960  005606
1961  005614
1962  005622
1963  005630
1964  005636
1965  005637  EM2201:: .ASCIZ \TX_DATA_VALID/TX_ACTION TEST\
1966  005644
1967  005652
1968  005660
1969  005666
1970  005674  EM2202:: .ASCIZ /  NO TX_ACTION FOUND AFTER VALID DATA WORD TX'D ON LINE: /
1971  005702
1972  005710
1973  005716
1974  005724
1975  005732
1976  005740
1977  005746
1978  005754
1979  005762
```

```
1980  005766  EM2203:: .ASCIZ /  INCORRECT LINE NUMBER FOUND WITH TX_ACT AFTER DATA TX'D ON LINE : /
1981  005774
1982  006002
1983  006010
1984  006016
1985  006024
1986  006032
1987  006040
1988  006046
1989  006054
1990  006062
1991  006070
1992  006073  EM2301:: .ASCIZ /TX_ENABLE (INACTIVE) BIT TEST/
1993  006100
1994  006106
1995  006114
1996  006122
1997  006130
1998  006131  EM2302:: .ASCIZ /  TX_ENABLE BIT BAD ON LINE: /
1999  006136
2000  006144
2001  006152
2002  006160
2003  006166
2004  006167  EM2401:: .ASCIZ /TX_ENABLE (ACTIVE) BIT TEST/
2005  006174
2006  006202
2007  006210
2008  006216
2009  006223  EM2601:: .ASCIZ /RECEIVE INTERRUPT TEST /
2010  006230
2011  006236
2012  006244
2013  006252
2014  006253  EM2602:: .ASCIZ /  NO RX INT GENERATED (DATA_VALID SET, RX INTS ENABLED)./
2015  006260
2016  006266
2017  006274
2018  006302
2019  006310
2020  006316
2021  006324
2022  006332
2023  006340
2024  006344  EM2603:: .ASCIZ /  NO RX INT GENERATED (NO CODES IN FIFO AFTER RESET)./
2025  006352
2026  006360
2027  006366
2028  006374
2029  006402
2030  006410
2031  006416
2032  006424
2033  006432  EM2604:: .ASCIZ /  NO RX INT GENERATED (RX_DATA_AVAIL CLR, RX INTS ENABLED)./
2034  006440
2035  006446
```

```
2036  006454
2037  006462
2038  006470
2039  006476
2040  006504
2041  006512
2042  006520
2043  006526  EM2605:: .ASCIZ /  RX INTERRUPT GENERATED WITH RX_DATA_AVAIL CLEAR./
2044  006534
2045  006542
2046  006550
2047  006556
2048  006564
2049  006572
2050  006600
2051  006606
2052  006611  EM2606:: .ASCIZ /TRANSMIT INTERRUPT TEST ERROR:/
2053  006616
2054  006624
2055  006632
2056  006640
2057  006646
2058  006650  EM2607:: .ASCIZ /  TX_ACTION SET REPEATEDLY AFTER BOARD RESET, NO DATA SENT./
2059  006656
2060  006664
2061  006672
2062  006700
2063  006706
2064  006714
2065  006722
2066  006730
2067  006736
2068  006744  EM2608:: .ASCIZ /  TX_ACTION STUCK SET AFTER BOARD RESET./
2069  006752
2070  006760
2071  006766
2072  006774
2073  007002
2074  007010
2075  007015  EM2609:: .ASCIZ /  TX INTERRUPT GENERATED WITH TX_ACTION CLEAR./
2076  007022
2077  007030
2078  007036
2079  007044
2080  007052
2081  007060
2082  007066
2083  007074  EM2610:: .ASCIZ /  NO TX INTERRUPT WITH TX_ACTION SET AND TX INTS ENABLED./
2084  007102
2085  007110
2086  007116
2087  007124
2088  007132
2089  007140
2090  007146
2091  007154
```

```
2092  007162
2093  007166  EM2611:: .ASCIZ /  TX_ACTION NOT SET AFTER CHARS SENT ON ALL LINES./
2094  007174
2095  007202
2096  007210
2097  007216
2098  007224
2099  007232
2100  007240
2101  007246
2102  007251  EM3001:: .ASCIZ /INTERRUPT BR LEVEL TEST /
2103  007256
2104  007264
2105  007272
2106  007300
2107  007302  EM3002:: .ASCIZ /  NO RX_DATA_AVAIL FROM SELFTEST CODES IN FIFO AFTER RESET./
2108  007310
2109  007316
2110  007324
2111  007332
2112  007340
2113  007346
2114  007354
2115  007362
2116  007370
2117  007376  EM3003:: .ASCIZ /  TX INTERRUPT GENERATED AT WRONG BR LEVEL:/
2118  007404
2119  007412
2120  007420
2121  007426
2122  007434
2123  007442
2124  007450
2125  007452  EM3004:: .ASCIZ /  RX INTERRUPT GENERATED AT WRONG BR LEVEL:/
2126  007460
2127  007466
2128  007474
2129  007502
2130  007510
2131  007516
2132  007524
2133  007526  EM3005:: .ASCIZ /  TX INTERRUPT GIVEN PRECEDENCE OVER SIMULTANEOUS RX INT./
2134  007534
2135  007542
2136  007550
2137  007556
2138  007564
2139  007572
2140  007600
2141  007606
2142  007614
2143  007620  EM3101:: .ASCIZ /DIAGNOSTIC FIELD (BMP) TEST/
2144  007626
2145  007634
2146  007642
2147  007650
```

```
2148  007654  EM3102:: .ASCIZ /  DIAGNOSTIC FIELD BAD ON LINE: /
2149  007662
2150  007670
2151  007676
2152  007704
2153  007712
2154  007715  EM4001:: .ASCIZ /DMA_START BIT TEST/
2155  007722
2156  007730
2157  007736
2158  007740  EM4002:: .ASCIZ /DMA_START BIT BAD ON LINE: /
2159  007746
2160  007754
2161  007762
2162  007770
2163  007774  EM4101:: .ASCIZ /DMA_ABORT BIT TEST/
2164  010002
2165  010010
2166  010016
2167  010017  EM4102:: .ASCIZ /DMA_ABORT BIT BAD ON LINE: /
2168  010024
2169  010032
2170  010040
2171  010046
2172  010053  EM4103:: .ASCIZ /DMA_START BIT FOUND SET AFTER DMA ABORTED ON LINE: /
2173  010060
2174  010066
2175  010074
2176  010102
2177  010110
2178  010116
2179  010124
2180  010132
2181  010137  EM4901:: .ASCIZ /OAUTO (INACTIVE) BIT TEST/
2182  010144
2183  010152
2184  010160
2185  010166
2186  010171  EM4902:: .ASCIZ /  OAUTO BIT BAD ON LINE: /
2187  010176
2188  010204
2189  010212
2190  010220
2191  010223  EM5001:: .ASCIZ /OAUTO (ACTIVE) BIT TEST/
2192  010230
2193  010236
2194  010244
2195  010252
2196  010253  EM5101:: .ASCIZ /IAUTO (INACTIVE) TEST/
2197  010260
2198  010266
2199  010274
2200  010301  EM5102:: .ASCIZ /IAUTO BIT FOUND SET ON LINE: /
2201  010306
2202  010314
2203  010322
```

```
2204  010330
2205  010336
2206  010337  EM5103:: .ASCIZ /IAUTO BIT BAD ON LINE: /
2207  010344
2208  010352
2209  010360
2210  010366
2211  010367  EM5201:: .ASCIZ /IAUTO (ACTIVE) TEST/
2212  010374
2213  010402
2214  010410
2215  010413  EM5202:: .ASCIZ /IAUTO BIT FOUND CLR ON LINE: /
2216  010420
2217  010426
2218  010434
2219  010442
2220  010450
2221  010451  EM5301:: .ASCIZ /FIFO VALID DATA TEST/
2222  010456
2223  010464
2224  010472
2225  010476  EM5302:: .ASCIZ /FIFO BAD DATA FIELD CORRUPTED, TEST USED LINE:/
2226  010504
2227  010512
2228  010520
2229  010526
2230  010534
2231  010542
2232  010550
2233  010555  EM5303:: .ASCIZ /BMP CODE FOUND IN FIFO, TEST INVAILDATED/
2234  010562
2235  010570
2236  010576
2237  010604
2238  010612
2239  010620
2240  010626  EM5401:: .ASCIZ \FIFO 3/4 ALARM (INACTIVE) TEST\
2241  010634
2242  010642
2243  010650
2244  010656
2245  010664
2246  010665  EM5402:: .ASCIZ /FIFO BAD, ALARM SIGNAL DEFECTIVE/
2247  010672
2248  010700
2249  010706
2250  010714
2251  010722
2252  010726  EM5501:: .ASCIZ \FIFO 3/4 ALARM (ACTIVE) TEST\
2253  010734
2254  010742
2255  010750
2256  010756
2257  010763  EM5601:: .ASCIZ \FIFO 3/4 ALARM (ACTIVE/INACTIVE) TEST\
2258  010770
2259  010776
```

```
2260  011004
2261  011012
2262  011020
2263  011026
2264  011031  EM5701:: .ASCIZ \FIFO 1/2 LEVEL (ACTIVE/INACTIVE) TEST\
2265  011036
2266  011044
2267  011052
2268  011060
2269  011066
2270  011074
2271  011077  EM7801:: .ASCIZ /MODEM CONTROL DTR BIT TEST/
2272  011104
2273  011112
2274  011120
2275  011126
2276  011132  EM7802:: .ASCIZ / DTR BIT FAULTY ON LINE:/
2277  011140
2278  011146
2279  011154
2280  011162
2281  011163  EM7901:: .ASCIZ /MODEM CONTROL RTS BIT TEST/
2282  011170
2283  011176
2284  011204
2285  011212
2286  011216  EM7902:: .ASCIZ / RTS BIT FAULTY ON LINE:/
2287  011224
2288  011232
2289  011240
2290  011246
2291  011247  EM8001:: .ASCIZ /DSR MODEM STATUS SIGNAL TEST /
2292  011254
2293  011262
2294  011270
2295  011276
2296  011304
2297  011305  EM8002:: .ASCIZ /  DSR MODEM STATUS SIGNAL DEFECTIVE/
2298  011312
2299  011320
2300  011326
2301  011334
2302  011342
2303  011350
2304  011351  EM8101:: .ASCIZ /RI MODEM STATUS SIGNAL TEST /
2305  011356
2306  011364
2307  011372
2308  011400
2309  011406  EM8102:: .ASCIZ /  RI MODEM STATUS SIGNAL DEFECTIVE/
2310  011414
2311  011422
2312  011430
2313  011436
2314  011444
2315  011451  EM8201:: .ASCIZ /CTS MODEM STATUS SIGNAL TEST /
```

```
2316  011456
2317  011464
2318  011472
2319  011500
2320  011506
2321  011507  EM8202:: .ASCIZ /  CTS MODEM STATUS SIGNAL DEFECTIVE/
2322  011514
2323  011522
2324  011530
2325  011536
2326  011544
2327  011552
2328  011553  EM8301:: .ASCIZ /DCD MODEM STATUS SIGNAL TEST /
2329  011560
2330  011566
2331  011574
2332  011602
2333  011610
2334  011611  EM8302:: .ASCIZ /  DCD MODEM STATUS SIGNAL DEFECTIVE/
2335  011616
2336  011624
2337  011632
2338  011640
2339  011646
2340  011654
2341  011655  EM8401:: .ASCIZ /DTR MODEM CONTROL SIGNAL INTERACTIONS TEST /
2342  011662
2343  011670
2344  011676
2345  011704
2346  011712
2347  011720
2348  011726
2349  011731  EM8402:: .ASCIZ /DTR/
2350  011735  EM8403:: .ASCIZ /DSR/
2351  011741  EM8404:: .ASCIZ /RI/
2352  011744  EM8405:: .ASCIZ /DCD/
2353  011750  EM8406:: .ASCIZ /CTS/
2354  011754  EM8501:: .ASCIZ /RTS MODEM CONTROL SIGNAL INTERACTIONS TEST /
2355  011762
2356  011770
2357  011776
2358  012004
2359  012012
2360  012020
2361  012026
2362  012030  EM8502:: .ASCIZ /RTS/
2363  012034  EM9009:: .ASCIZ /EXPECTED OR CORRECT/
2364  012042
2365  012050
2366  012056
2367  012060  EM9010:: .ASCIZ /ACTUAL OR MEASURED /
2368  012066
2369  012074
2370  012102
2371  012104  EM9026:: .ASCIZ /     LPR CONTENTS: /
```

```
2372  012112
2373  012120
2374  012126
2375  012130  EM9301:: .ASCIZ /BMP CODE REPORT/
2376  012136
2377  012144
2378  012150  EM9302:: .ASCIZ /BMP CODE FOUND IN TEST /
2379  012156
2380  012164
2381  012172
2382  012200  EM9303:: .ASCIZ /THE LAST BMP CODE WAS FOUND IN TEST /
2383  012206
2384  012214
2385  012222
2386  012230
2387  012236
2388  012244
2389  012245  EM9304:: .ASCIZ /UNEXPECTED BMP CODES FOUND DURING THIS PASS/
2390  012252
2391  012260
2392  012266
2393  012274
2394  012302
2395  012310
2396  012316
2397
2398                    .EVEN
2399                                .LIST BIN
```

```
2400
2401
2402                            .SBTTL  GLOBAL ERROR REPORT SECTION
2403
2404                            ;++
2405                            ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
2406                            ; USED BY MORE THAN ONE TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
2407                            ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
2408                            ;--
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER0101 -
                        ;*****************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ADDITIONAL ERROR
                        ;*      INFORMATION IF AN ERROR IS DETECTED IN TEST 1 (REGISTER ADDRESS
                        ;*      ACCESS TEST).  THIS SUBROUTINE REPORTS THE TYPE OF ACCESS (READ OR
                        ;*      WRITE OR BOTH) WHICH CAUSED A BUS TIME-OUT TRAP (004 TRAP).
                        ;*      A MESSAGE INDICATING THAT THE DHV MAY BE AT THE WRONG Q-BUS ADDRESS
                        ;*      IS ALSO PRINTED.
                        ;*
                        ;* INPUTS:      R5 - ERROR FLAG WORD.
                        ;*                      IF BIT 0 IS SET, A READ ERROR OCCURED.
                        ;*                      IF BIT 1 IS SET, A WRITE ERROR OCCURED.
                        ;*
                        ;* OUTPUTS:     MESSAGES ARE PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    INCLUDE THE LABEL "ER0101" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DRS ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES USED: NONE.
                        ;*****************************************************************************

012322                          BGNMSG  ER0101
012322                                                                  ER0101::
012322                          SAVE                    ;SAVE THE GPR CONTENTS.
012322  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

012326  032705  000001          BIT     #BIT0,R5        ;TEST FOR READ ERROR.
012332  001410                  BEQ     2$              ;SKIP READ ERROR MSG IF NO READ ERROR.
012334                          PRINTB  #MSG1           ;PRINT READ ERROR MESSAGE.
012334  012746  012426                                          MOV     #MSG1,-(SP)
012340  012746  000001                                          MOV     #1,-(SP)
012344  010600                                                  MOV     SP,R0
012346  104414                                                  TRAP    C$PNTB
012350  062706  000004                                          ADD     #4,SP
012354  032705  000002  2$:     BIT     #BIT1,R5        ;TEST FOR WRITE ERROR.
012360  001410                  BEQ     4$              ;SKIP WRITE ERROR MSG IF NO WRITE ERROR.
012362                          PRINTB  #MSG2           ;PRINT WRITE ERROR MESSAGE.
012362  012746  012504                                          MOV     #MSG2,-(SP)
012366  012746  000001                                          MOV     #1,-(SP)
012372  010600                                                  MOV     SP,R0
012374  104414                                                  TRAP    C$PNTB
012376  062706  000004                                          ADD     #4,SP
012402                  4$:     PRINTX  #MSG3           ;SUGGEST THAT DHV MAY BE AT WRONG ADDRESS.
012402  012746  012563                                          MOV     #MSG3,-(SP)
012406  012746  000001                                          MOV     #1,-(SP)
012412  010600                                                  MOV     SP,R0
012414  104415                                                  TRAP    C$PNTX
012416  062706  000004                                          ADD     #4,SP
012422                          PASS                    ;RESTORE THE GPR CONTENTS.
012422  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
012424                          ENDMSG
012424                                                                  L10002:
012424  104423                                                          TRAP    C$MSG
```

```
2465  012426  040445  052502  020123  MSG1::  .ASCIZ  /%ABUS TIME-OUT TRAP CAUSED BY READ ATTEMPT.%N/
2466  012434  044524  042515  047455
2467  012442  052125  052040  040522
2468  012450  020120  040503  051525
2469  012456  042105  041040  020131
2470  012464  042522  042101  040440
2471  012472  052124  046505  052120
2472  012500  022456  000116
2473  012504  040445  052502  020123  MSG2::  .ASCIZ  /%ABUS TIME-OUT TRAP CAUSED BY WRITE ATTEMPT.%N/
2474  012512  044524  042515  047455
2475  012520  052125  052040  040522
2476  012526  020120  040503  051525
2477  012534  042105  041040  020131
2478  012542  051127  052111  020105
2479  012550  052101  042524  050115
2480  012556  027124  047045     000
2481  012563     045  042101  053110  MSG3::  .ASCIZ  /%ADHV MAY BE AT THE WRONG Q-BUS ADDRESS.%N%N/
2482  012570  046440  054501  041040
2483  012576  020105  052101  052040
2484  012604  042510  053440  047522
2485  012612  043516  050440  041055
2486  012620  051525  040440  042104
2487  012626  042522  051523  022456
2488  012634  022516  000116
2489
2490                                  .EVEN
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER0503 -
                        ;*****************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS AN ADDITIONAL ERROR
                        ;*      MESSAGE WHOSE ADDRESS IS PASSED AS AN INPUT PARAMETER.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF THE MESSAGE TO PRINT.
                        ;*
                        ;* OUTPUTS:     A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    LOAD THE ADDRESS OF THE MESSAGE IN R1.
                        ;*                      INCLUDE THE LABEL "ER0503" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
                        ;*
                        ;* SUBORDINATE ROUTINES USED: NONE.
                        ;*****************************************************************************

012640                          BGNMSG  ER0503
012640                                                                  ER0503::

012640                          PRINTB  #EF0503,R1      ;PRINT THE MESSAGE.
012640  010146                                                          MOV     R1,-(SP)
012642  012746  004213                                                  MOV     #EF0503,-(SP)
012646  012746  000002                                                  MOV     #2,-(SP)
012652  010600                                                          MOV     SP,R0
012654  104414                                                          TRAP    C$PNTB
012656  062706  000006                                                  ADD     #6,SP

012662                          ENDMSG
012662                                                          L10003:
012662  104423                                                          TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER0504 -
                        ;*****************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ADDITIONAL ERROR
                        ;*      MESSAGES WHEN ILLEGAL INTERRUPTS ARE RECEIVED.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF THE MESSAGE TO PRINT.
                        ;*              R2 - NUMBER OF ILLEGAL INTERRUPTS RECEIVED.
                        ;*
                        ;* OUTPUTS:     MESSAGESS ARE PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    LOAD THE ADDRESS OF THE MESSAGE IN R1.
                        ;*                      LOAD THE NUMBER OF ILLEGAL INTS IN R2.
                        ;*                      INCLUDE THE LABEL "ER0504" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES USED: NONE.
                        ;*****************************************************************************

                              BGNMSG  ER0504
012664                                                          ER0504::
012664
                              PRINTB  #EF0503,R1        ;PRINT THE FIRST LINE OF THE MESSAGE.
012664  010146                                                          MOV     R1,-(SP)
012666  012746  004213                                                  MOV     #EF0503,-(SP)
012672  012746  000002                                                  MOV     #2,-(SP)
012676  010600                                                          MOV     SP,R0
012700  104414                                                          TRAP    C$PNTB
012702  062706  000006                                                  ADD     #6,SP
012706                        PRINTX  #EF0505,R2        ;PRINT THE NUMBER OF INTS RECEIVED.
012706  010246                                                          MOV     R2,-(SP)
012710  012746  004220                                                  MOV     #EF0505,-(SP)
012714  012746  000002                                                  MOV     #2,-(SP)
012720  010600                                                          MOV     SP,R0
012722  104415                                                          TRAP    C$PNTX
012724  062706  000006                                                  ADD     #6,SP
                        ENDMSG
012730                                                          L10004:
012730
012730  104423                                                          TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER1603 -
                        ;*********************************************************************
                        ;*      THIS ERROR REPORTING ROUTINE IS USED TO PRINT OUT A BASIC ERROR
                        ;*      MESSAGE, ALONG WITH A MESSAGE INFORMING THE OPERATOR WHICH TEST IS
                        ;*      ABOUT TO BE ABORTED.
                        ;*
                        ;* INPUTS:      R1 - CONTAINS THE ADDRESS OF THE MESSAGE TO BE PRINTED.
                        ;*              ERRMSG - CONTAINS THE ADDRESS OF THE MESSAGE THAT INDICATES
                        ;*              THE TEST THAT IS BEING PERFORMED, EG DMA, BREAK ETC.
                        ;*
                        ;* OUTPUTS:     MESSAGES ARE PRINTED AT THE OPERATORS CONSOLE.
                        ;*              "TESTNAME TEST ABORTED"
                        ;*
                        ;* CALLING SEQUENCE:    INCLUDE THE LABLE "ER1603" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DRS ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;*********************************************************************
012732                          BGNMSG  ER1603
012732                                                                  ER1603::
012732                          SAVE                    ;SAVE THE CONTENTS OF THE GPRS.
012732  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

012736                  PRINTB #EF0503,R1       ;PRINT BASIC MESSAGE ON OPERATORS CONSOLE.
012736  010146                                                  MOV     R1,-(SP)
012740  012746  004213                                          MOV     #EF0503,-(SP)
012744  012746  000002                                          MOV     #2,-(SP)
012750  010600                                                  MOV     SP,R0
012752  104414                                                  TRAP    C$PNTB
012754  062706  000006                                          ADD     #6,SP

012760  013702  004056          MOV     ERRMSG,R2       ;GET THE "TEST MESSAGE".
012764                  PRINTB #EF1601,R2       ;PRINT "TEST ABORTED" MESSAGE.
012764  010246                                                  MOV     R2,-(SP)
012766  012746  004273                                          MOV     #EF1601,-(SP)
012772  012746  000002                                          MOV     #2,-(SP)
012776  010600                                                  MOV     SP,R0
013000  104414                                                  TRAP    C$PNTB
013002  062706  000006                                          ADD     #6,SP

013006                  PASS                    ;RESTORE THE CONTENTS OF THE GPRS.
013006  004736                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
013010                  ENDMSG
013010                                                  L10005:
013010  104423                                                  TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER3001 -
                        ;*****************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH IS INTENDED FOR USE IN THE
                        ;*      INTERRUPT BR LEVEL TEST.  IT REPORTS ADDITIONAL INFORMATION WHEN AN
                        ;*      INTERRUPT HAS OCCURRED AT THE WRONG BR LEVEL.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF MESSAGE TO PRINT FIRST.
                        ;*              R4 - BR LEVEL AT WHICH THE INT REQUEST OCCURRED.
                        ;*              R5 - EXPECTED OR CORRECT BR LEVEL FOR THE DUT.
                        ;*
                        ;* OUTPUTS:     A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    INCLUDE THE LABEL "ER3001" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
                        ;*
                        ;* SUBORDINATE ROUTINES USED: NONE.
                        ;*****************************************************************************

                              BGNMSG  ER3001
013012                                                          ER3001::
013012
                              PRINTB  #EF0503,R1        ;PRINT THE FIRST LINE OF THE MESSAGE.
013012  010146                                                  MOV     R1,-(SP)
013014  012746  004213                                          MOV     #EF0503,-(SP)
013020  012746  000002                                          MOV     #2,-(SP)
013024  010600                                                  MOV     SP,R0
013026  104414                                                  TRAP    C$PNTB
013030  062706  000006                                          ADD     #6,SP
013034
                              PRINTX  #EF3001,R5        ;REPORT EXPECTED BR LEVEL.
013034  010546                                                  MOV     R5,-(SP)
013036  012746  004317                                          MOV     #EF3001,-(SP)
013042  012746  000002                                          MOV     #2,-(SP)
013046  010600                                                  MOV     SP,R0
013050  104415                                                  TRAP    C$PNTX
013052  062706  000006                                          ADD     #6,SP
013056
                              PRINTX  #EF3002,R4        ;REPORT ACTUAL BR LEVEL.
013056  010446                                                  MOV     R4,-(SP)
013060  012746  004366                                          MOV     #EF3002,-(SP)
013064  012746  000002                                          MOV     #2,-(SP)
013070  010600                                                  MOV     SP,R0
013072  104415                                                  TRAP    C$PNTX
013074  062706  000006                                          ADD     #6,SP
013100
                              ENDMSG
013100                                                  L10006:
013100  104423                                                  TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER7801 -
                        ;*****************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS AN ADDITIONAL ERROR
                        ;*      MESSAGE WHOSE ADDRESS IS PASSED AS AN INPUT PARAMETER.  A LINE NUMBER
                        ;*      IS INCLUDED AT THE END OF THE MESSAGE.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF THE MESSAGE TO PRINT.
                        ;*              R3 - NUMBER OF LINE ON WHICH ERROR OCCURRED.
                        ;*
                        ;* OUTPUTS:     A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    LOAD THE ADDRESS OF THE MESSAGE IN R1.
                        ;*                      LOAD THE LINE NUMBER INTO R3.
                        ;*                      INCLUDE THE LABEL "ER7801" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
                        ;*
                        ;* SUBORDINATE ROUTINES USED: NONE.
                        ;*****************************************************************************

013102                          BGNMSG  ER7801
013102                                                          ER7801::

                                PRINTB  #EF7801,R1,R3   ;PRINT THE MESSAGE.
013102  010346                                                          MOV     R3,-(SP)
013104  010146                                                          MOV     R1,-(SP)
013106  012746  004435                                                  MOV     #EF7801,-(SP)
013112  012746  000003                                                  MOV     #3,-(SP)
013116  010600                                                          MOV     SP,R0
013120  104414                                                          TRAP    C$PNTB
013122  062706  000010                                                  ADD     #10,SP

013126                  ENDMSG
013126                                                  L10007:
013126  104423                                                          TRAP    C$MSG
```

```
2696                            .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER8401 -
2697                            ;*****************************************************************************
2698                            ;*      THIS ERROR REPORTING SUBROUTINE IS INTENDED TO REPORT INTERACTIONS
2699                            ;*      WHICH HAVE BEEN FOUND BETWEEN A MODEM SIGNAL AND OTHER MODEM SIGNALS.
2700                            ;*      IT ANALYZES THE MODEM STATUS WHICH IS STORED IN THE STAT STORAGE AREA
2701                            ;*      AND REPORTS ANY DISCREPANCIES WHICH ARE FOUND BETWEEN THIS STORED DATA
2702                            ;*      AND THE PRESENT STATE OF THE STAT REGISTERS.  SPECIFIED BITS ON THE
2703                            ;*      LINE ASSOCIATED WITH THE SPECIFIED LINE ARE IGNORED.
2704                            ;*
2705                            ;* INPUTS:       R1 - ADDRESS OF SIGNAL NAME MESSAGE.
2706                            ;*               R2 - BIT MAP OF BITS TO IGNORE ON SPECIFIED LINE.
2707                            ;*               R3 - NUMBER OF SPECIFIED LINE.
2708                            ;*               CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
2709                            ;*               NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
2710                            ;*               STATA - CONTAINS THE ADDRESS OF THE DUT STAT REGISTER.
2711                            ;*               STSTB - LABEL AT BASE OF STAT STORAGE TABLE.
2712                            ;*               TXRLNB - LABEL AT BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
2713                            ;*
2714                            ;* OUTPUTS:      A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
2715                            ;*
2716                            ;* CALLING SEQUENCE:     INCLUDE THE LABEL "ER8401" AS THE MESSAGE POINTER
2717                            ;*                       PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
2718                            ;*
2719                            ;* COMMENTS:     THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
2720                            ;*
2721                            ;* SUBORDINATE ROUTINES USED: NONE.
2722                            ;*****************************************************************************
2723
2724  013130                        BGNMSG  ER8401
2725  013130                                                                  ER8401::
2726  013130                        SAVE                    ;PRESERVE THE CONTENTS OF THE GPRS.
2727  013130  004537  004062                JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
2728
2729  013134                        PRINTB  #EF8401,R1,R3   ;PRINT THE BASIC MESSAGE.
2730  013134  010346                                                         MOV     R3,-(SP)
2731  013136  010146                                                         MOV     R1,-(SP)
2732  013140  012746  004473                                                 MOV     #EF8401,-(SP)
2733  013144  012746  000003                                                 MOV     #3,-(SP)
2734  013150  010600                                                         MOV     SP,R0
2735  013152  104414                                                         TRAP    C$PNTB
2736  013154  062706  000010                                                 ADD     #10,SP
2737
2738  013160  010137  013370                MOV     R1,44$          ;SAVE THE ADDRESS OF THE SIGNAL NAME MESSAGE.
2739  013164  005001                        CLR     R1              ;CLEAR THE LINE COUNTER.
2740  013166  012704  002652                MOV     #STSTB,R4       ;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
2741  013172  010177  167050        2$:     MOV     R1,@CSRA        ;SET UP THE CSR IND.ADR.REG FIELD.
2742  013176  017700  167052                MOV     @STATA,R0       ;GET THE CONTENTS OF THIS LINE'S STAT REGISTER.
2743  013202  011405                        MOV     (R4),R5         ;GET THE PREVIOUS CONTENTS FROM STORAGE.
2744  013204  040005                        BIC     R0,R5
2745  013206  042400                        BIC     (R4)+,R0
2746  013210  050005                        BIS     R0,R5           ;XOR PRESENT AND STORED STAT VALUES.
2747  013212  012700  043777                MOV     #43777,R0       ;PREPARE TO MASK OUT UNUSED BITS.
2748  013216  120163  004012                CMPB    R1,TXRLNB(R3)   ;IS THIS LINE ASSOCIATED WITH SPECIFIED LINE?
2749  013222  001002                        BNE     4$              ;DON'T MASK OUT SPECIFIED BITS IF IT IS NOT.
2750  013224  056600  000006                BIS     R2SLOT(SP),R0   ;MASK OUT SPECIFIED BITS.
2751  013230  040005        4$:     BIC     R0,R5           ;GET BIT MAP OF UNDESIRED CHANGES.
```

```
2752  013232  032705  100000                BIT     #BIT15,R5       ;CHECK FOR DSR SIGNAL INTERACTION.
2753  013236  001404                        BEQ     6$              ;SKIP PRINTING LINE IF NO DSR INTERACTION.
2754  013240  012702  011735                MOV     #EM8403,R2      ;SELECT DSR ERROR MESSAGE.
2755  013244  004737  013334                JSR     PC,40$          ;PRINT THE LINE OF THE ERROR MESSAGE.
2756  013250  032705  020000        6$:     BIT     #BIT13,R5       ;CHECK FOR RI SIGNAL INTERACTION.
2757  013254  001404                        BEQ     8$              ;SKIP PRINTING LINE IF NO RI INTERACTION.
2758  013256  012702  011741                MOV     #EM8404,R2      ;SELECT RI ERROR MESSAGE.
2759  013262  004737  013334                JSR     PC,40$          ;PRINT THE LINE OF THE ERROR MESSAGE.
2760  013266  032705  010000        8$:     BIT     #BIT12,R5       ;CHECK FOR DCD SIGNAL INTERACTION.
2761  013272  001404                        BEQ     10$             ;SKIP PRINTING LINE IF NO DCD INTERACTION.
2762  013274  012702  011744                MOV     #EM8405,R2      ;SELECT DCD ERROR MESSAGE.
2763  013300  004737  013334                JSR     PC,40$          ;PRINT THE LINE OF THE ERROR MESSAGE.
2764  013304  032705  004000        10$:    BIT     #BIT11,R5       ;CHECK FOR CTS SIGNAL INTERACTION.
2765  013310  001404                        BEQ     12$             ;SKIP PRINTING LINE IF NO CTS INTERACTION.
2766  013312  012702  011750                MOV     #EM8406,R2      ;SELECT CTS ERROR MESSAGE.
2767  013316  004737  013334                JSR     PC,40$          ;PRINT THE LINE OF THE ERROR MESSAGE.
2768
2769  013322  005201        12$:    INC     R1              ;SELECT NEXT LINE.
2770  013324  020127  000010                CMP     R1,#NUMLNS      ;ALL LINES DONE?
2771  013330  002720                        BLT     2$              ;LOOP IF NOT ALL LINES DONE.
2772  013332  000417                        BR      60$             ;EXIT THIS ROUTINE.
2773                                ;+
2774                                ; LOCAL ERROR MESSAGE LINE PRINTING ROUTINE.
2775                                ;-
2776  013334                        40$:    PRINTX  #EF8402,44$,R3,R2,R1
2777  013334  010146                                                                MOV     R1,-(SP)
2778  013336  010246                                                                MOV     R2,-(SP)
2779  013340  010346                                                                MOV     R3,-(SP)
2780  013342  013746  013370                                                        MOV     44$,-(SP)
2781  013346  012746  004565                                                        MOV     #EF8402,-(SP)
2782  013352  012746  000005                                                        MOV     #5,-(SP)
2783  013356  010600                                                                MOV     SP,R0
2784  013360  104415                                                                TRAP    C$PNTX
2785  013362  062706  000014                                                        ADD     #14,SP
2786  013366  000207                RTS     PC              ;
2787  013370  000000        44$:    .WORD   0               ;LOCAL STORAGE FOR ADDRESS OF SIGNAL NAME.
2788  013372                60$:    PASS                    ;RESTORE ALL THE GPRS TO THE PRESERVED VALUES.
2789  013372  004736                        JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
2790  013374                        ENDMSG
2791  013374                                                                 L10010:
2792  013374  104423                                                                TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER9002 -
                        ;*******************************************************************************
                        ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH IS INTENDED FOR USE IN THE
                        ;*      TRANMSISSION AND RECEPTION TESTS.  IT REPORTS THE TYPE OF ERROR WHICH
                        ;*      HAS OCCURRED WHEN INCORRECT DATA IS RECEIVED FROM THE DUT.  THIS
                        ;*      ROUTINE ALSO REPORTS THE READ AND EXPECTED DATA VALUES.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF MESSAGE TO PRINT FIRST.
                        ;*              R2 - DATA BYTE READ FROM THE DUT.
                        ;*              R3 - LINE NUMBER MULTIPLIED BY 2.
                        ;*              R4 - EXPECTED DATA BYTE, BIT 15 SET IF 'NONE".
                        ;*
                        ;* OUTPUTS:     A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    INCLUDE THE LABEL "ER9002" AS THE MESSAGE POINTER
                        ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                        ;*
                        ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
                        ;*
                        ;* SUBORDINATE ROUTINES USED: PRTLPR.
                        ;*******************************************************************************

                              BGNMSG  ER9002
013376                                                          ER9002::
013376
013376  006203                 ASR     R3              ;CALCULATE THE LINE NUMBER.
013400  042702  177400         BIC     #177400,R2      ;MASK OUT ALL BUT DATA IN READ CHAR.
013404                       PRINTB  #EF9006,R1,R3     ;PRINT THE FIRST LINE OF THE MESSAGE.
013404  010346                                                  MOV     R3,-(SP)
013406  010146                                                  MOV     R1,-(SP)
013410  012746  004763                                         MOV     #EF9006,-(SP)
013414  012746  000003                                         MOV     #3,-(SP)
013420  010600                                                  MOV     SP,R0
013422  104414                                                  TRAP    C$PNTB
013424  062706  000010                                         ADD     #10,SP
013430                       PRINTX  #EF9004,#EM9010,R2       ;PRINT ACTUAL DATA.
013430  010246                                                  MOV     R2,-(SP)
013432  012746  012060                                         MOV     #EM9010,-(SP)
013436  012746  004702                                         MOV     #EF9004,-(SP)
013442  012746  000003                                         MOV     #3,-(SP)
013446  010600                                                  MOV     SP,R0
013450  104415                                                  TRAP    C$PNTX
013452  062706  000010                                         ADD     #10,SP
013456  005704                 TST     R4              ;CHECK FOR "NONE" CODE SET IN EXPECTED DATA.
013460  100414               BMI     2$              ;BRANCH TO PRINT "NONE" MESSAGE IF FLAG SET.
013462                       PRINTX  #EF9004,#EM9009,R4       ;PRINT EXPECTED DATA.
013462  010446                                                  MOV     R4,-(SP)
013464  012746  012034                                         MOV     #EM9009,-(SP)
013470  012746  004702                                         MOV     #EF9004,-(SP)
013474  012746  000003                                         MOV     #3,-(SP)
013500  010600                                                  MOV     SP,R0
013502  104415                                                  TRAP    C$PNTX
013504  062706  000010                                         ADD     #10,SP
013510  000412                 BR      60$             ;EXIT THIS ROUTINE.
013512                  2$:    PRINTX  #EF9005,#EM9009 ;PRINT MESSAGE INDICATING NO EXPECTED DATA.
013512  012746  012034                                         MOV     #EM9009,-(SP)
```

```
2849  013516  012746  004732                                                  MOV     #EF9005,-(SP)
2850  013522  012746  000002                                                  MOV     #2,-(SP)
2851  013526  010600                                                          MOV     SP,R0
2852  013530  104415                                                          TRAP    C$PNTX
2853  013532  062706  000006                                                  ADD     #6,SP
2854  013536  004737  015574          60$:    JSR     PC,PRTLPR       ;PRINT CONTENTS OF THE LPR REGISTER.
2855  013542                          ENDMSG
2856  013542                                                          L10011:
2857  013542  104423                                                          TRAP    C$MSG
```

```
                                .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER9101 -
                                ;*******************************************************************************
                                ;*      THIS IS A GENERAL ERROR REPORTING SUBROUTINE WHICH REPORTS A MESSAGE
                                ;*      WHICH TAKES A SINGLE, 2 DIGIT DECIMAL ARGUMENT AFTER THE END OF AN
                                ;*      ASCII MESSAGE.
                                ;*
                                ;* INPUTS:      R1 - VALUE TO BE PRINTED AFTER MSG AS 2 DECIMAL DIGITS.
                                ;*              R2 - ADDRESS OF MESSAGE TO PRINT FIRST.
                                ;*
                                ;* OUTPUTS:     A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
                                ;*
                                ;* CALLING SEQUENCE:    INCLUDE THE LABEL "ER9101" AS THE MESSAGE POINTER
                                ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                                ;*
                                ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
                                ;*
                                ;* SUBORDINATE ROUTINES USED: NONE.
                                ;*******************************************************************************

013544                                  BGNMSG  ER9101
013544                                                                  ER9101::
013544                                  PRINTB  #EF9006,R2,R1   ;REPORT THE STRING FOLLOWED BY THE NUMBER.
013544  010146                                                          MOV     R1,-(SP)
013546  010246                                                          MOV     R2,-(SP)
013550  012746  004763                                                  MOV     #EF9006,-(SP)
013554  012746  000003                                                  MOV     #3,-(SP)
013560  010600                                                          MOV     SP,R0
013562  104414                                                          TRAP    C$PNTB
013564  062706  000010                                                  ADD     #10,SP

013570                                  ENDMSG
013570                                                          L10012:
013570  104423                                                          TRAP    C$MSG
```

```
                                .SBTTL  GLOBAL ERROR REPORTING ROUTINE          - ER9301 -
                                ;*********************************************************************************
                                ;*      THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ANY BMP CODES
                                ;*      THAT ARE FOUND IN THE BMP CODE QUEUE, TOGETHER WITH THE THE NUMBER OF
                                ;*      THE TEST THAT WAS EXECUTING AT THE TIME THE BMP CODE WAS LOGGED.
                                ;*
                                ;* INPUTS:      R1 - THE ADDRESS OF THE FIRST MESSAGE TO BE REPORTED.
                                ;*              R2 - THE ADDRESS OF THE NEXT EMPTY CELL IN THE QUEUE.
                                ;*
                                ;* OUTPUTS:     THE TEST NUMBER FOLLOWED BY THE BMP CODE ARE PRINTED AT THE
                                ;*              OPERATOR CONSOLE.
                                ;*
                                ;* CALLING SEQUENCE:    INCLUDE THE LABEL "ER9301" AS THE MESSAGE POINTER
                                ;*                      PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
                                ;*
                                ;* COMMENTS:    THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
                                ;*
                                ;* SUBORDINATE ROUTINES USED: NONE.
                                ;*********************************************************************************

                                        BGNMSG  ER9301
                                                                                ER9301::
                                        SAVE                    ;SAVE THE GPRS ON THE STACK.
013572  004537  004062                          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

                                        PRINTB  #EF0503,R1      ;REPORT UNEXPECTED BMP CODES FOUND.
013576                                                                  MOV     R1,-(SP)
013576  010146                                                          MOV     #EF0503,-(SP)
013600  012746  004213                                                  MOV     #2,-(SP)
013604  012746  000002                                                  MOV     SP,R0
013610  010600                                                          TRAP    C$PNTB
013612  104414                                                          ADD     #6,SP
013614  062706  000006
013620  012703  002452                  MOV     #BMPCQB,R3      ;GET THE START ADDRESS OF THE BMP CODE QUEUE.
013624  012705  012150                  MOV     #EM9302,R5      ;GET THE MESSAGE TO BE REPORTED.
013630  012301          2$:             MOV     (R3)+,R1        ;GET THE NUMBER OF THE TEST THAT WAS EXECUTING.
013632  012304                          MOV     (R3)+,R4        ;GET BMP CODE THAT WAS REPORTED OFF THE QUEUE.
013634  004737  013716                  JSR     PC,50$          ;GO REPORT THE BMP CODE.
013640  020302                          CMP     R3,R2           ;CHECK IF ALL CODES HAVE BEEN REPORTED.
013642  103772                          BLO     2$              ;IF IT IS NOT THE LAST BMP CODE THEN LOOP.
                                ;+
                                ; CHECK IF OVERFLOW HAS OCCURRED.
                                ;  THE CONDITIONS FOR OVERFLOW ARE: THE POINTER CONTAINS THE ADDRESS OF THE
                                ;  LAST CELL IN THE QUEUE, AND A BMP CODE HAS ALREADY BEEN WRITTEN INTO THAT
                                ;  CELL.
                                ;-
013644  020227  002646                  CMP     R2,#BMPCQE-4    ;CHECK IF THE POINTER IS AT THE LAST LOCATION.
013650  001036                          BNE     60$             ;EXIT IF NOT AT THE LAST LOCATION.
013652  005762  000002                  TST     2(R2)           ;CHECK FOR A BMP CODE IN THE LAST CELL
013656  001433                          BEQ     60$             ;EXIT IF NO OVERFLOW HAS OCCURED, CELL EMPTY.
013660  012301                          MOV     (R3)+,R1        ;GET THE TEST NUMBER OFF THE QUEUE.
013662  011304                          MOV     (R3),R4         ;GET THE BMP CODE OFF THE QUEUE.
013664  012705  012200                  MOV     #EM9303,R5      ;SELECT THE MESSAGE TO BE REPORTED.
013670                                  PRINTX  #EF9302         ;REPORT OVERFLOW CONDITION.
013670  012746  005077                                                  MOV     #EF9302,-(SP)
013674  012746  000001                                                  MOV     #1,-(SP)
013700  010600                                                          MOV     SP,R0
```

```
2948  013702  104415                                                                  TRAP    C$PNTX
2949  013704  062706  000004                                                          ADD     #4,SP
2950  013710  004737  013716          JSR     PC,50$          ;REPORT THE LAST BMP CODE PLACED ON THE QUEUE.
2951  013714  000414                  BR      60$             ;EXIT.
2952
2953  013716                  50$:    PRINTX  #EF9301,R5,R1,R4  ;PRINT THE MESSAGE.
2954  013716  010446                                                                  MOV     R4,-(SP)
2955  013720  010146                                                                  MOV     R1,-(SP)
2956  013722  010546                                                                  MOV     R5,-(SP)
2957  013724  012746  005021                                                          MOV     #EF9301,-(SP)
2958  013730  012746  000004                                                          MOV     #4,-(SP)
2959  013734  010600                                                                  MOV     SP,R0
2960  013736  104415                                                                  TRAP    C$PNTX
2961  013740  062706  000012                                                          ADD     #12,SP
2962  013744  000207                  RTS     PC              ;RETURN.
2963  013746                  60$:    PASS                    ;RESTORE THE GPR CONTENTS.
2964  013746  004736                                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
2965
2966  013750                  ENDMSG
2967  013750                                                                  L10013:
2968  013750  104423                                                                  TRAP    C$MSG
```

```
                        .SBTTL  GLOBAL SUBROUTINES SECTION


                        ;++
                        ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                        ; THAT ARE USED IN MORE THAN ONE TEST.
                        ;--
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - ALTFLD -
                        ;++ ************************************************************************
                        ;*              - ALTER DEVICE REGISTER FIELDS ROUTINE -
                        ;*      THIS SUBROUTINE ALTERS THE SPECIFIED FIELD OF THE SPECIFIED DEVICE
                        ;*      REGISTER FOR THE SPECIFIED LINES.  THIS ROUTINE CAN BE USED TO SET
                        ;*      OR CLEAR BITS WITHIN SELECTED FIELDS OF SELECTED REGISTERS.
                        ;*      USE EXAMPLES:   SET RX.BAUD.RATE FIELDS ON LINES 3 AND 6.
                        ;*                      CLEAR TX.DMA BITS ON ALL LINES.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF THE REGISTERS TO ALTER.
                        ;*              R2 - BIT FIELDS SET TO DESIRED STATES.
                        ;*              R3 - BIT MAP OF LINES FOR WHICH TO ALTER REGISTER.
                        ;*              R4 - MASK OF BITS TO ALTER (1 INDICATES CHANGE BIT).
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DEVICE CSR.
                        ;*              IESTAT - SAVED STATES OF THE INTERRUPT ENABLE BITS.
                        ;*
                        ;* OUTPUTS:     DEVICE REGISTERS - SPECIFIED REGISTER FIELDS ALTERED.
                        ;*                  CSR IND.ADR.REG FIELD - DESTROYED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,ALTFLD
                        ;*
                        ;* COMMENTS:    THIS ROUTINE READS THE SPECIFIED REGISTERS FOR ALL LINES
                        ;*              WITH NUMBERS LOWER THAN THE HIGHES SPECIFIED LINE.
                        ;*              THIS ROUTINE DOES NOT READ THE CSR.
                        ;*
                        ;* SUBROUTINES CALLED: NONE.
                        ;-- ************************************************************************

013752                  ALTFLD:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
013752  004537  004062                 JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

                        ;+
                        ; SET UP TO LOOP FOR EACH LINE:
                        ;    PREPARE THE WORD TO BE ORED INTO THE REGISTER CONTENTS.
                        ;    SET UP THE WORD TO WRITE INTO THE IND.ADR.REG FIELD OF THE CSR.
                        ;-
013756  010400                  MOV     R4,R0           ;CALCULATE THE NEW CONTENTS OF THE
013760  005100                  COM     R0              ;  REGISTER FIELDS WHICH ARE TO BE
013762  040002                  BIC     R0,R2           ;  ALTERRED BY THIS ROUTINE.
013764  013705  002274          MOV     IESTAT,R5       ;SET UP TO WRITE IND.ADR.REG FIELD TO 0.
                        ;+
                        ; LOOP ONCE FOR EACH LINE, ALTERING THE SPECIFIED FIELD IN THE SPECIFIED
                        ;    REGISTER IF THE LINE HAS BEEN SELECTED FOR ALTERING.
                        ; EXIT THE LOOP IF NO MORE LINES TO ALTER, OR IF WE HAVE ALTERED THE MAX
                        ;    ALLOWABLE NUMBER OF LINES (AS SPECIFIED BY NUMLNS).
                        ;-
013770  000241                  CLC                     ;PREPARE FOR ROTATE, "TST R5" DOES THIS BELOW.
013772  006003          2$:     ROR     R3              ;GET THE LINE SELECT BIT FOR THIS LINE.
013774  103006                  BCC     4$              ;SKIP SETUP IF LINE IS NOT SELECTED.
013776  010577  166244          MOV     R5,@CSRA        ;SET DUT CSR IND.ADR.REG FIELD TO THIS LINE.
014002  011100                  MOV     (R1),R0         ;GET THE PRESENT CONTENTS OF THE REG TO ALTER.
014004  040400                  BIC     R4,R0           ;CLEAR THE BIT FIELDS WE ARE TO ALTER.
014006  050200                  BIS     R2,R0           ;OR IN THE NEW STATES OF THE FIELDS.
014010  010011                  MOV     R0,(R1)         ;WRITE THE NEW REGISTER CONTENTS TO THE REG.
014012  005205          4$:     INC     R5              ;SET LINE NUMBER TO THE NEXT LINE.
014014  005703                  TST     R3              ;CHECK FOR UNHANDLED LINES, CLEAR CARRY FLAG.
```

```
3032  014016  001365                    BNE     2$              ;LOOP IF SELECTED LINE(S) IS NOT HANDLED.
3033
3034  014020                    60$:    PASS                    ;RESTORE GPRS.
3035  014020  004736                            JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
3036  014022  000207                    RTS     PC              ;RETURN TO CALLING ROUTNE.
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - ASLNTL -
                        ;++ ************************************************************************
                        ;*              - SETUP ASSOCIATED LINE NUMBER TABLES ROUTINE -
                        ;*      THIS ROUTINE SETS UP THE TWO TABLES WHICH ARE CONTAIN INFORMATION
                        ;*      ABOUT THE TX/RX LINE WHICH IS ASSOCIATED WITH A PARTICULAR RX/TX
                        ;*      LINE.  ONE TABLE IS A TABLE OF WORDS WHICH CONTAINS WORD OFFSET
                        ;*      VALUES AND THE OTHER TABLE IS A TABLE OF BYTES WHICH CONTAINS
                        ;*      LINE NUMBER VALUES.
                        ;*
                        ;* INPUTS:      LOPBCK - STORAGE FOR THE TYPE OF LOOPBACK ON THE DUT.
                        ;*              NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
                        ;*              STGTRB - LABEL AT BASE OF STAGGERED LINE ASSOCIATION TBL.
                        ;*              TXRLNB - LABEL AT BASE OF BYTE TX/RX LINE NUMBER TABLE.
                        ;*              TXRXLB - LABEL AT BASE OF WORD TX/RX LINE NUMBER TABLE.
                        ;*              TXRXLE - LABEL AT END OF WORD TX/RX LINE NUMBER TABLE.
                        ;*
                        ;* OUTPUTS:     TXRXL, TXRLN - TABLES INITIALIZED FOR SELECTED LOOPBACK.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,ASLNTL
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

014024                  ASLNTL:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
014024  004537  004062                 JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014030  123727  002242  000002          CMPB    LOPBCK,#2       ;TEST FOR STAGGERED LOOPBACK.
014036  001411                        BEQ     4$              ;GO SET UP STAGGERED TABLE IF STAGGERED LPBCK.
                        ;+
                        ; SET UP THE WORD TABLE FOR NON-STAGGERED LOOPBACK.
                        ;-
014040  005005                          CLR     R5              ;CLEAR THE LINE COUNTER
014042  010565  003752  2$:             MOV     R5,TXRXLB(R5)   ;SET UP A WORD OF THE TABLE.
014046  005205                          INC     R5
014050  005205                          INC     R5              ;SET LINE COUNTER TO NEXT LINE OFFSET.
014052  020527  000020                  CMP     R5,#2*NUMLNS    ;TEST FOR ALL LINES DONE.
014056  002771                        BLT     2$              ;LOOP UNTIL ALL LINES DONE.
014060  000411                        BR      8$              ;GO SET UP THE BYTE TABLE.
                        ;+
                        ; SET UP THE WORD TABLE FOR STAGGERED LOOPBACK.
                        ;-
014062  012701  004032  4$:             MOV     #STGTRB,R1      ;SET UP THE SOURCE POINTER.
014066  012702  003752                  MOV     #TXRXLB,R2      ;SET UP THE DESTINATION POINTER.
014072  112122          6$:             MOVB    (R1)+,(R2)+     ;MOVE A BYTE INTO THE TABLE.
014074  105022                          CLRB    (R2)+           ;CLEAR THE UPPER BYTE OF WORD TABLE ENTRY.
014076  020227  004012                  CMP     R2,#TXRXLE      ;COMPARE POINTER WITH END ADR OF TABLE.
014102  002773                        BLT     6$              ;LOOP IF NOT AT END YET.
                        ;+
                        ; SET UP THE BYTE TABLE BASED ON THE WORD ASSOCIATION TABLE.
                        ;-
014104  012701  003752  8$:             MOV     #TXRXLB,R1      ;SET UP THE SOURCE POINTER.
014110  012702  004012                  MOV     #TXRLNB,R2      ;SET UP THE DESTINATION POINTER.
014114  012103          10$:            MOV     (R1)+,R3        ;GET THE WORD OFFSET VALUE FROM WORD TABLE.
014116  006203                          ASR     R3              ;DIVIDE BY 2 TO GET LINE NUMBER VALUE.
014120  110322                          MOVB    R3,(R2)+        ;LOAD THE BYTE LINE NUMBER INTO TABLE.
```

```
3093  014122  020127  004012                  CMP     R1,#TXRXLE      ;COMPARE SOURCE POINTER WITH ADR OF TABLE END.
3094  014126  002772                        BLT     10$             ;LOOP IF NOT AT END OF TABLE YET.
3095
3096  014130                          60$:    PASS                    ;RESTORE GPRS.
3097  014130  004736                                JSR     PC,@(SP)+               ;RETURN TO PREGO5 SUBRT.
3098  014132  000207                        RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - CALMSL -
                        ;++ ************************************************************************
                        ;*              - CALIBRATE MILLI SECOND LOOP COUNT SUBROUTINE -
                        ;*      THIS SUBROUTINE CALIBRATES THE TIMING LOOP WHICH IS USED IN THE MSLOOP
                        ;*      ROUTINE.  THIS SUBROUTINE CALCULATES A VALUE FOR THE MSLCNT VARIABLE
                        ;*      WHICH IS THE NUMBER OF SOFTWARE LOOPS WHICH TAKES 1 MS TO EXECUTE IN
                        ;*      THE MSLOOP ROUTINE.  THIS ROUTINE CALIBRATES THE COUNT BY USING THE
                        ;*      LINE TIME CLOCK (LTC), SO IF NO LTC IS AVAILABLE THE DEFAULT VALUE FOR
                        ;*      THE DELAY COUNT MUST BE USED.
                        ;*
                        ;* INPUTS:      MSLCNT - DEFAULT 1 MS DELAY LOOP COUNT VALUE, OR
                        ;*                       VALUE FROM PREVIOUS CALIBRATION.
                        ;*              MSTICK - NUMBER OF MS PER LTC CLOCK TICK.
                        ;*              TIMER1 - TIMER COUNTER CHANGED BY LTC INTERRUPT SERVICE RTN.
                        ;*              CLKHRZ - NUMBER OF LTC CLICKS PER SECOND (50 OR 60).
                        ;*
                        ;* OUTPUTS:     CARRY - SET IF LTC IS AVAILABLE, AND NEW CALIBRATION PERFORMED.
                        ;*              MSLCNT - NEW 1 MS DELAY LOOP COUNT VALUE IF LTC AVAILABLE, OR
                        ;*                       UNCHANGED IF NO LTC IS AVAILABLE.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,CALMSL
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: UNSDIV,OOPS.
                        ;-- ************************************************************************

014134                  CALMSL:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
014134  004537  004062                      JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014140  005037  014354          CLR     62$                     ;CLEAR THE 2ND TIME FLAG.
                        ;+
                        ; SYNCHRONIZE WITH THE LTC.
                        ;-
014144  012705  000001  2$:     MOV     #1,R5                   ;SET OUTER LOOP COUNTER TO 1 LOOP.
                                                        ;INCREASE THE VALUE LOADED INTO THIS COUNTER IF THE <**
                                                        ;FOLLOWING LOOP FAILS ON FUTURE, FASTER PROCESSORS. <**
014150  005000          CLR     R0                      ;CLEAR THE WAIT FOR CLOCK INT COUNTER.
014152  012737  000001  002332  MOV     #1,TIMER1               ;SET UP COUNT OF 1 TO SYNCH WITH LTC.
014160  005737  002332  4$:     TST     TIMER1                  ;CHECK FOR COUNTER HAVING GONE TO ZERO.
014164  001410          BEQ     6$                      ;JUMP OUT OF LOOP IF LTC HAS INTERRUPTED.
014166  005200          INC     R0                      ;COUNT THIS ITERATION OF THE INNER LOOP.
014170  001373          BNE     4$                      ;LOOP IF COUNTER HAS NOT TURNED OVER.
014172  005305          DEC     R5                      ;DECREMENT THE INNER LOOP COUNTER.
014174  003371          BGT     4$                      ;LOOP IF OUTER LOOP COUNT NOT UP.
                        ;+
                        ; IF WE GOT NO LTC INTERRUPT, INDICATE THAT THERE IS NO LTC AVAILABLE.
                        ; LTC MUST BE FLAKEY, OR NOT REALLY AN LTC AT ALL.
                        ;-
014176  005037  002330          CLR     CLKHRZ                  ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
014202  000241          CLC                             ;INDICATE FAILURE FOR RETURN.
014204  000461          BR      60$                     ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
                        ;+
                        ; WE ARE NOW SYNCHRONIZED WITH THE LTC.
                        ; SET UP FOR THE CALIBRATION LOOP.
                        ;-
```

```
3155  014206  012704  002332          6$:     MOV     #TIMER1,R4      ;WILL TEST TIMER1 IN THE LOOP BELOW.
3156  014212  005001                          CLR     R1              ;CLEAR THE OUTER LOOP COUNTER.
3157  014214  005002                          CLR     R2              ;INDICATE TO CHECK ALL BITS OF TIMER1.
3158  014216  005003                          CLR     R3              ;INDICATE TO CHECK FOR TIMER1 CLEAR.
3159  014220  012714  000001                  MOV     #1,(R4)         ;LOAD TIMER1 WITH COUNT OF 1.
3160
3161  014224  013705  002344          8$:     MOV     MSLCNT,R5       ;LOAD MS LOOP COUNT.
3162  014230  011400                  10$:    MOV     (R4),R0         ;GET THE TIMER1 VALUE.
3163  014232  010037  014356                  MOV     R0,64$          ;SAVE WORD (LIKE IN THE REAL LOOP).
3164  014236  040200                          BIC     R2,R0           ;LEAVE ALL THE BITS.
3165  014240  020003                          CMP     R0,R3           ;COMPARE AGAINST ZERO.
3166  014242  000261                          SEC                     ;SET CARRY IN CASE OF SUCCESS.
3167  014244  001406                          BEQ     12$             ;EXIT LOOP IF TIMER1 HAS CLEARED.
3168  014246  005305                          DEC     R5              ;COUNT DOWN THE INSIDE MS LOOP COUNT.
3169  014250  001367                          BNE     10$             ;LOOP IF MS NOT UP.
3170  014252  005301                          DEC     R1              ;DECREMENT THE MS TIME COUNT.
3171  014254  001363                          BNE     8$              ;KEEP LOOPING.
3172  014256  004737  015346                  JSR     PC,OOPS         ;WE OVERFLOWED, SOMETHING IS WRONG, ABORT.
3173                                  ;+
3174                                  ; WE HAVE NOW HAVE LOOP COUNT INFORMATION FOR ONE CLOCK TICK.
3175                                  ; WE HAVE NEGATIVE OF NUMBER OF OUTER LOOPS IN R1, EACH IS MSLCNT INNER LOOPS.
3176                                  ; WE HAVE THE PORTION OF THE LAST OUTER LOOP NOT EXECUTED, IN R5.
3177                                  ; NOW WE CALCULATE THE TOTAL NUMBER OF INNER LOOPS EXECUTED.
3178                                  ;-
3179  014262  005401                  12$:    NEG     R1              ;GET NUMBER OF OUTER LOOPS.
3180  014264  013702  002344                  MOV     MSLCNT,R2       ;GET THE NUMBER OF INNER LOOPS PER OUTER LOOP.
3181  014270  010203                          MOV     R2,R3           ;COPY NUMBER OF LOOPS FOR MULTIPLY.
3182  014272  160502                          SUB     R5,R2           ;CALC # OF INNER LOOPS DONE IN LAST OUTER LOOP
3183  014274  010204                          MOV     R2,R4           ;  AND ADD TO ACCUMULATOR LSWORD.
3184  014276  005005                          CLR     R5              ;CLEAR ACCUMULATOR MSWORD.
3185  014300  005301                  14$:    DEC     R1              ;CHECK R1 FOR 0 CONDITION
3186  014302  100403                          BMI     16$             ; SKIP MULTIPLICATION IF ZERO
3187  014304  060304                          ADD     R3,R4           ;MULTIPLY NUMBER OF INNER
3188  014306  005505                          ADC     R5              ; LOOPS PER OUTER LOOP BY
3189  014310  000773                          BR      14$             ;NUMBER OF OUTER LOOPS PERFORMED.
3190                                  ;+
3191                                  ; DIVIDE THE TOTAL NUMBER OF INNER LOOPS BY THE NUMBER OF MS PER LTC TICK.
3192                                  ;-
3193  014312  013701  002342          16$:    MOV     MSTICK,R1       ;# OF MS PER LTC TICK IS DIVISOR.
3194  014316  010403                          MOV     R4,R3           ;LSWORD OF LOOP COUNT IS LSWORD OF DIVIDEND.
3195  014320  010502                          MOV     R5,R2           ;MSWORD OF LOOP COUNT IS MSWORD OF DIVIDEND.
3196  014322  004737  017066                  JSR     PC,UNSDIV       ;DIVIDE NUMBER OF LOOPS BY MS PER LTC TICK.
3197  014326  103402                          BCS     18$             ;BYPASS OOPS IF WE'RE OK.
3198  014330  004737  015346                  JSR     PC,OOPS         ;CLOCK ROUTINES ARE NOT LONG ENOUGH, OR BUG.
3199  014334  010137  002344          18$:    MOV     R1,MSLCNT       ;SET NEW VALUE FOR MS LOOP COUNT.
3200  014340  005137  014354                  COM     62$             ;SET THE 2ND ITERATION FLAGS IF 1ST ITERATION.
3201  014344  001277                          BNE     2$              ;BRANCH IF ONLY ONE ITERATION DONE.
3202  014346  000261                          SEC                     ;SET THE SUCCESS FLAG FOR EXIT.
3203
3204  014350                          60$:    PASS                    ;RESTORE GPRS,
3205  014350  004736                                  JSR     PC,@(SP)+       ;RETURN TO PREG05 SUBRT.
3206  014352  000207                          RTS     PC              ; CARRY - SUCCESS FLAG. SET IF SUCCESS.
3207
3208  014354  000000          62$:    .WORD   0                       ;2ND CALIBRATION ITERATION FLAGS.
3209  014356  000000          64$:    .WORD   0                       ;DUMMY WORD FOR STORAGE OF THE READ WORD.
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - CHKBMP -
                                ;++ ************************************************************************
                                ;*                        - CHECK IF CHARACTER IS A BMP CODE -
                                ;*      THIS SUBROUTINE IS USED TO CHECK FOR BMP CODES.
                                ;*      IF A BMP CODE IS DETECTED, IT WILL BE SAVED ON THE QUEUE TO BE REPORTED
                                ;*      LATER.  THE CARRY IS USED AS A FLAG TO INDICATE A CODE HAS BEEN FOUND.
                                ;*
                                ;* INPUTS:      R2 - CONTAINS THE DATA TO BE CHECKED.
                                ;*
                                ;* OUTPUTS:     R1 - CONTIANS THE MESSAGE TO BE REPORTED.
                                ;*              ERRBLK - CONTAINS THE ERROR REPORTING ROUTINE.
                                ;*              CARRY BIT IS USED TO INDICATE A BMP CODE FOUND, CARRY SET.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,CHKBMP
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: SAVBMP.
                                ;-- ************************************************************************

014360                          CHKBMP:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
014360  004537  004062                         JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014364  012700  170301                 MOV     #170301,R0              ;SET UP THE FLAGS OF A BMP CODE.
014370  040200                         BIC     R2,R0                   ;TRY TO CLEAR THE BMP CODE FLAGS.
014372  001011                        BNE     2$                      ;IF NOT A BMP CODE, EXIT WITH FAILURE.
014374  004737  016220                JSR     PC,SAVBMP               ;SAVE THE BMP CODE ON THE QUEUE.
014400  012701  010555                 MOV     #EM5303,R1              ;PASS THE MESSAGE TO BE REPORTED.
014404  012737  012732  004060         MOV     #ER1603,ERRBLK          ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
014412  000261                         SEC                             ;PASS FLAG TO INDICATE SUCCESS, BMP CODE FOUND.
014414  000401                        BR      60$                     ;EXIT.
014416  000241                2$:      CLC                             ;PASS FLAG TO INDICATE FAILURE.
014420                          60$:    PASS    R1                      ;RESTORE GPRS, EXCEPT
014420  010166  000004                                  MOV     R1,R1SLOT(SP)           ;PUT R1 IN STACK SLOT.
014424  004736                                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                                        ;R1 - CONTAINS THE ADDRESS OF ERROR MESSAGE.
                                                                        ;CARRY BIT - SET INDICATES SUCCESS.
014426  000207                        RTS     PC
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - CKTRAP -
                                ;*************************************************************************
                                ;*      CHECK TRAP ROUTINE -
                                ;*      THIS SUBROUTINE IS USED TO CHECK FOR A BUS TIME-OUT TRAP (004 TRAP)
                                ;*      WHICH IS CAUSED BY AN ACCESS TO A NON-EXISTENT MEMORY OR I/O LOCATION.
                                ;*      IF THE TRAP DOES NOT OCCUR, THIS ROUTINE RETURNS A SUCCESS INDICATION.
                                ;*
                                ;* INPUTS:      R0 - SOURCE ADDRESS FOR MOVE.
                                ;*              R1 - DESTINATION ADDRESS FOR MOVE.
                                ;*              (R0) - SOURCE FOR THE MOVE.
                                ;*
                                ;* OUTPUTS:     (R1) - WRITTEN TO THE CONTENTS OF (R0).
                                ;*              CARRY FLAG - SET ON RETURN IF NO 004 TRAP DETECTED.
                                ;*              TP4FLG - NONZERO IF TRAP OCCURRED, CLEARED OTHERWISE.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,CKTRAP
                                ;*
                                ;* COMMENTS:    IF THIS SUBROUTINE CAUSES A TRAP, EITHER THE ADDRESS WHICH
                                ;*              IS LABELED ADRPTR WILL BE THE TRAP PC ADDRESS ON THE STACK.
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE.
                                ;*************************************************************************

014430                          CKTRAP:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
014430  004537  004062                        JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014434  005037  002316                CLR     TP4FLG          ;CLEAR THE 004 TRAP FLAGS.
014440  011011                        MOV     (R0),(R1)       ;PERFORM THE MOVE IN QUESTION.
014442  005737  002316          ADRPTR:: TST  TP4FLG          ;CHECK FOR OCCURENCE OF TRAP.
014446  000261                        SEC                     ;INDICATE SUCCESS.
014450  001401                        BEQ     60$             ;EXIT WITH SUCCESS IF TRAP DID NOT OCCUR.
014452  000241                        CLC                     ;INDICATE FAILURE.
014454                          60$:    PASS                    ;RESTORE GPRS.
014454  004736                                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
014456  000207                        RTS     PC
```

```
3281                            .SBTTL  GLOBAL SUBROUTINE                - CLNRST -
3282                            ;*************************************************************************
3283                            ;*              - CLEAN RESET OF THE DEVICE UNDER TEST -
3284                            ;*      THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
3285                            ;*      THE DUT'S SELF-TEST IS SKIPPED,AND THE FIFO IS PURGED OF ANY ERROR
3286                            ;*      CODES, ETC.
3287                            ;*      IF THE RESET DOES NOT SUCCESFULLY COMPLETE, THEN THE CARRY BIT IS
3288                            ;*      PASSED BACK TO THE CALLING ROUTINE (CLEAR).
3289                            ;*
3290                            ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR
3291                            ;*              TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
3292                            ;*              ERRNBR - ERROR NUMBER FOR POSSIBLE ERROR REPORT.
3293                            ;*              ERRTBL- ERRTYP,ERNBR,AND ERRMSG SET UP CORRECTLY.
3294                            ;*
3295                            ;* OUTPUTS:     THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
3296                            ;*              CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
3297                            ;*              ERRBLK - VALUE MAY BE DESTROYED.
3298                            ;*              IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
3299                            ;*              TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.
3300                            ;*
3301                            ;* CALLING SEQUENCE:    JSR     PC,CLNRST
3302                            ;*
3303                            ;* COMMENTS:    THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS ERRNBR.
3304                            ;*              THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.
3305                            ;*
3306                            ;* SUBORDINATE ROUTINES CALLED: DELAY,MSLGET,PUFIFO,RESETT.
3307                            ;*************************************************************************
3308
3309  014460                    CLNRST:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
3310  014460  004537  004062                    JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
3311                            ;+
3312                            ; RESET THE DUT.
3313                            ;  THIS ROUTINE REPORTS ERRORS WITH NUMBERS FROM ERRNBR THRU ERRNBR+2.
3314                            ;-
3315  014464  004737  016022            JSR     PC,RESETT               ;RESET THE DUT TO A KNOWN STATE.
3316  014470  103002            BCC     60$                             ;EXIT ROUTINE WITH ABORT TEST INDICATOR.
3317                            ;+
3318                            ; PURGE THE FIFO OF ERROR CODES, SAVE ANY BMP CODES FOUND.
3319                            ;-
3320  014472  004737  015656            JSR     PC,PUFIFO               ;PURGE THE FIFO.
3321
3322  014476                    60$:                                    ;EXIT THE TEST USING RESETT OR PUFIFO STATUS.
3323  014476                            PASS                            ;RESTORE GPRS, PASS THE FOLLOWING INTACT:
3324  014476  004736                            JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
3325                                                                    ;CARRY BIT:IF CLEAR, THEN ABORT THE TEST.
3326  014500  000207            RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - CMPMST -
                        ;++ ************************************************************************
                        ;*                      - COMPARE MODEM STATUS ROUTINE -
                        ;*      THIS ROUTINE IS USED TO COMPARE THE PRESENT MODEM STATUS AGAINST THE
                        ;*      MODEM STATUS WHICH IS STORED IN THE MODEM STATUS STORAGE TABLE.  IT
                        ;*      IGNORES THE STATES OF THE SPECIFIED SIGNALS ON A SPECIFIED LINE.
                        ;*
                        ;* INPUTS:      R1 - LINE NUMBER OF SPECIFIED LINE.
                        ;*              R2 - BIT MAP OF BITS TO IGNORE ON SPECIFIED LINE.
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
                        ;*              NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
                        ;*              STATA - CONTAINS THE ADDRESS OF THE DUT STAT REGISTER.
                        ;*              STSTB - LABEL AT BASE OF STAT STORAGE TABLE.
                        ;*              TXRLNB - LABEL AT BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
                        ;*
                        ;* OUTPUTS:     CARRY - SUCCESS FLAG (SET IF NO DISCREPANCIES WERE FOUND).
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,CMPMST
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

014502                  CMPMST:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
014502  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014506  005003                  CLR     R3              ;CLEAR THE LINE COUNTER.
014510  012704  002652          MOV     #STSTB,R4       ;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
014514  010377  165526  2$:     MOV     R3,@CSRA        ;SET UP THE CSR IND.ADR.REG FIELD.
014520  017700  165530          MOV     @STATA,R0       ;GET THE CONTENTS OF THIS LINE'S STAT REGISTER.
014524  011405                  MOV     (R4),R5         ;GET THE PREVIOUS CONTENTS FROM STORAGE.
014526  040005                  BIC     R0,R5
014530  042400                  BIC     (R4)+,R0
014532  050005                  BIS     R0,R5           ;XOR PRESENT AND STORED STAT VALUES.
014534  012700  043777          MOV     #43777,R0       ;PREPARE TO MASK OUT UNUSED BITS.
014540  120301                  CMPB    R3,R1           ;TEST FOR THIS BEING SPECIFIED LINE.
014542  001001                BNE     10$             ;DON'T MASK OUT SPECIFIED BITS IF IT IS NOT.
014544  050200                  BIS     R2,R0           ;MASK OUT SPECIFIED BITS.
014546  040005          10$:    BIC     R0,R5           ;GET BIT MAP OF UNDESIRED CHANGES.
014550  001006                BNE     50$             ;EXIT WITH FAILURE IF CHANGES OCCURRED.
014552  005203                  INC     R3              ;SELECT NEXT LINE.
014554  020327  000010          CMP     R3,#NUMLNS      ;ALL LINES DONE?
014560  002755                BLT     2$              ;LOOP IF NOT ALL LINES DONE.
014562  000261                  SEC                     ;INDICATE SUCCESS.
014564  000401                BR      60$             ;EXIT THIS ROUTINE WITH SUCCESS.

014566  000241          50$:    CLC                     ;INDICATE FAILURE.

014570                  60$:    PASS                    ;RESTORE GPRS.
014570  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
014572  000207                  RTS     PC              ;  CARRY - SUCCESS FLAG (SET IF SUCCESS).
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - DELAY -
                        ;*****************************************************************************
                        ;*                                - DELAY SUBROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO DELAY A VARIABLE NUMBER OF MILLI-SECONDS.
                        ;*
                        ;* INPUTS:      R4 - CONTAINS THE NUMBER OF MS TO DELAY.
                        ;*               MSLCNT.
                        ;*
                        ;* OUTPUTS:     NONE.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,DELAY
                        ;*
                        ;* COMMENTS:    IF NO HARDWARE CLOCK INTERRUPTS ARE OCCURING, CONTROL-CS WILL
                        ;*              NOT BE HONORED FOR THE DURATION OF THE DELAY.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;*****************************************************************************

014574                  DELAY:: SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
014574  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014600  010401          MOV     R4,R1                   ;PASS NUMBER OF MS DELAY AS TIME-OUT VALUE.
014602  012702  177777  MOV     #-1,R2                  ;TELL MSLOOP ROUTINE TO CHECK ALL BITS.
014606  005003          CLR     R3                      ;TELL MSLOOP RTN TO CHECK FOR ALL BITS CLEAR.
014610  012704  014632  MOV     #62$,R4                 ;TELL MSLOOP TO CHECK DUMMY NON-ZERO WORD.
014614  004737  015332  JSR     PC,MSLOOP               ;DELAY THE REQUESTED # OF MS.
014620  103002          BCC     60$                     ;EXIT ROUTINE IF WE TIMED-OUT.]
014622  004737  015346  JSR     PC,OOPS                 ;IF NO TIME-OUT, BAD PROGRAM OR HOST MACHINE.
014626                  60$:    PASS                    ;RESTORE GPRS.
014626  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
014630  000207          RTS     PC

014632  177777          62$:    .WORD   -1              ;DUMMY, NON-ZERO WORD.
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - DODMA -
                        ;++ ************************************************************************
                        ;*                        - INITIATE DMA TRANSMISSION ROUTINE -
                        ;*      THIS ROUTINE WRITES THE DMA PARAMETER TO THE SPECIFIED DEVICE AND
                        ;*      INITIATES THE DMA TRANSMISSION.
                        ;*
                        ;* INPUTS:       R1 - LINE NUMBER ON WHICH TO INITIATE THE DMA.
                        ;*               R2 - START ADDRESS OF THE DMA BUFFER (16 BIT VIRTUAL).
                        ;*               R3 - CHARACTER COUNT OF THE DMA BUFFER.
                        ;*               CSRA - CONTAINS ADDRESS OF THE DUT CSR.
                        ;*               IESTAT - STORAGE FOR STATES OF THE INTERRUPT ENABLE BITS.
                        ;*               MMENAB - MEMMORY MANAGEMENT FLAG (0 IF MEM MGT NOT ENABLED).
                        ;*               HOST MEM MGT PAR REGISTERS - IF MEM MGT IS IN USE.
                        ;*               TXAD1A - CONTAINS ADDRESS OF DMA TX BUFFER ADDRESS REG #1.
                        ;*               TXAD2A - CONTAINS ADDRESS OF DMA TX BUFFER ADDRESS REG #2.
                        ;*               TXBFCA - CONTAINS ADDRESS OF DMA CHARACTER COUNT REGISTER.
                        ;*
                        ;* OUTPUTS:      CARRY - SUCCESS FLAG (SET IF DMA_START FOUND CLEAR).
                        ;*               DUT TBUFFAD1 - LS 16 BITS OF DMA BUFFER ADDRESS (INITIALIZED).
                        ;*               DUT TBUFFAD2 - MS 6 BITS OF DMA BUFFER ADDRESS (INITIALIZED),
                        ;*                              DMA START BIT SET.
                        ;*               DUT TBUFFCT - DMA BUFFER CHARACTER COUNT (INITIALIZED).
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,DODMA
                        ;*
                        ;* COMMENTS:     THIS ROUTINE DETERMINES IF MEMORY MANAGEMENT IS BEING USED
                        ;*               AND SETS UP THE FULL 22 BIT PHYSICAL ADDRESS IF NECESSARY.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

014634                  DODMA:: SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
014634  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
014640  012704  000200          MOV     #200,R4         ;PREPARE TO CLEAR UPPER 6 BITS OF DMA BUFF ADR.
014644  005737  002352          TST     MMENAB          ;CHECK FOR MEMORY MANAGEMENT IN USE.
014650  001427                 BEQ     6$              ;GOTO SET UP DEVICE IF MEM MGT NOT IN USE.
                        ;+
                        ; MEMMORY MANAGEMENT IS IN USE.
                        ; CONSTRUCT 22 BIT PHYSICAL ADDRESS FROM THE 16 BIT VIRTUAL ADDRESS.
                        ;-
014652  010205          MOV     R2,R5           ;STRIP THE MOST SIGNIFICANT 3 BITS OF THE
014654  012700  000005          MOV     #5,R0           ;  DMA BUFFER VIRTUAL ADDRESS AND MULTIPLY
014660  006105          2$:     ROL     R5              ;  THEIR VALUE BY TWO TO GET AN OFFSET INTO
014662  005300          DEC     R0              ;  THE TABLE OF MEMMORY MANAGEMENT PAGE
014664  001375         BNE     2$              ;  ADDRESS REGISTERS (PAR).
014666  042705  177761          BIC     #177761,R5      ;
014672  063705  002354          ADD     PAR0A,R5        ;ADD IN THE BASE VALUE OF THE MM PAR REGISTERS.
014676  011505          MOV     (R5),R5         ;GET THE 16 BIT PHYSICAL ADDRESS BLOCK COUNT.
014700  012700  000006          MOV     #6,R0           ;SHIFT UPPER 6 BITS OF THE PHYSICAL ADDRESS
014704  006305          4$:     ASL     R5              ;  BLOCK COUNT (GOTTEN FROM THE PROPER PAR)
014706  006104          ROL     R4              ;  INTO THE LS 6 BITS OF THE WORD TO WRITE
014710  005300          DEC     R0              ;  INTO THE DUT TBUFFAD2 REGISTER.
014712  001374         BNE     4$              ;
014714  042702  160000          BIC     #160000,R2      ;ADD THE 13 BIT DISPLACEMENT FIELD FROM VIRTUAL
014720  060502          ADD     R5,R2           ;  ADR TO THE SHIFTED BLOCK NUMBER FROM THE
014722  005504          ADC     R4              ;  MEMMORY MANAGEMENT PAR.
```

```
3466  014724  052704  000200                  BIS    #200,R4          ;SET THE DMA_START BIT IN WORD FOR TBUFFAD2.
3467                                        ;+
3468                                        ; WRITE THE DMA PARAMETERS OUT TO THE DUT DMA REGISTERS.
3469                                        ;    DISABLE INTERRUPTS.
3470                                        ;    SET UP DUT CSR IND.ADR.REG FIELD.
3471                                        ;    WRITE THE DMA TRANSMIT CHARACTER COUNT.
3472                                        ;    WRITE THE LEAST SIGNIFICANT 16 BITS OF THE DMA BUFFER START ADDRESS.
3473                                        ;    WRITE THE MOST SIGNIFICANT 6 BITS OF THE ADDRESS,
3474                                        ;      SETTING THE DMA_START BIT, AND INITIATING THE DMA TRANSMISSION.
3475                                        ;-
3476  014730                                6$:     GETPRI  R5               ;GET THE PRESENT PROCESSOR PRIORITY.
3477  014730  104440                                                           TRAP    C$GPRI
3478  014732  010005                                                           MOV     R0,R5
3479  014734                                        SETPRI  #PRI07           ;DISABLE ALL HARDWARE INTERRUPTS.
3480  014734  012700  000340                                                  MOV     #PRI07,R0
3481  014740  104441                                                          TRAP    C$SPRI
3482  014742  053701  002274                        BIS     IESTAT,R1        ;PREPARE FOR SETUP OF LINE NUMBER IN DUT CSR.
3483  014746  010177  165274                        MOV     R1,@CSRA         ;SET UP THE DUT CSR IND.ADR.REG FIELD.
3484  014752  105777  165304                        TSTB    @TXAD2A          ;TEST THE DUT DMA_START BIT.
3485  014756  000241                               CLC                      ;INDICATE FAILURE IN CASE DMA.HO BIT IS SET.
3486  014760  100411                              BMI     60$              ;EXIT WITH FAILURE IF DMA.HO BIT IS SET.
3487  014762  010377  165276                        MOV     R3,@TXBFCA       ;WRITE THE DMA CHARACTER COUNT.
3488  014766  010277  165266                        MOV     R2,@TXAD1A       ;WRITE THE LS 16 BITS OF BUFFER ADDRESS.
3489  014772  110477  165264                        MOVB    R4,@TXAD2A       ;WRITE MS 6 BITS OF ADR AND START DMA TX.
3490  014776                                      SETPRI  R5               ;RESTORE THE PROCESSOR PRIORITY.
3491  014776  010500                                                          MOV     R5,R0
3492  015000  104441                                                          TRAP    C$SPRI
3493  015002  000261                               SEC                      ;INDICATE SUCCESS.
3494
3495  015004                                60$:    PASS
3496  015004  004736                                       JSR     PC,@(SP)+        ;RETURN TO PREG05 SUBRT.
3497  015006  000207                               RTS     PC               ;  CARRY - SUCCESS FLAG (SET IF SUCCESS).
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - FINACT -
                        ;++ ************************************************************************
                        ;*                        - FIND FIRST ACTIVE LINE -
                        ;*      THIS SUBROUTINE CALCULATES THE NUMBER OF THE FIRST ACTIVE LINE THAT
                        ;*      IS FOUND IN THE ACTIVE LINE BIT MAP ACTLNS.
                        ;*
                        ;* INPUTS:      ACTLNS - CONTAINS THE ACTIVE LINE BIT MAP.
                        ;*
                        ;* OUTPUTS:     R1 - CONTAINS THE NUMBER OF THE FIRST ACTIVE LINE.
                        ;*              R5 - CONTAINS THE BIT MAP REPRESENTATION OF THE ACTIVE LINE.
                        ;*              CARRY SET INDICATES SUCCESS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,FINACT
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

015010                  FINACT:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
015010  004537  004062                JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                            ;+
                            ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
                            ;-
015014  005001                  CLR     R1                      ;CLEAR THE LINE NUMBER COUNTER.
015016  012703  000010          MOV     #NUMLNS,R3              ;GET MAX LINE NUMBER.
015022  013700  002240          MOV     ACTLNS,R0               ;GET THE ACTIVE LINE BIT MAP.
015026  012705  000001          MOV     #1,R5                   ;SET UP A LINE BIT MASK.
015032  030500          2$:     BIT     R5,R0                   ;LOOK FOR AN ACTIVE LINE.
015034  001006                  BNE     4$                      ;BRANCH TO BEGIN TEST IF A LINE HAS BEEN FOUND.
015036  006305                  ASL     R5                      ;SHIFT THE BIT MASK FOR THE NEXT LINE.
015040  005201                  INC     R1                      ;INCREMENT THE LINE NUMBER COUNTER.
015042  020103                  CMP     R1,R3                   ;CHECK IF ALL LINES HAVE BEEN TRIED.
015044  002772                  BLT     2$                      ;LOOP TO TRY THE NEXT LINE.
015046  000241                  CLC                             ;CLEAR CARRY BIT, NO ACTIVE LINE FOUND.
015050  000401                  BR      60$                     ;EXIT WITH FAILURE.
015052  000261          4$:     SEC                             ;SET CARRY, SUCCESS.

015054                  60$:    PASS    R1,R5                   ;RESTORE GPRS, EXCEPT
015054  010166  000004                          MOV     R1,R1SLOT(SP)           ;PUT R1 IN STACK SLOT.
015060  010566  000014                          MOV     R5,R5SLOT(SP)           ;PUT R5 IN STACK SLOT.
015064  004736                                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                ;R1 - CONTAINS THE NUMBER OF FIRST ACTIVE LINE.
                                                ;R5 - CONTAINS THE BIT MAP OF THE ACTIVE LINE.
                                                ;CARRY - SET INDICATES SUCCESS.
015066  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - INDATP -
                        ;++ ************************************************************************
                        ;*                        - INITIALISE DATA PATTERN -
                        ;*      THIS SUBROUTINE IS USED TO INITIALISE AN INCREMENTAL BYTE DATA PATTERN
                        ;*      IN THE GENERAL BUFFER AREA.
                        ;*      THE DATA PATTERN WILL BE SEQUENTIAL FROM 0 TO 255 (DECIMAL).
                        ;*
                        ;* INPUTS:      BUFBAS - ADDRESS OF THE START OF THE GENERAL BUFFER AREA.
                        ;*              BUFMID - ADDRESS OF THE 255 TH LOCATION.
                        ;*
                        ;* OUTPUTS:     THE FIRST 255 LOCATIONS OF THE GENERAL BUFFER AREA CONTAIN DATA
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,INIDATP
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

015070                  INDATP:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
015070  004537  004062                 JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

015074  012702  002712          MOV     #BUFBAS,R2      ;INITIALIZE THE DATA PATTERN IN THE GENERAL
015100  005003                  CLR     R3              ;  DATA BUFFER TO A 256 BYTE PATTERN.
015102  110322          2$:     MOVB    R3,(R2)+        ;
015104  005203                  INC     R3              ;SELECT THE NEXT CHARACTER.
015106  020227  003312          CMP     R2,#BUFMID      ;CHECK IF WE HAVE 256 DATA PATTERNS.
015112  103773                  BLO     2$              ;

015114                  60$:    PASS                    ;RESTORE GPRS.
015114  004736                          JSR     PC,@(SP)+       ;RETURN TO PREG05 SUBRT.
015116  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - INDTPX -
                        ;++ ************************************************************************
                        ;*              - INITIALISE DATA PATTERN WITHOUT XON OR XOFF -
                        ;*      THIS SUBROUTINE IS USED TO INITIALISE AN INCREMENTAL BYTE DATA PATTERN
                        ;*      IN THE GENERAL BUFFER AREA.
                        ;*      THE DATA PATTERN WILL BE FROM 0 TO 255, BUT WLL EXCLUDE THE FOLLOWING
                        ;*      TWO CHARACTERS; (ASCII DC1, DC3) XON AND XOFF.  THIS WILL CAUSE THE
                        ;*      LAST TWO DATA CHARACTERS TO BE THE SAME AS THE FIRST TWO.
                        ;*
                        ;* INPUTS:      BUFBAS - ADDRESS OF THE START OF THE GENERAL BUFFER AREA.
                        ;*              BUFMID - ADDRESS OF THE 255 TH LOCATION.
                        ;*
                        ;* OUTPUTS:     THE FIRST 255 LOCATIONS OF THE GENERAL BUFFER AREA CONTAIN DATA
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,INDTPX
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

015120                  INDTPX:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
015120  004537  004062                      JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; INITIALIZE THE 256 BYTE DATA PATTERN.
                        ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                        ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                        ;-
015124  012702  002712          MOV     #BUFBAS,R2      ;INITIALIZE THE DATA PATTERN IN THE GENERAL
                              CLR     R3              ;  DATA BUFFER TO A 256 BYTE PATTERN.
015132  110322          2$:     MOVB    R3,(R2)+        ;
015134  105203              INCB    R3              ;SELECT THE NEXT CHARACTER.
015136  122703  000021          CMPB    #21,R3          ;CHECK FOR AN XON CHARACTER.
015142  001001              BNE     4$              ;BRANCH IF CHAR NOT AN XON.
015144  105203              INCB    R3              ;FORCE THE NEXT CHARACTER.
015146  122703  000023  4$:     CMPB    #23,R3          ;CHECK FOR AN XOFF CHARACTER.
015152  001001              BNE     6$              ;BRANCH IF NOT AN XOFF CHARACTER.
015154  105203              INCB    R3              ;FORCE THE NEXT CHARACTER.
015156  020227  003312  6$:     CMP     R2,#BUFMID      ;CHECK IF WE HAVE 256 DATA PATTERNS.
015162  103763              BLO     2$              ;

015164                  60$:    PASS                            ;RESTORE GPRS.
015164  004736                              JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
015166  000207              RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - LINBIT -
                        ;++ ************************************************************************
                        ;*              - LINE NUMBER TO BIT MAP CONVERSION SUBROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO GENERATE A BIT MAP (ONE BIT OF 16 SET)
                        ;*      BASED ON A LINE NUMBER (RANGE: 1 TO 16).  ONLY THE LS 4 BITS OF THE
                        ;*      LINE NUMBER WORD ARE USED, THE OTHERS ARE MASKED OUT (SO UNMASKED
                        ;*      MSBYTES OF DUT CSRS CAN BE PASSED TO THIS ROUTINE WITHOUT ERROR).
                        ;*
                        ;* INPUTS:      R1 - LINE NUMBER (ONLY LS 4 BITS USED, OTHERS DISREGARDED).
                        ;*              BITTBL - BASE LABEL OF A 16 WORD BIT TABLE.
                        ;*
                        ;* OUTPUTS:     R0 - BIT MAP, BIT CORRESPONDING TO LINE NUMBER IS SET:
                        ;*                      IF LINE NUMBER IS 3, THEN BIT3 IS SET, ETC.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,LINBIT
                        ;*
                        ;* COMMENTS:    NO CHECKING IS PERFORMED TO VERIFY THAT THE LINE NUMBER IS
                        ;*              A LEGAL LINE NUMBER FOR THE DUT (IE - LESS THAN NUMLNS).
                        ;*              NOTE: THE LINE NUMBER IS NOT DESTROYED OF ALTERED, SO THIS
                        ;*                      ROUTINE CAN BE USED EASILY IN LOOPS.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

015170                  LINBIT:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
015170  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
015174  042701  177760          BIC     #177760,R1      ;MASK OUT ALL BUT 4 LSBITS OF THE LINE #.
015200  006301                  ASL     R1              ;MULTIPLY LINE # BY 2 TO GET WORD TABLE OFFSET.
015202  016100  002374          MOV     BITTBL(R1),R0   ;GET THE SINGLE BIT BIT MAP.
015206                  60$:    PASS    R0              ;RESTORE GPRS, EXCEPT THE FOLLOWING,
015206  010066  000002                  MOV     R0,R0SLOT(SP)           ;PUT R0 IN STACK SLOT.
015212  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
015214  000207                  RTS     PC              ;R0 - BIT MAP WITH LINE # BIT SET.
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - MSLGET -
                        ;*****************************************************************************
                        ;*      - MILLI SECONDS LOOP WHICH RETURNS READ WORD AND REMAINING TIME -
                        ;*      THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE.  IT IS USED
                        ;*      TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD.  THE
                        ;*      CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
                        ;*      DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
                        ;*      THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
                        ;*      ROUTINE AND THEN ONCE EACH MILLI-SECOND THERE AFTER.
                        ;*      UPON RETURN, THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION
                        ;*      IS RETURNED BY THIS SUBROUTINE.
                        ;*
                        ;* INPUTS:      R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
                        ;*              R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
                        ;*              R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
                        ;*              R4 - ADDRESS OF THE WORD TO TEST.
                        ;*              MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.
                        ;*
                        ;* OUTPUTS:     R0 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
                        ;*              R1 - REMAINING NUMBER OF MS IN TIME-OUT TIME.
                        ;*              CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,MSLGET
                        ;*
                        ;* COMMENTS:    THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
                        ;*              CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
                        ;*              ON THE SYSTEM.
                        ;*              THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
                        ;*              DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
                        ;*              LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
                        ;*              IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
                        ;*              THE DESIRED CONDITION BEFORE RETURNING.  IT INDICATES SUCCESS
                        ;*              IF THE CONDITION IS MET, FAILURE OTHERWISE.
                        ;*
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;*****************************************************************************

015216                  MSLGET:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
015216  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; SET UP MASK FOR REMOVING UNUSED BITS IN THE TEST WORD, AND CLEAR UNUSED
                        ; BITS IN THE DESIRED STATE WORD TO ALLOW DIRECT COMPARISON.
                        ;-
015222  005102                  COM     R2              ;GET MASK OF UNUSED BITS.
015224  040203                  BIC     R2,R3           ;MASK OUT UNUSED BITS IN DESIRED STATE WORD.
                        ;+
                        ; HANDLE THE TEST AND EXIT IF WE HAVE A 0 TIME-OUT VALUE.
                        ;-
015226  005701                  TST     R1              ;TEST THE TIME-OUT VALUE FOR ZERO.
015230  001011                  BNE     2$              ;IF NON-ZERO TIME-OUT, GO LOOP AND TEST.
015232  011400                  MOV     (R4),R0         ;GET THE WORD TO TEST BEFORE EXITING.
015234  010037  015330          MOV     R0,62$          ;SAVE VALUE SO WE CAN RETURN IT.
015240  040200                  BIC     R2,R0           ;MASK OUT UNTESTED BITS OF WORD.
015242  020003                  CMP     R0,R3           ;COMPARE AGAINST DESIRED STATE WORD.
015244  000261                  SEC                     ;INDICATE SUCCESS IN CASE WORDS ARE EQUAL.
```

```
3709  015246  001420                        BEQ     6$              ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.
3710  015250  000241                        CLC                     ;INDICATE FAILURE (TIME-OUT).
3711  015252  000416                        BR      6$              ;EXIT WITH FAILURE, WORDS AREN'T EQUAL.
3712                                ;+
3713                                ; NON-ZERO TIME-OUT VALUE.  LOOP, WAITING FOR CONDITION OR TIME-OUT.
3714                                ;-
3715  015254  013705  002344       2$:     MOV     MSLCNT,R5       ;LOAD MS LOOP COUNT.
3716  015260  011400                4$:     MOV     (R4),R0         ;GET THE WORD TO TEST.
3717  015262  010037  015330               MOV     R0,62$          ;SAVE WORD IN CASE THIS IS THE LAST.
3718  015266  040200                        BIC     R2,R0           ;MASK OUT UNTESTED BITS OF WORD.
3719  015270  020003                        CMP     R0,R3           ;COMPARE AGAINST DESIRED STATE WORD.
3720  015272  000261                        SEC                     ;SET CARRY IN CASE OF SUCCESS.
3721  015274  001405                        BEQ     6$              ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.
3722  015276  005305                        DEC     R5              ;COUNT DOWN THE INSIDE MS LOOP COUNT.
3723  015300  001367                        BNE     4$              ;LOOP IF MS NOT UP.
3724  015302  005301                        DEC     R1              ;DECREMENT THE MS TIME COUNT.
3725  015304  001363                        BNE     2$              ;IF TIME NOT UP, LOOP TO COUNT ANOTHER MS.
3726  015306  000241                        CLC                     ;CLEAR CARRY, WE TIMED-OUT.
3727                                ;+
3728                                ; HAVE EITHER FOUND CONDITION, OR TIMED-OUT (POSSIBLY FROM 0 TIME-OUT VALUE).
3729                                ; RESTORE THE LAST CONTENTS READ FROM THE TEST WORD.  EXIT ROUTINE.
3730                                ;-
3731  015310  013700  015330       6$:     MOV     62$,R0          ;PASS OUT THE LAST READ WORD.
3732  015314                        60$:    PASS    R0,R1           ;RESTORE GPRS, EXCEPT THE FOLLOWING:
3733  015314  010066  000002                       MOV     R0,R0SLOT(SP)           ;PUT R0 IN STACK SLOT.
3734  015320  010166  000004                       MOV     R1,R1SLOT(SP)           ;PUT R1 IN STACK SLOT.
3735  015324  004736                               JSR     PC,@(SP)+               ;RETURN TO PREGOS SUBRT.
3736                                                 ;R0 - LAST READ WORD CHECKED FOR CONDITION.
3737                                                 ;R1 - REMAINING TIME (0 IF TIME-OUT OCCURED).
3738  015326  000207                        RTS     PC              ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.
3739                                ;+
3740                                ; LOCAL STORAGE.
3741                                ;-
3742  015330  000000                62$:    .WORD   0               ;STORAGE FOR THE LAST READ WORD.
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - MSLOOP -
                        ;*****************************************************************************
                        ;*      - TEST LOOP SUBROUTINE -
                        ;*      THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE.  IT IS USED
                        ;*      TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD.  THE
                        ;*      CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
                        ;*      DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
                        ;*      THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
                        ;*      ROUTINE AND THEN ONCE EACH MILLI-SECOND THEREAFTER.
                        ;*
                        ;* INPUTS:      R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
                        ;*              R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
                        ;*              R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
                        ;*              R4 - ADDRESS OF THE WORD TO TEST.
                        ;*              MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.
                        ;*
                        ;* OUTPUTS:     CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,MSLOOP
                        ;*
                        ;* COMMENTS:    THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
                        ;*              CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
                        ;*              ON THE SYSTEM.
                        ;*              THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
                        ;*              DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
                        ;*              LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
                        ;*              IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
                        ;*              THE DESIRED CONDITION BEFORE RETURNING.  IT INDICATES SUCCESS
                        ;*              IF THE CONDITION IS MET, FAILURE OTHERWISE.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: MSLGET.
                        ;*****************************************************************************

015332                  MSLOOP:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
015332  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.

                        ;+
                        ; CALLING THE MSLGET ROUTINE FROM THE MSLOOP ROUTINE ISOLATES THE CALLER OF
                        ; MSLOOP FROM THE RETURNED TEST WORD AND REMAINING TIME-OUT VALUES.
                        ;-
015336  004737  015216          JSR     PC,MSLGET       ;CALL THE MULTI-PURPOSE MS LOOP AND SEARCH RTN.

015342                  60$:    PASS                    ;RESTORE GPRS,
015342  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
015344  000207          RTS     PC                      ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - OOPS -
                                ;++ ************************************************************************
                                ;*                        - PROGRAM ABORT SUBROUTINE -
                                ;*      THIS SUBROUTINE IS USED TO ABORT THE PROGRAM WHEN A FATAL ERROR IS
                                ;*      DETECTED IN THE PROGRAM OR THE HOST SYSTEM HARDWARE.  AN ERROR MESSAGE
                                ;*      IS PRINTED GIVING SOME INFORMATION ABOUT THE NATURE OF THE ABORT.
                                ;*
                                ;* INPUTS:      R1 - ERROR CODE GIVING REASON FOR ABORT.
                                ;*
                                ;* OUTPUTS:     AN ERROR MESSAGE IS PRINTED.
                                ;*              A LIST OF RETURN PC VALUES FOR ALL SUBROUTINE CALLS IS PRINTED.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,OOPS
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE.
                                ;-- ************************************************************************

015346                          OOPS::  SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
015346  004537  004062                          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                                    ; REPORT "HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED." ERROR.
015352                                  ERRSF   101,EM0101
015352  104454                                                                  TRAP    C$ERSF
015354  000145                                                                  .WORD   101
015356  015412                                                                  .WORD   EM0101
015360  000000                                                                  .WORD   0
                                    ; REPORT "PROGRAM HUNG, WAITING FOR A CONTROL-C."
015362                                  PRINTF  #EM0102
015362  012746  015476                                                          MOV     #EM0102,-(SP)
015366  012746  000001                                                          MOV     #1,-(SP)
015372  010600                                                                  MOV     SP,R0
015374  104417                                                                  TRAP    C$PNTF
015376  062706  000004                                                          ADD     #4,SP
015402                          2$:     BREAK                   ;LOOK FOR OPERATOR CONTROL-C INPUT.
015402  104422                                                                  TRAP    C$BRK
015404  000776                          BR      2$              ;INFINITE LOOP.
015406                          60$:    PASS                    ;DON'T NEED THIS, BUT SOMEBODY MAY CHANGE THIS
015406  004736                                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
015410  000207                          RTS     PC              ; ROUTINE IN THE FUTURE, SO BE CONSISTANT.

015412  047510  052123  041440  EM0101:: .ASCIZ /HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED./
015420  046517  052520  042524
015426  020122  040510  042122
015434  040527  042522  047440
015442  020122  047523  052106
015450  040527  042522  041040
015456  043525  042440  041516
015464  052517  052116  051105
015472  042105  000056
015476  047045  040445  051120  EM0102:: .ASCIZ /%N%APROGRAM HUNG, WAITING FOR A CONTROL-C. <**************%N%N/
015504  043517  040522  020115
015512  052510  043516  020054
015520  040527  052111  047111
015526  020107  047506  020122
015534  020101  047503  052116
```

```
 3844  015542  047522  026514  027103
 3845  015550  036040  025052  025052
 3846  015556  025052  025052  025052
 3847  015564  025052  022452  022516
 3848  015572  000116
 3849                                  .EVEN
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - PRTLPR -
                        ;++ ************************************************************************
                        ;*                      -PRINT THE CONTENTS OF THE LPR.
                        ;*      THIS ROUTINE IS USED TO PRINT OUT EXTENDED INFORMATION ON THE
                        ;*      CONTENTS OF THE LINE PARAMETER REGISTER (LPR).
                        ;*
                        ;* INPUTS:      R3 - CONTAINS THE NUMBER OF THE LINE YOU WISH TO EXAMINE.
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DUT'S CSR.
                        ;*              IESTAT - CONTAINS THE CURRENT STATUS OF THE TX AND RX INTERRUPT
                        ;*                       ENABLE BITS IN THE DUT'S CSR.
                        ;*              LPRA - CONTAINS THE ADDRESS OF THE DUT'S LPR REGISTER.
                        ;*
                        ;* OUTPUTS:     AN EXTENDED INFORMATION MESSAGE IS PRINTED ON THE OPERATORS
                        ;*              CONSOLE.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,PRTLPR
                        ;*
                        ;* COMMENTS:    THIS ROUTINE CHANGES THE INDIRECT ADDRESS FIELD OF THE DEVICE
                        ;*              UNDER TEST'S CSR.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

015574                  PRTLPR::SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
015574  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
015600  013701  002246          MOV     CSRA,R1         ;GET THE CSR ADDRESS.
015604  013702  002252          MOV     LPRA,R2         ;GET THE LPR ADDRESS.
015610  042703  177760          BIC     #177760,R3      ;CLEAR ANY UNWANTED BITS.
015614  053703  002274          BIS     IESTAT,R3       ;SET STATE OF TX AND RX INTERRUPT ENABLE BITS.
015620  010311                  MOV     R3,(R1)         ;SELECT LINE.
015622  011204                  MOV     (R2),R4         ;GET CONTENTS OF THE LPR.
                                ;PRINT MESSAGE"CONTENTS OF THE LPR:NNNNNN"
015624                          PRINTX #EF9019,#EM9026,R4;PRINT OUT MESSAGE ON OPERATORS CONSOLE.
015624  010446                                                  MOV     R4,-(SP)
015626  012746  012104                                          MOV     #EM9026,-(SP)
015632  012746  005002                                          MOV     #EF9019,-(SP)
015636  012746  000003                                          MOV     #3,-(SP)
015642  010600                                                  MOV     SP,R0
015644  104415                                                  TRAP    C$PNTX
015646  062706  000010                                          ADD     #10,SP
015652                  60$:    PASS                    ;RESTORE GPRS.
015652  004736                          JSR     PC,@(SP)+       ;RETURN TO PREG05 SUBRT.
015654  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - PUFIFO -
                        ;*****************************************************************************
                        ;*      - PURGE THE FIFO
                        ;*      THIS ROUTINE TRIES TO REMOVE ALL THE CHARACTERS FROM THE FIFO.
                        ;*      ANY BMP CODES THAT ARE FOUND ARE SAVED ON THE BMP CODE QUEUE.
                        ;*
                        ;* INPUTS:      RBUFA- CONTAINS THE  ADDRESS OF THE RECEIVER.
                        ;*
                        ;*
                        ;* OUTPUTS:     CARRY BIT - INDICATES THE STATE OF THE FIFO, SET:= PURGED.
                        ;*              BMPCQ - THE CONTENTS OF THE BMP CODE QUEUE MAY BE UPDATED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,PUFIFO
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: SAVBMP.
                        ;*****************************************************************************

015656                  PUFIFO::SAVE                            ;SAVE CONTENTS OF GPRS R0 THRU R5.
015656  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
015662  012701  001000          MOV     #512.,R1                ;SET MAXIMUM TRY COUNT OF 512.
015666  013704  002250          MOV     RBUFA,R4                ;GET ADDRESS OF THE RECEIVER BUFFER REGISTER.

015672  011402          2$:     MOV     (R4),R2                 ;GET THE CONTENTS OF THE RECEIVER BUFFER REG.
015674  100016          BPL     6$                              ;EXIT IF THE FIFO IS EMPTY, DATA_VALID CLR.
                        ;+
                        ; CHECK IF THE READ CHARACTER IS ACTUALLY A BMP CODE.
                        ;  IF IT IS, THEN SAVE IT ON THE BMP CODE QUEUE TO BE REPORTED LATER.
                        ;-
015676  012700  070000          MOV     #70000,R0               ;GENERATE A BIT MAP OF CHAR ERROR BITS
015702  040200          BIC     R2,R0                           ; WHICH ARE NOT SET FOR CHAR.
015704  001006          BNE     4$                              ;THROW CHAR AWAY IF NOT BMP OR SELFTEST CODE.
                          ;+
                          ; CHECK IF THE READ DATA IS MODEM STATUS , BMP OR SELFTEST?.
                          ;-
015706  012700  000300          MOV     #300,R0                 ; CHECK IF BMP OR SELFTEST?.
015712  040200          BIC     R2,R0                           ;TRY TO CLEAR BMP FLAGS IN THE READ DATA.
015714  001002          BNE     4$                              ;IF IT IS MODEM OR SELFTEST CODE THROW IT AWAY.
015716  004737  016220          JSR     PC,SAVBMP               ;SAVE BMP CODE ON THE QUEUE.

015722  005301          4$:     DEC     R1                      ;DECREMENT THE TRY COUNT.
015724  001362          BNE     2$                              ;LOOP TO TRY AGAIN.
015726  000241          CLC                                     ;CLEAR CARRY,TO INDICATE FIFO NOT PURGED.
015730  000401          BR      60$                             ;EXIT WITH CARRY CLEAR.
015732  000261          6$:     SEC                             ;SET CARRY, TO INDICATE FIFO PURGED.

015734                  60$:    PASS                            ;RESTORE GPRS,
015734  004736                          JSR     PC,@(SP)+       ;RETURN TO PREG05 SUBRT.
                                                                ;CARRY BIT, SET INDICATES FIFO PURGED.
015736  000207          RTS     PC
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - READBX -
                                ;++ ************************************************************************
                                ;*      - READ CHARACTERS FROM THE FIFO AND CHECKS FOR BMPS AND XONS-
                                ;*      THIS SUBROUTINE IS USED IN THE FIHAVL.TST.
                                ;*      IT READS THE SPECIFIED NUMBER OF CHARACTERS FROM THE FIFO AND CHECKS
                                ;*      FOR BMP CODES AND XON CHARACTERS.
                                ;*
                                ;* INPUTS:      R0 - CONTAINS THE NUMBER OF CHARS TO READ FROM THE FIFO.
                                ;*
                                ;* OUTPUTS:     R1 - CONTAINS ADDRESS OF ERROR MESSAGE TO BE REPORTED
                                ;*                   CLEAR IF NO ERROR FOUND.
                                ;*              CARRY USED TO INDICATE IF FIFO WAS FOUND EMPTY, CARRY CLEAR.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,READ
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: CHKBMP.
                                ;-- ************************************************************************

015740                          READBX:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
015740  004537  004062                        JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
015744  005001                    CLR     R1                          ;CLEAR GPR THAT HOLDS THE ADDRESS OF ERRMSG.
015746  013703  002250            MOV     RBUFA,R3                    ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
015752  011302                2$: MOV     (R3),R2                     ;READ A CHARACTER FROM THE FIFO.
015754  100015                    BPL     8$                          ;BRANCH IF FIFO IS EMPTY.
                                  ;+
                                  ; CHECK IF THE READ CHARACTER IS A BMP CODE.
                                  ;  IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
                                  ;  ABORT THE TEST.
                                  ;-
015756  004737  014360                JSR     PC,CHKBMP               ;CHECK IF CHARACTER IS A BMP CODE.
015762  103410                    BCS     6$                          ;BRANCH IF A BMP CODE WAS FOUND.
015764  120227  000021            CMPB    R2,#21                      ;CHECK IF IT IS AN XON.
015770  001003                    BNE     4$                          ;BRANCH IF NOT AN XON.
015772  012701  010665            MOV     #EM5402,R1                  ;PASS THE MESSAGE TO BE REPORTED.
015776  000402                    BR      6$                          ;GO EXIT TEST.
016000  005300                4$: DEC     R0                          ;DECREMENT THE READ COUNT.
016002  001363                    BNE     2$
016004  000261                6$: SEC                                 ;SET CARRY TO INDICATE SUCCESS.
016006  000401                    BR      60$                         ;EXIT
016010  000241                8$: CLC                                 ;CLEAR CARRY BIT TO INDICATE FAILURE.

016012                          60$:    PASS    R1                  ;RESTORE GPRS,
016012  010166  000004                                    MOV     R1,R1SLOT(SP)           ;PUT R1 IN STACK SLOT.
016016  004736                                            JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016020  000207                    RTS     PC
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - RESETT -
                                ;*****************************************************************************
                                ;*              - RESET DEVICE UNDER TEST -
                                ;*      THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
                                ;*      IF RESET DOES NOT SUCCESFULLY COMPLETE, IE. TIME-OUT OCCURS, THEN
                                ;*      AN ABORT TEST ERROR MESSAGE IS REPORTED.
                                ;*
                                ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR
                                ;*              TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
                                ;*              ERRTBL- ERRTYP,ERNBR,AND ERRMSG SET UP CORRECTLY.
                                ;*
                                ;* OUTPUTS:     THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
                                ;*              CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
                                ;*              ERRBLK - VALUE MAY BE DESTROYED.
                                ;*              IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
                                ;*              TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,RESETT
                                ;*
                                ;* COMMENTS:    THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS INITIAL ERRNBR
                                ;*              THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: DELAY,MSLGET.
                                ;*****************************************************************************

016022                          RESETT:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
016022  004537  004062                         JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016026  012702  000040                 MOV     #BIT05,R2       ;SET BIT MASK OF MASTER RESET BIT.
                                ;+
                                ; TEST THE STATE OF THE MASTER RESET BIT IN THE CSR.
                                ;  IF MR IS SET THEN WAIT FOR SELF-TEST TO COMPLETE.
                                ;   IF TIME-OUT OCCURS, REPORT THE ERROR AND PASS-OUT ABORT TEST INDICATOR.
                                ;-
016032  013704  002246                  MOV     CSRA,R4         ;GET THE ADDRESS OF THE DUT'S CSR.
016036  030214                  BIT     R2,(R4)         ;CHECK STATE OF MASTER RESET BIT.
016040  001406                 BEQ     2$              ;DON'T DELAY IF MR IS ALREADY CLEAR.
016042  005003                  CLR     R3              ;SET UP DESIRED STATE OF MASTER RESET BIT.
016044  012701  004704          MOV     #2500.,R1       ;PASS TIME-OUT VALUE OF 2.5 SECONDS.
016050  004737  015216          JSR     PC,MSLGET       ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.
016054  103012                  BCC     4$              ;GO REPORT ERROR IF TIMEOUT OCCURRED.

                                ;+
                                ; SET MASTER RESET BIT IN CSR.  CLEAR TX AND RX ENABLE BITS, ETC.
                                ;  SKIP THE SELFTEST.
                                ;  TIME-OUT OF 2.5 SECS, JUST IN CASE THE SELF-TEST EXECUTES.
                                ;-
016056  010277  164164          2$:     MOV     R2,@CSRA        ;SET MASTER RESET BIT, DISABLE TX AND RX INTS.
016062  004737  016400          JSR     PC,SKPSTS       ;TRY TO SKIP THE SELFTEST.
                                  ;+
                                  ;   SET SELF-TEST TIME-OUT OF 2.5 SECONDS, AND WAIT FOR M.R TO CLEAR.
                                  ;   IF TIME-OUT OCCURS, THEN REPORT THE FATAL ERROR AND PASS-OUT THE ABORT
                                  ;   TEST INDICATOR.
                                  ;-
016066  005003                    CLR     R3              ;SET UP DESIRED STATE OF MASTER RESET BIT.
016070  012701  004704            MOV     #2500.,R1       ;PASS TIME-OUT VALUE OF 2.5 SECONDS.
016074  004737  015216            JSR     PC,MSLGET       ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.
```

```
4047  016100  103410                    BCS     6$              ;SKIP ERROR REPORT IF MR CLEARED IN TIME.
4048                                    ;+
4049                                    ; SET UP ERROR MESSAGE TO REPORT "FATAL ERROR FOUND DURING RESET,TEST ABORTED".
4050                                    ; INDICATE TEST IS TO BE ABORTED BY CLEARING THE CARRY BIT.
4051                                    ;-
4052  016102  012701  005415            4$:     MOV     #EM1601,R1      ;PASS ERROR MESSAGE TO REPORT.
4053  016106  012737  012732  004060            MOV     #ER1603,ERRBLK ;PASS ADDRESS OF ERROR HANDLING ROUTINE.
4054                                            ;REPORT ERROR "TIME-OUT OCCURRED WAITING FOR MASTER RESET TO CLEAR"
4055                                            ; "TEST ABORTED"
4056  016114                            ;       ERROR                   ;               >>>>> ERROR <<<<<
4057  016114  104460                                                                            TRAP    C$ERROR
4058  016116  000241                            CLC                     ;INDICATE TEST IS TO BE ABORTED.
4059  016120  000403                            BR      60$             ;EXIT THIS SUBROUTINE, ABORT TEST INDICATOR.
4060                                    ;+
4061                                    ; CLEAR TX AND RX INTERRUPT ENABLE STATUS FLAGS IN IESTAT.
4062                                    ; EXIT WITH CONTINUE TEST INDICATOR SET (IE,CARRY SET).
4063                                    ;-
4064  016122  005037  002274            6$:      CLR    IESTAT          ;CLEAR TX AND RX INTERRUPT STATUS FLAGS.
4065  016126  000261                             SEC                    ;INDICATE SUCCESS, CONTINUE TEST.
4066
4067  016130                            60$:    PASS                    ;RESTORE GPRS, PASS THE FOLLOWING INTACT:
4068  016130  004736                                           JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
4069                                                                    ;CARRY BIT:IF CLEAR,INDICATES ABORT TEST.
4070  016132  000207                            RTS     PC
4071
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - RXIE0 -
                        ;++ ************************************************************************
                        ;*                        - RECEIVER INTERRUPT DISABLE -
                        ;*      THIS ROUTINE IS USED TO DISABLE RECEIVER INTERRUPTS IN THE DHV11.
                        ;*
                        ;* INPUTS:       NONE.
                        ;*
                        ;* OUTPUTS:      THE RX.INT.ENBL BIT IS CLEARED IN THE DUT CSR.
                        ;*               IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
                        ;*               ENABLE BITS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,RXIE0
                        ;*
                        ;* COMMENTS:     THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
                        ;*               THE DUT CSR ARE DESTROYED.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************
016134  010046          RXIE0:: MOV     R0,-(SP)        ;SAVE CONTENTS OF R0 ON THE STACK.
016136                          GETPRI -(SP)            ;SAVE PROCESSOR PRIORITY ON STACK.
016136  104440                                                          TRAP    C$GPRI
016140  010046                                                          MOV     R0,-(SP)
016142                          SETPRI #PRI07           ;IGNORE ANY INTERRUPT THAT MAY BE GENERATED.
016142  012700  000340                                                  MOV     #PRI07,R0
016146  104441                                                          TRAP    C$SPRI
016150  042737  137777  002274      BIC     #137777,IESTAT  ;CLEAR RX.INT.ENBL BIT IN IESTAT.
016156  013777  002274  164062      MOV     IESTAT,@CSRA    ;DISABLE RX INTERRUPTS.
016164                          SETPRI (SP)+            ;ENABLE INTERRUPTS TO THE PROCESSOR AGAIN.
016164  012600                                                          MOV     (SP)+,R0
016166  104441                                                          TRAP    C$SPRI
016170  012600                  MOV     (SP)+,R0        ;RESTORE R0.
016172  000207                RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - RXIE1 -
                        ;++ ******************************************************************************
                        ;*                          - RECEIVER INTERRUPT ENABLE -
                        ;*      THIS ROUTINE IS USED TO ENABLE RECEIVER INTERRUPTS IN THE DHV11.
                        ;*
                        ;* INPUTS:      NONE.
                        ;*
                        ;* OUTPUTS:     THE RX.INT.ENBL BIT IS SET IN THE DUT CSR.
                        ;*              IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
                        ;*              ENABLE BITS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,RXIE1
                        ;*
                        ;* COMMENTS:    THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
                        ;*              THE DUT CSR ARE DESTROYED.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ******************************************************************************

016174  052737  000100  002274  RXIE1::  BIS     #BIT06,IESTAT   ;SET RX.INT.ENBL BIT IN IESTAT.
016202  042737  137677  002274           BIC     #137677,IESTAT  ;CLEAR ALL OTHER BITS, EXCEPT TX AND RX I.E.
016210  013777  002274  164030           MOV     IESTAT,@CSRA    ;ENABLE RX INTERRUPTS.
016216  000207                           RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - SAVBMP -
                        ;++ ************************************************************************
                        ;*                      - SAVE BMP CODES ROUTINE -
                        ;*      THIS ROUTINE SAVES THE PARAMETER PASSED IN, ONTO THE BMP CODE QUEUE
                        ;*      TOGETHER WITH THE NUMBER OF THE CURRENTLY EXECUTING TEST.
                        ;*
                        ;* INPUTS:      R2 - CONTAINS THE BMP CODE THAT IS TO BE PLACED ON THE QUEUE.
                        ;*              BMPCQP - CONTAINS ADDRESS OF NEXT LOCATION IN THE BMP QUEUE.
                        ;*              BMPCQB - LABEL AT BASE OF THE BMP CODE QUEUE.
                        ;*              BMPCQE - LABEL OF NEXT LOCATION AFTER THE END OF THE BMP QUEUE.
                        ;*              TSTNUM - CONTAINS THE NUMBER OF THE CURRENT TEST.
                        ;*
                        ;* OUTPUTS:     BMPCQP -  INCREMENTED BY 4.
                        ;*              THE CONTENTS OF THE BMP CODE QUEUE ARE UPDATED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,SAVBMP
                        ;*
                        ;* COMMENTS:    IF THE OVERFLOW OCCURS THEN THE LAST LOCATION WILL BE
                        ;*              OVERWRITTEN BY ANY SUBSEQUENT ATTEMPTS TO UPDATE THE QUEUE.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

016220                  SAVBMP:: SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
016220  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016224  013704  002450          MOV     BMPCQP,R4       ;GET THE POINTER TO THE NEXT LOCATION IN QUEUE.
016230  113724  002272          MOVB    TSTNUM,(R4)+    ;SAVE THE CURRENT TEST NUMBER ON THE QUEUE.
016234  005204                  INC     R4              ;INCREMENT THE POINTER TO GIVE AN EVEN ADDRESS.
016236  042702  177400          BIC     #177400,R2      ;CLEAR THE UNWANTED BITS FROM THE BMP CODE.
016242  010224                  MOV     R2,(R4)+        ;SAVE THE BMP CODE ON THE QUEUE.
016244  020427  002652          CMP     R4,#BMPCQE      ;CHECK IF OVERFLOW WILL OCCUR THE NEXT TIME.
016250  103402                  BLO     2$              ;GO SAVE THE POINTER IF WE WILL NOT OVERFLOW.
016252  162704  000004          SUB     #4,R4           ;RESET THE POINTER TO THE LAST LOCATION IN QUE.
016256  010437  002450  2$:     MOV     R4,BMPCQP       ;SAVE THE POINTER.

016262                  60$:    PASS                    ;RESTORE GPRS.
016262  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016264  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - SAVMST -
                        ;++ ************************************************************************
                        ;*                      - SAVE MODEM STATUS ROUTINE -
                        ;*      THIS ROUTINE SAVES THE PRESENT CONTENTS OF THE DUT STAT REGISTERS IN
                        ;*      THE STAT STORAGE TABLE.
                        ;*
                        ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
                        ;*              IESTAT - STATE OF THE DUT CSR INTERRUPT ENABLE BITS.
                        ;*              NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
                        ;*              STATA - CONTAINS THE ADDRESS OF THE DUT STAT REGISTER.
                        ;*              STSTB - LABEL AT BASE OF THE STAT STORAGE TABLE.
                        ;*
                        ;* OUTPUTS:     STST TABLE - OVERWRITTEN WITH PRESENT STAT CONTENTS.
                        ;*              CSR IND.ADR.REG FIELD - DESTROYED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,SAVMST
                        ;*
                        ;* COMMENTS:    IF THE CONTENTS OF IESTAT CHANGES DURING THIS TEST THE CSR
                        ;*              INTERRUPT ENABLE BITS WILL NOT TRACK THE CHANGE.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

016266                  SAVMST:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
016266  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016272  013701  002274          MOV     IESTAT,R1       ;GET IE STATES FOR UPDATING IND.ADR.REG FIELD.
016276  012702  002652          MOV     #STSTB,R2       ;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
016302  012703  000010          MOV     #NUMLNS,R3
016306  050103                  BIS     R1,R3           ;FORM COMPLETION COMPARISON WORD.
016310  010177  163732  2$:     MOV     R1,@CSRA        ;SET UP THE CSR IND.ADR.REG FIELD.
016314  017722  163734          MOV     @STATA,(R2)+    ;SAVE CONTENTS OF THIS LINE'S STAT REGISTER.
016320  005201                  INC     R1              ;SET LINE COUNTER TO NEXT LINE.
016322  020103                  CMP     R1,R3           ;CHECK FOR ALL LINES DONE.
016324  002771                  BLT     2$              ;LOOP IF NOT ALL LINES DONE.

016326                  60$:    PASS                    ;RESTORE GPRS.
016326  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016330  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - SETPAR -
                        ;++ ************************************************************************
                        ;*                          - SET TX AND CONTROL PARAMETERS -
                        ;*      THIS SUROUTINE IS USED IN THE FIHAVL.TST.
                        ;*      IT INITIALISES THE SELECTED LINE TO THE FOLLOWING STATE:
                        ;*      INTERNAL LOOPBACK, IAUTO ENABLED, LPR:38.4K, 8 BITS/CHAR, 2 STOP,
                        ;*      ODD PARITY.
                        ;*
                        ;* INPUTS:      R1 - CONTAINS NUMBER OF THE LINE TO BE INITIALISED.
                        ;*
                        ;* OUTPUTS:     LNCTRL AND LPR REGISTERS FOR THE SELECTED LINE ARE DESTROYED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,SETPAR
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: DELAY,WTWLNC,WTWLPR.
                        ;-- ************************************************************************

016332                  SETPAR:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
016332  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016336  004737  015170          JSR     PC,LINBIT               ;GET A BIT MAP FOR THIS LINE.
016342  010005           MOV     R0,R5                           ;COPY THE LINE BIT MAP.
016344  012700  000206   MOV     #206,R0                         ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
016350  004737  017412          JSR     PC,WTWLNC               ;INITILAISE THE LINE CONTROL REGISTER.
016354  012700  177670   MOV     #177670,R0                      ;PASS THE LPR CONTENTS.
016360  004737  017466          JSR     PC,WTWLPR               ;SET THE LPR CONTENTS TO 38.4K BAUD.
016364  013704  000012   MOV     10.,R4                          ;PASS DELAY TIME OF 10 MILLI SECONDS.
016370  004737  014574          JSR     PC,DELAY                ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.

016374                  60$:    PASS                            ;RESTORE GPRS.
016374  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016376  000207          RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - SKPSTS -
                        ;++ ************************************************************************
                        ;*                        - SKIP SELFTEST ROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO SKIP THE SELFTEST AFTER A DUT RESET HAS BEEN
                        ;*      INITIATED.  IT MUST BE ENTERED IMMEDIATELY AFTER SETTING THE DUT MASTER
                        ;*      RESET ROUTINE OR AFTER THE EXECUTION OF A BUS RESET (BECAUSE OF TIMING
                        ;*      CONSIDERATIONS).
                        ;*
                        ;* INPUTS:      CSRA - CONTAINS ADDRESS OF THE DUT CSR.
                        ;*              TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
                        ;*
                        ;* OUTPUTS:     SKIP SELFTEST CODES ARE WRITTEN TO THE DUT REGISTERS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,SKPSTS
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: DELAY.
                        ;-- ************************************************************************

016400                  SKPSTS:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
016400  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016404  012704  000012          MOV     #10.,R4                 ;PASS DELAY VALUE OF 10 MILLI-SECONDS.
016410  004737  014574          JSR     PC,DELAY                ;DELAY FOR 10 MILLI-SECONDS.
                        ;+
                        ; WRITE SKIP SELF-TEST CODE (52525) TO ALL THE INDEXED DUT REGISTERS.
                        ;-
016414  012701  000050          MOV     #NUMLNS!BIT05,R1 ;FORM IND.ADR.REG FIELD (PLUS M.R. BIT) WORD.
                        ;THE ABOVE INCLUSION OF THE M.R. BIT IS NECESSARY BECAUSE OF THE
                        ;  LACK OF A M.R. BIT WRITE LOCK-OUT ON THE DHV-11.
016420  012703  052525          MOV     #52525,R3               ;INITIALISE THE SKIP SELF-TEST CODE.
016424  005301          4$:     DEC     R1                      ;SELECT THE NEXT SET OF DEVICE REGISTERS.
016426  013704  002246          MOV     CSRA,R4                 ;GET THE ADDRESS OF THE CSR OF THE DUT.
016432  010124                  MOV     R1,(R4)+                ;SELECT A BANK OF DUT REGISTERS.
016434  010324          6$:     MOV     R3,(R4)+                ;WRITE THE CODE TO A DUT REGISTER.
016436  020437  002264          CMP     R4,TXBFCA               ;COMPARE POINTER WITH LAST REGISTER ADDRESS.
016442  103774                  BLO     6$                      ;LOOP IF NOT ALL REGS DONE IN THIS BANK.
016444  032701  000017          BIT     #17,R1                  ;TEST FOR IND.ADR.REG FIELD DECREMENTED TO 0.
016450  001365                  BNE     4$                      ;LOOP UNTIL ALL REGISTERS CONTAIN THE CODE.

016452          60$:    PASS                                    ;RESTORE GPRS.
016452  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016454  000207                  RTS     PC
```

```
                                .SBTTL  GLOBAL SUBROUTINE                - TSABRT -
                                ;++ ******************************************************************************
                                ;*                          - TEST ABORT ROUTINE -
                                ;*      THIS SUBROUTINE IS USED WHEN A NON-TEST RELATED ERROR HAS BEEN FOUND
                                ;*      DURING THE EXECUTION OF THE CURRENT TEST.
                                ;*      IT IS USED TO INFORM THE OPERATOR THAT THE CURRENT TEST HAS BEEN
                                ;*      ABORTED.
                                ;*
                                ;* INPUTS:      ERRMSG - CONTAINS THE NAME OF THE CURRENT TEST.
                                ;*              ERRNBR - CONTAINS THE CORRECT ERROR NUMBER.
                                ;*              THE REMAINDER OF THE ERRTBL IS CORRECTLY INITIALISED.
                                ;*
                                ;* OUTPUTS:     MESSAGES ARE REPORTED TO THE OPERATOR.
                                ;*
                                ;* CALLING SEQUENCE:    JSR     PC,TSABRT
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: ER1603.
                                ;-- ******************************************************************************

016456                          TSABRT:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
016456  004537  004062                         JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016462  012701  016500                 MOV     #2$,R1                  ;PASS ADDRESS OF FIRST MESSAGE TO BE REPORTED.
016466  012737  012732  004060         MOV     #ER1603,ERRBLK          ;SET-UP THE ERROR REPORTING ROUTINE.
016474                                 ERROR                           ;               >>>>> ERROR <<<<<.
016474  104460                                                                          TRAP    C$ERROR
016476  000432                         BR      60$                     ;
016500  047040  047117  051055  2$:    .ASCIZ / NON-RELATED TEST ERROR FOUND DURING TEST EXECUTION/
016506  046105  052101  042105
016514  052040  051505  020124
016522  051105  047522  020122
016530  047506  047125  020104
016536  052504  044522  043516
016544  052040  051505  020124
016552  054105  041505  052125
016560  047511  000116
                                .EVEN
016564                          60$:    PASS                            ;RESTORE GPRS.
016564  004736                                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016566  000207                         RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - TXDATP -
                        ;++ ************************************************************************
                        ;*                        - TRANSMIT DATA PATTERN -
                        ;*      THIS SUBROUTINE IS USED IN THE FIHAVL.TST.
                        ;*      IT TRANSMITS A SPECIFIED NUMBER OF DATA BYTES ON THE SPECIFIED LINE.
                        ;*
                        ;* INPUTS:      R0 - CONTAINS THE NUMBER OF DATA BYTES TO TX.
                        ;*              R1 - CONTAINS LINE NUMB ON WHICH TRANSMISSION IS TO TAKE PLACE.
                        ;*              BUFBAS TO BUFMID CONTAINS A 256 BYTE DATA PATTERN.
                        ;*
                        ;* OUTPUTS:     DATA IS SENT OUT ON THE SPECIFIED LINE.
                        ;*              CARRY SET = TX SUCCESSFUL.
                        ;*
                        ;* CALLING SEQUENCE:    TXDATP
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: DODMA.
                        ;-- ************************************************************************

016570                  TXDATP:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
016570  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016574  010003                  MOV     R0,R3           ;PASS THE NUMBER OF CHARS TO TX.
016576  012702  002712          MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
016602  004737  014634          JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
016606                  60$:    PASS                    ;RESTORE GPRS.
016606  004736                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
016610  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - TXDSBL -
                        ;++ ************************************************************************
                        ;*                        - TRANSMITTER DISABLE -
                        ;*      THIS SUBROUTINE IS USED TO DISABLE TRANSMISSION ON SELECTED LINES BY,
                        ;*      CLEARING THE ASSOCIATED TX.ENABLE BIT ON THE DUT.
                        ;*
                        ;* INPUTS:      R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO CLEAR TX.ENABLE.
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
                        ;*              IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
                        ;*              NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
                        ;*              TXAD2A - CONTAINS THE ADDRESS OF THE TBUFFAD2 REGISTER.
                        ;*
                        ;* OUTPUTS:     R5 - BIT'S SET INDICATE THE INITIAL STATES OF ALL TX.ENBLE BITS.
                        ;*              TBUFFAD2 - THE STATE OF THE TX.ENBLE BIT MAY BE ALTERED.
                        ;*              THE CONTENTS OF THE IND.ADD.REG FIELD IN THE CSR ARE DESTROYED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,TXDSBL
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

016612                  TXDSBL:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
016612  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016616  010500          MOV     R5,R0           ;COPY BIT MAP OF LINES TO DISABLE TRANSMISSION.
016620  012701  000001  MOV     #BIT0,R1        ;INITIALIZE THE SELECTED LINE BIT MASK.
016624  013702  002262  MOV     TXAD2A,R2       ;GET THE ADDRESS OF THE TBUFFAD2 REGISTER.
016630  005202          INC     R2              ;GET THE ADDRESS OF THE MSBYTE OF TBUFFAD2 REG.
016632  012703  000010  MOV     #NUMLNS,R3      ;GET MAXIMUM LINE NUMBER PLUS ONE.
016636  013704  002274  MOV     IESTAT,R4       ;GET THE STATES OF THE INT ENABLE BITS.
016642  005005          CLR     R5              ;LOG POSSIBLE TX DISABLED ON ALL LINES.
                        ;+
                        ; SELECT EVERY LINE IN TURN, AND LOG THE STATE OF EACH TX.ENABLE BIT.
                        ;-
016644  010477  163376  2$:     MOV     R4,@CSRA        ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
016650  105712          TSTB    (R2)            ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
016652  100001          BPL     4$              ;SKIP NEXT INSTRUCTION IF TX.ENABLE CLEAR.
016654  050105          BIS     R1,R5           ;LOG TX ENABLE BIT SET FOR SELECTED LINE.
                        ;+
                        ; CLEAR TX.ENBLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE TX DISABLE
                        ; LINE BIT MAP.
                        ;-
016656  030100          4$:     BIT     R1,R0           ;CHECK STATE OF DISABLE LINE BIT MAP.
016660  001402          BEQ     6$              ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
016662  142712  000200  BICB    #BIT7,(R2)      ;CLEAR TX.ENABLE BIT ON SELECTED LINE.
016666  005204          6$:     INC     R4              ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
016670  006301          ASL     R1              ;SHIFT BIT MAP FOR NEXT LINE.
016672  005303          DEC     R3              ;DECREMENT LINE NUMBER.
016674  001363          BNE     2$              ;LOOP TO CHECK NEXT LINE.

016676                  60$:    PASS    R5              ;RESTORE GPRS,EXCEPT
016676  010566  000014                  MOV     R5,R5SLOT(SP)           ;PUT R5 IN STACK SLOT.
016702  004736                  JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                        ;R5 - PREVIOUS STATES OF ALL TX.ENABLE BITS.
016704  000207          RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - TXENBL -
                        ;++ ************************************************************************
                        ;*                        - TRANSMITTER ENABLE -
                        ;*      THIS SUBROUTINE IS USED TO ENABLE TRANSMISSION ON SELECTED LINES BY
                        ;*      SETTING THE ASSOCIATED TX.ENABLE BIT ON THE DUT.
                        ;*
                        ;* INPUTS:      R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO SET TX.ENABLE.
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
                        ;*              IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
                        ;*              NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
                        ;*              TXAD2A - CONTAINS THE ADDRESS OF THE TBUFFAD2 REGISTER.
                        ;*
                        ;* OUTPUTS:     R5 - BIT'S SET INDICATE PREVIOUSLY DISABLED LINES.
                        ;*              TBUFFAD2 - THE STATE OF THE TX.ENBLE BIT MAY BE ALTERED.
                        ;*              THE CONTENTS OF THE IND.ADD.REG FIELD IN THE CSR ARE DESTROYED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,TXENBL
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

016706                  TXENBL:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
016706  004537  004062                 JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
016712  010500                  MOV     R5,R0                   ;COPY BIT MAP OF LINES TO ENABLE.
016714  012701  000001          MOV     #BIT0,R1                ;INITIALIZE THE SELECTED LINE BIT MASK.
016720  013702  002262          MOV     TXAD2A,R2               ;GET THE ADDRESS OF THE TBUFFAD2 REGISTER.
016724  005202                  INC     R2                      ;GET THE ADDRESS OF THE MSBYTE OF TBUFFAD2 REG.
016726  012703  000010          MOV     #NUMLNS,R3              ;GET MAXIMUM LINE NUMBER.
016732  013704  002274          MOV     IESTAT,R4               ;GET THE STATES OF THE INT ENABLE BITS.
016736  005005                  CLR     R5                      ;CLEAR TX.ENABLE BIT LOG OF DISABLED LINES.
                        ;+
                        ; SELECT EVERY LINE IN TURN,AND LOG ANY TX.ENBLE BIT THAT IS CLEAR.
                        ;-
016740  010477  163302  2$:      MOV     R4,@CSRA                ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
016744  105712                  TSTB    (R2)                    ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
016746  100401                BMI     4$                      ;SKIP NEXT INSTRUCTION IF TX.ENABLE SET.
016750  050105                  BIS     R1,R5                   ;LOG TX ENABLE BIT CLEAR FOR SELECTED LINE.
                         ;+
                         ;  SET TX.ENBLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE TX ENABLE
                         ;  LINE BIT MAP.
                         ;-
016752  030100          4$:      BIT     R1,R0                   ;CHECK STATE OF TX.ENABLE LINE BIT MAP.
016754  001402                BEQ     6$                      ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
016756  152712  000200          BISB    #BIT7,(R2)              ;ENABLE TRANSMISSION ON SELECTED LINE.
016762  005204          6$:      INC     R4                      ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
016764  006301                  ASL     R1                      ;SHIFT BIT MAP FOR NEXT LINE.
016766  005303                  DEC     R3                      ;DECREMENT LINE NUMBER.
016770  001363                BNE     2$                      ;LOOP TO CHECK NEXT LINE.

016772                  60$:    PASS    R5                      ;RESTORE GPRS,EXCEPT
016772  010566  000014                          MOV     R5,R5SLOT(SP)           ;PUT R5 IN STACK SLOT.
016776  004736                              JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                                ;R5 - LINE BIT MAP CORRESPONDING TO THE
                                                                ; PREVIOUS LINES THAT WERE DISABLED.
```

```
  4460  017000  000207                        RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - TXIE0 -
                        ;++ ************************************************************************
                        ;*                      - TRANSMITTER INTERRUPT DISABLE -
                        ;*      THIS ROUTINE IS USED TO DISABLE TRANSMITTER INTERRUPTS IN THE DHV11.
                        ;*
                        ;* INPUTS:      NONE.
                        ;*
                        ;* OUTPUTS:     THE TX.INT.ENBL BIT IS CLEARED IN THE DUT CSR.
                        ;*              IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
                        ;*              ENABLE BITS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,TXIE0
                        ;*
                        ;* COMMENTS:    THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
                        ;*              THE DUT CSR ARE DESTROYED.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************
017002  010046          TXIE0:: MOV     R0,-(SP)        ;SAVE CONTENTS OF R0 ON THE STACK.
017004                          GETPRI  -(SP)           ;SAVE CURRENT PROCESSOR PRIORITY ON THE STACK.
017004  104440                                                          TRAP    C$GPRI
017006  010046                                                          MOV     R0,-(SP)
017010                          SETPRI  #PRI07          ;IGNORE ANY INTERRUPTS THAT MAY BE GENERATED.
017010  012700  000340                                                  MOV     #PRI07,R0
017014  104441                                                          TRAP    C$SPRI
017016  042737  177677  002274          BIC     #177677,IESTAT  ;CLEAR TX.INT.ENBL BIT IN IESTAT.
017024  013777  002274  163214          MOV     IESTAT,@CSRA    ;DISABLE TX INTERRUPTS.
017032                                  SETPRI (SP)+            ;ENABLE INTERRUPTS TO THE PROCESSOR AGAIN.
017032  012600                                                          MOV     (SP)+,R0
017034  104441                                                          TRAP    C$SPRI
017036  012600                          MOV     (SP)+,R0        ;RESTORE R0.
017040  000207                  RTS     PC
```

```
4493                                    .SBTTL  GLOBAL SUBROUTINE                - TXIE1 -
4494                                    ;++ ************************************************************************
4495                                    ;*                        - TRANSMITTER INTERRUPT ENABLE -
4496                                    ;*      THIS ROUTINE IS USED TO ENABLE TRANSMITTER INTERRUPTS IN THE DHV11.
4497                                    ;*
4498                                    ;* INPUTS:       NONE.
4499                                    ;*
4500                                    ;* OUTPUTS:      THE TX.INT.ENBL BIT IS SET IN THE DUT CSR.
4501                                    ;*               IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
4502                                    ;*               ENABLE BITS.
4503                                    ;*
4504                                    ;* CALLING SEQUENCE:    JSR     PC,TXIE1
4505                                    ;*
4506                                    ;* COMMENTS:     THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
4507                                    ;*               THE DUT CSR ARE DESTROYED.
4508                                    ;*
4509                                    ;* SUBORDINATE ROUTINES CALLED: NONE.
4510                                    ;-- ************************************************************************
4511
4512  017042  052737  040000  002274  TXIE1::  BIS     #BIT14,IESTAT   ;SET TX.INT.ENBL BIT IN IESTAT.
4513  017050  042737  137677  002274          BIC     #137677,IESTAT  ;CLEAR ALL BITS EXCEPT TX RX I.E BITS.
4514  017056  013777  002274  163162          MOV     IESTAT,@CSRA    ;DISABLE TX INTERRUPTS.
4515  017064  000207                          RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE               - UNSDIV -
                        ;++ ************************************************************************
                        ;*                      - UNSIGNED DIVIDE ROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO DIVIDE A 32 BIT UNSIGNED DIVIDEND BY A
                        ;*      16 BIT UNSIGNED DIVISOR GIVING A 16 BIT QUOTIENT.  ALL NUMBERS ARE
                        ;*      CONSIDERED TO BE UNSIGNED.  A SUCCESS FLAG IS NOT SET ON RETURN IF
                        ;*      THE QUOTIENT WAS TOO BIG TO BE CONTAINED IN 16 BITS.
                        ;*
                        ;* INPUTS:      R1 - THE DIVISOR, UNSIGNED, 16 BITS.
                        ;*              R2 - MOST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
                        ;*              R3 - LEAST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
                        ;*
                        ;* OUTPUTS:     R1 - QUOTIENT, UNSIGNED, 16 BITS (177777 IF OVERFLOW).
                        ;*              CARRY - SUCCESS FLAG, SET IF COMPLETE QUOTIENT FITS IN 16 BITS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,UNSDIV
                        ;*
                        ;* COMMENTS:    IF THE DIVISOR IS 0 THE QUOTIENT IS RETURNED AS ALL ONES
                        ;*              (177777) AND THE CARRY IS CLEAR REGARDLESS OF THE DIVIDEND.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

017066                  UNSDIV:: SAVE                   ;SAVE CONTENTS OF GPRS R0 THRU R5.
017066  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; CHECK FOR QUOTIENT GREATER THAN 16 BITS CONDITION.
                        ;-
017072  010204                  MOV     R2,R4           ;GET MSW OF DIVIDEND FOR SUBTRACT.
017074  160104                  SUB     R1,R4           ;SUBTRACT DIVISOR FROM MSW OF DIVIDEND.
017076  103403                  BCS     2$              ;IF IT DIDN'T GO, WE HAVE QUOTIENT < 16 BITS.
017100  012701  177777          MOV     #-1,R1          ;SET QUOTIENT TO ALL ONES (177777),
017104  000442                  BR      60$             ;EXIT WITH CARRY CLEAR.
                        ;+
                        ; SET UP COUNTERS AND VARIOUS WORKING GPRS.
                        ;-
017106  005004          2$:     CLR     R4              ;CLEAR THE LSW OF THE DIVISOR.
017110  000241                  CLC                     ;CLEAR CARRY FOR THE SHIFT OF THE DIVISOR.
017112  006001                  ROR     R1              ; DIVISOR BY
017114  006004                  ROR     R4              ; 2(UNSIGNED)
017116  012700  000020          MOV     #16.,R0         ;SET UP INITIAL SHIFT COUNT TO 16.
                        ;+
                        ; THE SUBTRACT AND SHIFT LOOP.
                        ;-
017122  010246          4$:     MOV     R2,-(SP)        ;SAVE MSWORD OF DIVIDEND.
017124  010346                  MOV     R3,-(SP)        ;SAVE LSWORD OF DIVIDEND.
017126  160403                  SUB     R4,R3           ;LSWORD DIVIDEND - LSWORD OF DIVISOR.
017130  005602                  SBC     R2              ;MSWORD DIVIDEND - BORROW .
017132  103402                  BCS     6$              ;IF BORROW FROM BORROW SUBTRACT, IT DIDN'T GO.
017134  160102                  SUB     R1,R2           ;MSWORD DIVIDEND - MSWORD OF DIVISOR.
017136  103003                  BCC     8$              ;IF NO BORROW, IT WENT, CARRY IS CLEAR.
                            ;+
                            ; IT DIDN'T GO, SO WE SHIFT A 1 INTO THE QUOTIENT (COMPLEMENTED LATER).
                            ; CARRY IS SET.
                            ;-
017140  012603          6$:     MOV     (SP)+,R3        ;RESTORE LSWORD OF DIVIDEND.
```

```
4572  017142  012602                  MOV     (SP)+,R2        ;RESTORE MSWORD OF DIVIDEND.
4573  017144  000401                  BR      10$             ;GOTO SHIFT 1 INTO THE QUOTIENT.
4574                                  ;+
4575                                  ; IT WENT, SO WE RESTORE THE STACK AND SHIFT A 0 INTO QUOTIENT (WILL BE
4576                                  ; COMPLEMENTED LATER).   CARRY IS CLEAR.
4577                                  ;-
4578  017146  012626                  8$:     MOV     (SP)+,(SP)+     ;POP THE SAVED DIVIDEND OFF OF THE STACK.
4579                                  ;+
4580                                  ; SHIFT THE RESULT OF THE SUBTRACT ATTEMPT INTO THE QUOTIENT SHIFT REG.
4581                                  ;-
4582  017150  006105                  10$:    ROL     R5              ;SHIFT NEXT BIT INTO THE INVERTED QUOTIENT.
4583  017152  000241                          CLC                     ;DIVIDE THE
4584  017154  006001                          ROR     R1              ;  DEVISOR BY
4585  017156  006004                          ROR     R4              ;  2 (UNSIGNED).
4586  017160  005300                          DEC     R0              ;COUNT THIS SHIFT AND SUBTRACT.
4587  017162  001357                          BNE     4$              ;LOOP FOR ANOTHER SHIFT & SUB IF NOT DONE.
4588  017164  005105                          COM     R5              ;GET QUOTIENT FROM INVERTED QUOTIENT.
4589                                  ;+
4590                                  ; NOW WE EITHER ROUND UP OR LEAVE QUOTIENT ALONE.
4591                                  ;-
4592  017166  000241                          CLC                     ;CLEAR THE CARRY FOR THE SHIFT OF THE DIVIDEND.
4593  017170  006103                          ROL     R3              ;MULTIPLY LSWORD OF DIVIDEND BY 2, MSWORD IS 0.
4594  017172  103402                          BCS     12$             ;IF CARRY FROM SHIFT, ROUND UP.
4595  017174  160403                          SUB     R4,R3           ;SUBTRACT DIVISOR FROM DIVIDEND.
4596  017176  103403                          BCS     14$             ;IF BORROW, DON'T ROUND UP.
4597                                  ;+
4598                                  ; ROUND UP, EXTRA SUBTRACT WENT.
4599                                  ;-
4600  017200  005205                  12$:    INC     R5              ;INCREMENT THE QUOTIENT BY ONE.
4601  017202  001001                          BNE     14$             ;IF NO OVERFLOW, WE LEAVE THE ROUND UP.
4602  017204  005305                          DEC     R5              ;DON'T LET ROUNDING CAUSE OVERFLOW.
4603                                  ;+
4604                                  ; ALL DONE, PASS QUOTIENT AND EXIT.
4605                                  ;-
4606  017206  010501                  14$:    MOV     R5,R1           ;PASS QUOTIENT BACK IN R1.
4607  017210  000261                          SEC                     ;INDICATE NO OVERFLOW.
4608
4609  017212                          60$:    PASS    R1              ;RESTORE GPRS, LEAVE THE FOLLOWING INTACT:
4610  017212  010166  000004                                  MOV     R1,R1SLOT(SP)           ;PUT R1 IN STACK SLOT.
4611  017216  004736                                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
4612                                                                   ;R1 - 16 BIT, UNSIGNED QUOTIENT,
4613  017220  000207                          RTS     PC              ;CARRY - SET INDICATES NO OVERFLOW (SUCCESS).
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WAIBIC -
                        ;++ ************************************************************************************
                        ;*                        - WAIT FOR BIT CLEAR ROUTINE -
                        ;*      THIS SUBROUTINE WAITS FOR THE SPECIFIED BIT TO BECOME CLEAR.  IF THE
                        ;*      SPECIFIED BIT GOES TO A CLEAR STATE WITHIN THE SPECIFIED TIME-OUT
                        ;*      PERIOD A SUCCESS INDICATION IS RETURNED BY THIS ROUTINE.
                        ;*      THE LAST VALUE WHICH IS READ LOOKING FOR THE CONDITION IS RETURNED TO
                        ;*      ALLOW THE USE OF THIS ROUTINE TO LOOK FOR DESTRUCTIVE READ CONDITIONS.
                        ;*
                        ;* INPUTS:      R1 - TIME-OUT VALUE AND BIT NUMBER INDICATION:
                        ;*                BITS 15 THRU 12 - NUMBER OF BIT TO TEST (RANGE 0 THRU 15).
                        ;*                BITS 11 THRU 0  - TIME-OUT VALUE IN MILLI-SECONDS (4095 MAX).
                        ;*              R2 - ADDRESS OF WORD CONTAINING THE BIT TO TEST.
                        ;*                   MSLCNT.
                        ;*
                        ;* OUTPUTS:     R2 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
                        ;*              CARRY - SUCCESS FLAG (CARRY SET IF BIT CLR BEFORE TIME-OUT).
                        ;*
                        ;* CALLING SEQUENCE:    MOV     #130040,R1      ;PASS BIT 11 (13 OCTAL) AND
                        ;*                                               ;  32 (40 OCTAL) MS DELAY.
                        ;*                       MOV     #LABEL,R2       ;TEST BIT IN WORD AT "LABEL".
                        ;*                       JSR     PC,WAIBIC       ;WAIT 32 MS FOR BIT 11 TO CLR.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: MSLGET.
                        ;-- ************************************************************************************

017222                  WAIBIC:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
004537  004062                          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017226  010204                  MOV     R2,R4                   ;SET UP THE ADDRESS PARAMETER FOR MSLGET.
017230  010102                  MOV     R1,R2
017232  042701  170000          BIC     #170000,R1              ;SEPERATE DELAY COUNT OUT OF PASSED PARAMETER.
017236  042702  007777          BIC     #7777,R2                ;SEPERATE LINE NUMBER FIELD OF PASSED PARAM.
017242  000302                  SWAB    R2                      ;PUT LINE NUMBER FIELD IN LSBYTE.
017244  006202                  ASR     R2                      ;SHIFT THE LINE NUMBER FIELD INTO THE PROPER
017246  006202                  ASR     R2                      ;  POSITION TO USE IT AS A WORD TABLE OFFSET
017250  006202                  ASR     R2                      ;  FOR THE TABLE LOOKUP OF THE LINE BIT MAP.
017252  016202  002374          MOV     BITTBL(R2),R2           ;GET BIT MAP OF LINE TO TEST FROM TABLE.
017256  005003                  CLR     R3                      ;INDICATE THAT THE BIT SHOULD BE CLR.
017260  004737  015216          JSR     PC,MSLGET               ;WAIT FOR THE BIT TO BE CLR WITHIN TIME-OUT.
                                                                ;  CARRY IS CORRECT UPON MSLGET RETURN.
017264  010002                  MOV     R0,R2                   ;PASS LAST VALUE READ AS OUTPUT PARAMETER.
017266                  60$:    PASS    R2                      ;RESTORE GPRS, EXCEPT THE FOLLOWING:
017266  010266  000006                          MOV     R2,R2SLOT(SP)           ;PUT R2 IN STACK SLOT.
017272  004736                              JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                                ;  R2 - LAST VALUE READ LOOKING FOR CONDITION.
017274  000207                  RTS     PC                      ;  CARRY - SUCCESS FLAG (SET IF BIT FOUND CLR).
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WAIBIS -
                        ;++ ************************************************************************
                        ;*                        - WAIT FOR BIT SET ROUTINE -
                        ;*      THIS SUBROUTINE WAITS FOR THE SPECIFIED BIT TO BECOME SET.  IF THE
                        ;*      SPECIFIED BIT GOES TO A SET STATE WITHIN THE SPECIFIED TIME-OUT
                        ;*      PERIOD A SUCCESS INDICATION IS RETURNED BY THIS ROUTINE.
                        ;*      THE LAST VALUE WHICH IS READ LOOKING FOR THE CONDITION IS RETURNED TO
                        ;*      ALLOW THE USE OF THIS ROUTINE TO LOOK FOR DESTRUCTIVE READ CONDITIONS.
                        ;*
                        ;* INPUTS:      R1 - TIME-OUT VALUE AND BIT NUMBER INDICATION:
                        ;*                BITS 15 THRU 12 - NUMBER OF BIT TO TEST (RANGE 0 THRU 15).
                        ;*                BITS 11 THRU 0  - TIME-OUT VALUE IN MILLI-SECONDS (4095 MAX).
                        ;*              R2 - ADDRESS OF WORD CONTAINING THE BIT TO TEST.
                        ;*                   MSLCNT.
                        ;*
                        ;* OUTPUTS:     R2 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
                        ;*              CARRY - SUCCESS FLAG (CARRY SET IF BIT SET BEFORE TIME-OUT).
                        ;*
                        ;* CALLING SEQUENCE:    MOV     #130040,R1      ;PASS BIT 11 (13 OCTAL) AND
                        ;*                                               ;  32 (40 OCTAL) MS DELAY.
                        ;*                      MOV     #LABEL,R2       ;TEST BIT IN WORD AT 'LABEL'.
                        ;*                      JSR     PC,WAIBIS       ;WAIT 32 MS FOR BIT 11 TO SET.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: MSLGET.
                        ;-- ************************************************************************

017276                  WAIBIS:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017276  004537  004062                      JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017302  010204                  MOV     R2,R4           ;SET UP THE ADDRESS PARAMETER FOR MSLGET.
017304  010102                  MOV     R1,R2
017306  042701  170000          BIC     #170000,R1      ;SEPERATE DELAY COUNT OUT OF PASSED PARAMETER.
017312  042702  007777          BIC     #7777,R2        ;SEPERATE LINE NUMBER FIELD OF PASSED PARAM.
017316  000302                  SWAB    R2              ;PUT LINE NUMBER FIELD IN LSBYTE.
017320  006202                  ASR     R2              ;SHIFT THE LINE NUMBER FIELD INTO THE PROPER
017322  006202                  ASR     R2              ;  POSITION TO USE IT AS A WORD TABLE OFFSET
017324  006202                  ASR     R2              ;  FOR THE TABLE LOOKUP OF THE LINE BIT MAP.
017326  016202  002374          MOV     BITTBL(R2),R2   ;GET BIT MAP OF LINE TO TEST FROM TABLE.
017332  010203                  MOV     R2,R3           ;INDICATE THAT THE BIT SHOULD BE SET.
017334  004737  015216          JSR     PC,MSLGET       ;WAIT FOR THE BIT TO BE SET WITHIN TIME-OUT.
                                                        ;  CARRY IS CORRECT UPON MSLGET RETURN.
017340  010002                  MOV     R0,R2           ;PASS LAST VALUE READ AS OUTPUT PARAMETER.
017342                  60$:    PASS    R2              ;RESTORE GPRS, EXCEPT THE FOLLOWING:
017342  010266  000006                          MOV     R2,R2SLOT(SP)           ;PUT R2 IN STACK SLOT.
017346  004736                              JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
                                                        ;  R2 - LAST VALUE READ LOOKING FOR CONDITION.
017350  000207                  RTS     PC              ;  CARRY - SUCCESS FLAG (SET IF BIT FOUND SET).
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WAITTX -
                        ;++ ************************************************************************
                        ;*                        - WAIT FOR TX TO FINISH -
                        ;*       THIS SUBROUTINE IS USED IN THE FIHAVL.TST.
                        ;*      IT WAITS FOR TRANSMISSION TO COMPLETE IE TX_ACTION. THEN DELAYS
                        ;*      FOR 5 MILLISECONDS TO ALLOW TIME FOR THE LAST CHARACTER TO GET INTO
                        ;*      THE FIFO.
                        ;*
                        ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR.
                        ;*
                        ;* OUTPUTS:     CARRY - SET INDICATES SUCCESS.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,WAITTX
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: DELAY,WAIBIS.
                        ;-- ************************************************************************

017352                  WAITTX:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017352  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017356  012701  170454          MOV     #170454,R1              ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
017362  013702  002246          MOV     CSRA,R2                 ;PASS THE ADDRESS OF THE CSR.
017366  004737  017276          JSR     PC,WAIBIS               ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
017372  103005                  BCC     60$                     ;BRANCH IF FIFO EMPTY, ABORT THE TEST.
017374  012704  000005          MOV     #5,R4                   ;PASS DELAY OF 5 MILLI SECS.
017400  004737  014574          JSR     PC,DELAY                ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
017404  000261                  SEC                             ;SET CARRY TO I8NDICATE SUCCESS.

017406                  60$:    PASS                            ;RESTORE GPRS.
017406  004736                          JSR     PC,@(SP)+       ;RETURN TO PREG05 SUBRT.
                                                                ;PASS THE CARRY BIT, SET INDICATES SUCCESS.
017410  000207                  RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WTWLNC -
                        ;++ ************************************************************************
                        ;*                        - LINE CONTROL REGISTER SETUP ROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO SET THE DEVICE UNDER TEST (DUT) LINE
                        ;*      CONTROL REGISTERS (LNCTRL) TO THE SPECIFIED STATE.  ONLY THE LNCTRLS
                        ;*      FOR THE SPECIFIED LINES ARE ALTERED.
                        ;*
                        ;* INPUTS:      R0 - NEW LINE PARAMETERS.
                        ;*              R5 - BIT MAP OF LINES TO BE ALTERED.
                        ;*              CSRA - CONTAINS ADDRESS OF THE DUT CSR.
                        ;*              IESTAT - CONTAINS THE CURRENT STATE OF THE TX AND RX INTERRUPT
                        ;*               ENABLE BITS IN THE CSR.
                        ;*              LNCTRA - CONTAINS ADDRESS OF THE DUT LNCTRL REGISTERS.
                        ;*
                        ;* OUTPUTS:     LNCTRL - SPECIFIED DUT LINE CONTROL REGISTERS ARE ALTERED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,WTWLNC
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: ALTFLD.
                        ;-- ************************************************************************

017412                  WTWLNC:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017412  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; SET UP THE PARAMETERS FOR THE CALL TO ALTFLD.
                        ;-
017416  013701  002256          MOV     LNCTRA,R1       ;SET UP THE REGISTER ADDRESS PARAMETER.
017422  010002                  MOV     R0,R2           ;SET UP THE DESIRED REGISTER CONTENTS.
017424  010503                  MOV     R5,R3           ;SET UP THE BIT MAP OF LINES TO ALTER.
017426  012704  177777          MOV     #-1,R4          ;SELECT ALL REGISTER BITS TO BE ALTERED.
                        ;+
                        ; CALL THE SUBROUTINE WHICH ALTERS THE REGISTER CONTENTS.
                        ;-
017432  004737  013752          JSR     PC,ALTFLD       ;ALTER THE REGISTER CONTENTS.

017436                  60$:    PASS                    ;RESTORE GPRS.
017436  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017440  000207          RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WTWLNS -
                        ;*****************************************************************************
                        ;*                        - WRITE WORD TO ALL LINES ROUTINE -
                        ;*      THIS SUBROUTINE WRITES A SPECIFIED WORD TO THE SPECIFIED DHV DEVICE
                        ;*      REGISTER FOR EACH OF THE DHV LINES.  IT COULD BE USED TO CLEAR ALL
                        ;*      OF THE LNCTRL REGISTERS OR TO INITIALIZE ALL OF THE LPR REGISTERS TO
                        ;*      THE SAME PARAMETERS.
                        ;*
                        ;* INPUTS:      R1 - ADDRESS OF THE SPECIFIED REGISTERS.
                        ;*              R2 - WORD TO WRITE INTO THE SPECIFIED REGISTERS.
                        ;*              IESTAT - SAVED STATES OF THE TX.IE AND RX.IE BITS.
                        ;*              MAPLNS - EQUATED TO BIT MAP OF LINES ON DEVICE (8 FOR DHV11).
                        ;*                CSRA.
                        ;*
                        ;* OUTPUTS:     DEVICE REGISTERS - SPECIFIED REGISTERS GIVEN NEW VALUE.
                        ;*              CSR IND.ADR.REG FIELD - DESTROYED.
                        ;*              CSR INTERRUPT ENABLE BITS - SET TO STATES IN IESTAT.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,WTWLNS
                        ;*
                        ;* COMMENTS:    NOTE THAT THE SPECIFIED REGISTERS FOR ALL LINES ARE ALTERED
                        ;*              BY THIS ROUTINE.  THIS ROUTINE SHOULD NOT BE USED TO ALTER
                        ;*              THE STATES OF PARTIAL REGISTER FIELDS OR TO ALTER A REGISTER
                        ;*              FOR FEWER THAN ALL OF THE LINES.
                        ;*              THE SPECIFIED REGISTERS ARE READ BEFORE BEING WRITTEN.
                        ;*
                        ;* SUBROUTINES CALLED: ALTFLD.
                        ;*****************************************************************************

017442                  WTWLNS:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017442  004537  004062                 JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; SET UP THE BIT MAP OF LINES TO CHANGE AND MASK OF BITS TO ALTER PARAMETERS.
                        ;-
017446  012703  000377          MOV     #MAPLNS,R3      ;GET THE BIT MAP OF LINES TO CHANGE.
017452  012704  177777          MOV     #-1,R4          ;INDICATE ALL 16 BITS TO BE CHANGED.
                        ;+
                        ; CALL THE SUBROUTINE TO WRITE THE SPECIFIED REGISTERS.
                        ;-
017456  004737  013752          JSR     PC,ALTFLD       ;CHANGE THE REGISTERS.

017462                  60$:    PASS                            ;RESTORE GPRS.
017462  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017464  000207          RTS     PC
```

```
                        .SBTTL  GLOBAL SUBROUTINE                - WTWLPR -
                        ;++ ************************************************************************
                        ;*                        - LINE PARAMETER REGISTER SETUP ROUTINE -
                        ;*      THIS SUBROUTINE IS USED TO SET THE DEVICE UNDER TEST (DUT) LINE
                        ;*      PARAMETER REGISTERS (LPR) TO THE SPECIFIED STATE.  ONLY THE LPRS FOR
                        ;*      THE SPECIFIED LINES ARE ALTERED.
                        ;*
                        ;* INPUTS:      R0 - NEW LINE PARAMETERS.
                        ;*              R5 - BIT MAP OF LINES TO BE ALTERED.
                        ;*              CSRA - CONTAINS ADDRESS OF THE DUT CSR.
                        ;*              IESTAT - CONTAINS THE CURRENT STATE OF THE TX AND RX INTERRUPT
                        ;*               ENABLE BITS IN THE CSR.
                        ;*              LPRA - CONTAINS ADDRESS OF THE DUT LPR.
                        ;*
                        ;* OUTPUTS:     LPR - SPECIFIED DUT LINE PARAMTER REGISTERS ARE ALTERED.
                        ;*
                        ;* CALLING SEQUENCE:    JSR     PC,WTWLPR
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: ALTFLD.
                        ;-- ************************************************************************

017466                  WTWLPR:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017466  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
                        ;+
                        ; SET UP THE PARAMETERS FOR THE CALL TO ALTFLD.
                        ;-
017472  013701  002252          MOV     LPRA,R1         ;SET UP THE REGISTER ADDRESS PARAMETER.
017476  010002                  MOV     R0,R2           ;SET UP THE DESIRED REGISTER CONTENTS.
017500  010503                  MOV     R5,R3           ;SET UP THE BIT MAP OF LINES TO ALTER.
017502  012704  177777          MOV     #-1,R4          ;SELECT ALL REGISTER BITS TO BE ALTERED.
                        ;+
                        ; CALL THE SUBROUTINE WHICH ALTERS THE REGISTER CONTENTS.
                        ;-
017506  004737  013752          JSR     PC,ALTFLD       ;ALTER THE REGISTER CONTENTS.

017512                  60$:    PASS                    ;RESTORE GPRS.
017512  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017514  000207          RTS     PC
```

```
                        .SBTTL  INTERRUPT SERVICE ROUTINE       - CACHRX -
                        ;++ ************************************************************************
                        ;*      - CATCH RECEIVER INTERRUPT.
                        ;*      THIS ROUTINE IS USED IN SEVERAL TESTS,TO LOG A COUNT OF THE
                        ;*      NUMBER OF RECEIVER INTERUPTS THAT OCCUR.
                        ;*
                        ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR.
                        ;*              RXINTC - HOLDS THE COUNT OF THE NUMBER OF RX INTERRUPTS
                        ;*                       THAT OCCURRED.
                        ;*
                        ;* OUTPUTS:     RXINTC - CONTAINS THE UPDATED INTERRUPT COUNT.
                        ;*
                        ;*
                        ;* CALLING SEQUENCE:    PUT THE ADDRESS OF THE LABEL CACHRX IN THE VECTOR
                        ;*                      LOCATION.
                        ;*
                        ;* COMMENTS:
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE
                        ;-- ************************************************************************

017516                  CACHRX::SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
017516  004537  004062          JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017522  013701  002304          MOV     RXINTC,R1       ;GET THE RECEIVER INTERRUPT COUNT
017526  005201                  INC     R1              ;INCREMENT THE COUNT
017530  102001                  BVC     2$              ;BRANCH IF NO OVERFLOW OCCURRED
017532  005301                  DEC     R1              ;RESET THE COUNT TO 177777
017534  010137  002304  2$:     MOV     R1,RXINTC       ;SAVE NEW COUNT VALUE
017540                  60$:    PASS                    ;RESTORE GPRS.
017540  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017542  000002                  RTI
```

```
                                .SBTTL  INTERRUPT SERVICE ROUTINE       - CACHTX -
                                ;++ ******************************************************************************
                                ;*      - CATCH TRANSMITER INTERRUPT.
                                ;*      THIS ROUTINE IS USED IN SEVERAL TESTS,TO LOG A COUNT OF THE
                                ;*      NUMBER OF TRANSMISSION INTERRUPTS THAT OCCUR.
                                ;*
                                ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR.
                                ;*              TXINTC - HOLDS THE COUNT OF THE NUMBER OF TX INTERRUPTS
                                ;*                       THAT OCCURRED.
                                ;*
                                ;* OUTPUTS:     TXINTC - CONTAINS THE UPDATED INTERRUPT COUNT.
                                ;*
                                ;*
                                ;* CALLING SEQUENCE:    PUT THE ADDRESS OF THE LABEL CACHTX IN THE VECTOR
                                ;*                      LOCATION.
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE
                                ;-- ******************************************************************************

017544                          CACHTX::SAVE                    ;SAVE CONTENTS OF GPRS R0 THRU R5.
017544  004537  004062                  JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017550  013701  002310          MOV     TXINTC,R1               ;GET THE TRANSMISSION INTERRUPT COUNT
017554  005201                  INC     R1                      ;INCREMENT THE COUNT
017556  102001                  BVC     2$                      ;BRANCH IF NO OVERFLOW OCCURRED
017560  005301                  DEC     R1                      ;RESET THE COUNT TO 177777
017562  010137  002310  2$:     MOV     R1,TXINTC               ;SAVE NEW COUNT VALUE
017566                  60$:    PASS                            ;RESTORE GPRS.
017566  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017570  000002                  RTI
```

```
                        .SBTTL  INTERRUPT SERVICE ROUTINE       - CLKINT -
                        ;++ ************************************************************************
                        ;*      THIS ROUTINE IS EXECUTED CLKHRZ TIMES PER SECOND.  IT DECREMENTS THE
                        ;*      TWO TIMER COUNTERS DOWN TO ZERO.
                        ;*
                        ;* INPUTS:      TIMER1 - TIMER COUNTER #1.
                        ;*              TIMER2 - TIMER COUNTER #2.
                        ;*              TIMER3 - TIMER COUNTER FOR CALL OF BREAK MACRO.
                        ;*
                        ;* OUTPUTS:     THE 2 TIMER COUNTERS ARE DECREMENTED IF THEY ARE NOT ZERO.
                        ;*
                        ;* CALLING SEQUENCE:    PUT #CLKINT IN THE CLOCK INTERRUPT VECTOR SLOT.
                        ;*                      PUT THE DESIRED TIME PERIOD (SECONDS TIMES CLKHRZ) IN
                        ;*                      EITHER TIMER1 OR TIMER2 AND POLL THE RESPECTIVE TIMER
                        ;*                      COUNTER TO DETECT ITS GOING TO 0 ON TIME-OUT.
                        ;*
                        ;* COMMENTS:    THE 2 COUNTERS WILL NOT WRAPAROUND BUT WILL STOP AT 0.  THIS
                        ;*              ALLOWS THE DETECTION OF A TIME-OUT ANY TIME AFTER THE TIME-OUT
                        ;*              HAS OCCURRED UNTIL THE TIMER COUNTER IS SET TO ANOTHER VALUE.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

017572  005737  002332  CLKINT:: TST     TIMER1          ;CHECK FOR TIMER1 AT ZERO.
017576  001402                  BEQ     2$              ;BRANCH TO LEAVE IT AT ZERO IF IT IS ZERO.
017600  005337  002332          DEC     TIMER1          ;DECREMENT TIME COUNT.
017604  005737  002334  2$:     TST     TIMER2          ;CHECK FOR TIMER2 AT ZERO.
017610  001402                  BEQ     4$              ;BRANCH TO LEAVE IT ALONE IF IT'S ALREADY ZERO.
017612  005337  002334          DEC     TIMER2          ;DECREMENT TIME COUNT.
017616  005337  002336  4$:     DEC     TIMER3          ;DECREMENT THE BREAK COUNT.
017622  001006                  BNE     60$             ;EXIT IF NOT TIME TO CALL BREAK.
017624  013737  002340  002336  MOV     BCOUNT,TIMER3   ;SET UP TIME TILL NEXT BREAK.
017632  010046                  MOV     R0,-(SP)        ;SAVE CONTENTS OF R0 FROM BREAK MACRO.
017634                          BREAK                   ;CHECK FOR OPERATOR CONTROL/C.
017634  104422                                                  TRAP    C$BRK
017636  012600                  MOV     (SP)+,R0        ;RESTORE CONTENTS OF R0.
017640  000002          60$:    RTI
```

```
                                .SBTTL  INTERUPT SERVICE ROUTINE        - RXBRRT -
                                ;++ ************************************************************************
                                ;*      - BR LEVEL TEST RECEIVE INTERRUPT SERVICE ROUTINE -
                                ;*      THIS SERVICE ROUTINE HANDLES RECEIVE INTERRUPTS DURING THE INTERRUPT
                                ;*      BR LEVEL TEST.  THIS ROUTINE COUNTS THE INTERRUPT AND SETS A FLAG
                                ;*      TO INDICATE THAT THE INTERRUPT HAS OCCURRED.  IT ALSO CHECKS THE
                                ;*      FLAG WHICH INDICATES THAT A TX INTERRUPT HAS OCCURRED.  IF THE TX
                                ;*      INTERRUPT FLAG IS SET, THIS ROUTINE SETS AN INTERRUPT ORDER ERROR
                                ;*      FLAG INDICATING THAT A TRANSMIT INTERRUPT WAS SERVICED BEFORE A
                                ;*      SIMULTANEOUS RECEIVE INTERRUPT.
                                ;*
                                ;* INPUTS:      RXINTC - HOLDS THE COUNT OF THE NUMBER OF RX INTERUPTS.
                                ;*              RXINTF - RX INTERRUPT FLAGS.
                                ;*
                                ;* OUTPUTS:     RXINTC - CONTAINS THE UPDATED INTERUPT COUNT.
                                ;*              RXINTF - RX INT FLAGS:
                                ;*                       (BIT 0 SET, BIT 14 SET IF TXINTF BIT 0 IS SET.)
                                ;*
                                ;* CALLING SEQUENCE:    PUT THE ADDRESS OF THE LABEL RXBRRT IN THE VECTOR
                                ;*                      LOCATION.
                                ;*
                                ;* COMMENTS:    NOTE: THE FIFO IS PURGED BY THIS ROUTINE.
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE.
                                ;-- ************************************************************************

017642                          RXBRRT:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017642  004537  004062                         JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
017646  017700  162376                 MOV     @RBUFA,R0               ;READ THE CHAR OUT OF THE FIFO.
017652  013701  002304                 MOV     RXINTC,R1               ;GET THE INTERUPT COUNT.
017656  005201                         INC     R1                      ;INCREMENT THE COUNT.
017660  001402                        BEQ     2$                      ;BYPASS UPDATING COUNT IF OVERFLOW OCCURRED.
017662  010137  002304                 MOV     R1,RXINTC               ;SAVE NEW COUNT VALUE.
017666  013701  002306                2$:     MOV     RXINTF,R1       ;GET THE RX INTERRUPT FLAGS.
017672  052701  000001                 BIS     #BIT0,R1                ;SET THE RX INTERRUPT HAS OCCURRED FLAG.
017676  032737  000001  002312         BIT     #BIT0,TXINTF            ;TEST THE "TX INT HAS OCCURRED" FLAG.
017704  001402                        BEQ     4$                      ;SKIP SETTING ERROR FLAG IF NO TX INT.
017706  052701  040000                 BIS     #BIT14,R1               ;SET THE INTERRUPT ORDER ERROR FLAG.
                                ;+
                                ; 8 FIFO CODES WILL CAUSE 8 INTERRUPTS, AFTER THESE 8 CODES WE DON'T WANT
                                ;  TO CHECK THE INTERRUPT ORDER, BECAUSE PERHAPS A BMP CODE HAS COME IN
                                ;  BETWEEN THE SERVICING OF THE 8 FIFO CODE INTERRUPTS AND THE SERVICING
                                ;  OF ONE OF THE TX INTERRUPTS.
                                ;-
017712  023737  002304  000010  4$:      CMP     RXINTC,NUMLNS   ;TEST FOR ALL SELFTEST CODE INTS DONE.
017720  003002                        BGT     60$             ;SKIP UPDATING RX INT FLAGS IF EXTRA RX INTS.
017722  010137  002306                 MOV     R1,RXINTF       ;UPDATE THE RX INTERRUPT FLAGS.
017726                          60$:    PASS                    ;RESTORE GPRS.
017726  004736                                 JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
017730  000002                        RTI
```

```
                        .SBTTL  INTERUPT SERVICE ROUTINE        - RXINPT -
                        ;++ ************************************************************************
                        ;*      - RECEIVE CHARACTER INPUT INTERRUPT SERVICE ROUTINE -
                        ;*      THIS SERVICE ROUTINE INPUTS A CHARACTER FROM THE DUT AND LOADS THE
                        ;*      CHAR (COMPLETE WITH STATUS FLAGS) INTO A RECEIVE CHAR BUFFER IN
                        ;*      MEMORY.  THE INTERRUPT IS ALSO COUNTED.  THE RECEIVE CHAR BUFFER IS
                        ;*      MONITORED TO ENSURE THAT IT DOES NOT OVERFLOW.
                        ;*
                        ;* INPUTS:      BUFEND - LABELS THE END OF THE HOST MEMORY BUFFER.
                        ;*              BUFPTR - CONTAINS ADDRESS OF NEXT FREE BUFFER LOCATION.
                        ;*              CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
                        ;*              RBUFA - CONTAINS THE ADDRESS OF THE RBUF DUT REGISTER.
                        ;*              RXINTC - HOLDS THE COUNT OF THE NUMBER OF RX INTERUPTS.
                        ;*              RXINTF - RX INTERRUPT FLAGS.
                        ;*
                        ;* OUTPUTS:     BUFPTR - CONTAINS UPDATED ADDRESS OF NEXT FREE BUFFER LOCATION.
                        ;*              RXINTC - CONTAINS THE UPDATED INTERUPT COUNT.
                        ;*              RXINTF - RX INT FLAGS (BIT 15 SET IF RX.DATA.AVAIL IS CLEAR).
                        ;*
                        ;* CALLING SEQUENCE:    PUT THE ADDRESS OF THE LABEL RXINPT IN THE VECTOR
                        ;*                      LOCATION.
                        ;*
                        ;* COMMENTS:    IN CASE OF OVERFLOW OF THE MEMORY BUFFER, BUFPTR WILL BE
                        ;*              MAINTAINED EQUAL TO BUFEND AND THE WORD AT BURFPTR WILL BE
                        ;*              THE LAST WORD READ FROM THE DUT FIFO.
                        ;*              NOTE: THIS ROUTINE CAN DESTROY TX.ACTIONS BY READING THE CSR.
                        ;*
                        ;* SUBORDINATE ROUTINES CALLED: NONE.
                        ;-- ************************************************************************

017732                  RXINPT:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
017732  004537  004062          JSR     R5,PREG05               ;CALL REGISTER SAVE SUBRT.
017736  032777  000200  162302          BIT     #BIT7,@CSRA     ;TEST RX.DATA.AVAIL BIT OF THE CSR (READS CSR).
017744  001003                  BNE     2$                      ;BRANCH AROUND SETTING FLAG IF BIT IS SET.
017746  052737  100000  002306          BIS     #BIT15,RXINTF   ;SET THE RX.DATA.AVAIL CLEAR FLAG.
017754  013701  002304          2$:     MOV     RXINTC,R1       ;GET THE INTERUPT COUNT.
017760  005201                  INC     R1                      ;INCREMENT THE COUNT.
017762  001402                  BEQ     4$                      ;BYPASS UPDATING COUNT IF OVERFLOW OCCURRED.
017764  010137  002304          MOV     R1,RXINTC               ;SAVE NEW COUNT VALUE.
017770  013702  002266          4$:     MOV     BUFPTR,R2       ;GET THE POINTER TO NEXT FREE BUFFER WORD.
017774  017722  162250          MOV     @RBUFA,(R2)+            ;READ A CHAR FROM THE FIFO INTO BUFFER.
020000  020237  003712          CMP     R2,BUFEND               ;TEST FOR POINTER BEYOND END OF BUFFER.
020004  103002                  BHIS    60$                     ;SKIP THE PTR UPDATE IF PTR OUT OF BOUNDS.
020006  010237  002266          MOV     R2,BUFPTR               ;UPDATE THE BUFFER POINTER.
020012                  60$:    PASS                            ;RESTORE GPRS.
020012  004736                          JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
020014  000002                  RTI
```

```
                                .SBTTL  GLOBAL TRAP SERVICE ROUTINE     - TP4RTN -
                                ;*********************************************************************************
                                ;*      BUS TIME-OUT TRAP (004 TRAP) SERVICE ROUTINE -
                                ;*      THIS ROUTINE IS USED DURING THE DEVICE REGISTER ADDRESS ACCESS TEST.
                                ;*      IT DETERMINES IF THE 004 TRAP WAS CAUSED BY AN "EXPECTED" ERROR OR
                                ;*      NOT BY EXAMINING THE RETURN PC VALUE ON THE STACK.  IF THE TRAP IS
                                ;*      UNEXPECTED, THIS ROUTINE JUMPS TO THE NORMAL DIAGNOSTIC SUPERVISOR
                                ;*      004 TRAP HANDLING ROUTINE.
                                ;*
                                ;* INPUTS:      SP - POINTS TO THE PC WHERE THE TRAP OCCURED.
                                ;*              ADRPTR - LABEL AT THE ADDRESS WHERE "EXPECTED" TRAPS OCCUR.
                                ;*              TP4FLG - 004 TRAP FLAGS.
                                ;*
                                ;* OUTPUTS:     TP4FLG - BIT 15 IS SET IF "EXPECTED" TRAP OCCURED.
                                ;*
                                ;* CALLING SEQUENCE:    PUT ADDRESS POINTED TO BY TP4RTN IN 004 VECTOR.
                                ;*                      OCCURENCE OF 004 TRAP VECTORS TO THIS ROUTINE.
                                ;*
                                ;* COMMENTS:    ANY 004 TRAP WHICH OCCURS AT AN ADDRESS OTHER THAN THAT LABELED
                                ;*              ADRPTR WILL BE HANDLED BY THE NORMAL 004 TRAP SERVICE ROUTINE.
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE.
                                ;*********************************************************************************

020016  021627  014442          TP4RTN:: CMP    (SP),#ADRPTR    ;COMPARE EXPECTED ADR AGAINST TRAP RET PC.
020022  001402                          BEQ     2$              ;IF THEY MATCH, CONTINUE THIS ROUTINE.
020024  000177  162264                  JMP     @TP4VEC         ;IF NOT,JUMP TO NORMAL 004 TRAP SERVICE RTN.
020030  052737  100000  002316  2$:     BIS     #BIT15,TP4FLG   ;SET THE 004 TRAP OCCURED FLAG.
020036  000002                          RTI                     ;ALL DONE, GO BACK TO THE TEST.
```

```
                                .SBTTL  INTERUPT SERVICE ROUTINE        - TXINTR -
                                ;++ ************************************************************************
                                ;*      - TRANSMIT INTERRUPT SERVICE ROUTINE -
                                ;*      THIS ROUTINE HANDLES A TRANSMIT INTERRUPT FROM THE DEVICE UNDER TEST
                                ;*      (DUT) BY COUNTING THE INTERRUPT AND READING THE DUT CSR TO CLEAR THE
                                ;*      INTERRUPT REQUEST.  THIS ROUTINE ALSO SETS A FLAG TO INDICATE THAT
                                ;*      A TX INTERRUPT HAS OCCURRED AND SETS A FLAG IF THE TX.ACTION BIT IS
                                ;*      NOT SET IN THE READ CONTENTS OF THE DUT CSR.
                                ;*
                                ;* INPUTS:      CSRA - CONTAINS THE ADDRESS OF THE CSR.
                                ;*              TXINTC - HOLDS THE COUNT OF THE NUMBER OF TX INTERUPTS.
                                ;*              TXINTF - TX INTERRUPT FLAGS.
                                ;*
                                ;* OUTPUTS:     TXINTC - CONTAINS THE UPDATED TX INTERUPT COUNT.
                                ;*              TXINTF - TX INT FLAGS (BIT 0 SET, BIT 15 SET IF TX.ACTION CLR).
                                ;*
                                ;* CALLING SEQUENCE:    PUT THE ADDRESS OF THE LABEL TXINTR IN THE VECTOR
                                ;*                      LOCATION.
                                ;*
                                ;* COMMENTS:
                                ;*
                                ;* SUBORDINATE ROUTINES CALLED: NONE
                                ;-- ************************************************************************

020040                          TXINTR:: SAVE                           ;SAVE CONTENTS OF GPRS R0 THRU R5.
020040  004537  004062                         JSR     R5,PREG05       ;CALL REGISTER SAVE SUBRT.
020044  013701  002310                 MOV     TXINTC,R1               ;GET THE TX INTERUPT COUNT.
020050  005201                         INC     R1                      ;INCREMENT THE COUNT.
020052  102001                        BVC     2$                      ;BRANCH IF NO OVERFLOW OCCURRED.
020054  005301                         DEC     R1                      ;RESET THE COUNT TO 177777.
020056  010137  002310         2$:     MOV     R1,TXINTC               ;SAVE NEW COUNT VALUE.
020062  013703  002312                 MOV     TXINTF,R3               ;GET THE TX INTERRUPT FLAGS.
020066  017702  162154                 MOV     @CSRA,R2                ;READ THE CSR.
020072  100402                        BMI     4$                      ;SKIP SETTING OF FLAG IF TX.ACTION IS SET.
020074  052703  100000                 BIS     #BIT15,R3               ;SET THE TX.ACTION CLEAR FLAG.
020100  052703  000001         4$:     BIS     #BIT0,R3                ;SET THE TX INT HAS OCCURRED FLAG.
020104  010337  002312                 MOV     R3,TXINTF               ;UPDATE THE TX INTERRUPT FLAGS.
020110                          60$:    PASS                            ;RESTORE GPRS.
020110  004736                                         JSR     PC,@(SP)+               ;RETURN TO PREG05 SUBRT.
020112  000002                         RTI
```

```
                        .SBTTL  REPORT CODING SECTION

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

020114                          BGNRPT
020114                                                          L$RPT::

020114                          EXIT    RPT
020114  000167                                                          .WORD   J$JMP
020116  000000                                                          .WORD   L10014-2-.

                        .EVEN

020120                          ENDRPT
020120                                                          L10014:
020120  104425                                                          TRAP    C$RPT
```

```
                                .SBTTL  PROTECTION TABLE


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

                                        BGNPROT
                                                                        L$PROT::

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

                                        ENDPROT
```

```
                        .SBTTL  INITIALIZE SECTION
                        ;++
                        ;*****************************************************************************
                        ;*      THIS SECTION CONTAINS THE CODE WHICH IS PERFORMED AT THE BEGINNING OF
                        ;*      EACH PASS OR AFTER A CONTINUE COMMAND.
                        ;*      THIS CODE PERFORMS THE FOLLOWING ACTIONS:
                        ;*
                        ;*      MOVES THE INFORMATION HELD IN THE HARDWARE P-TABLE INTO THE GLOBAL
                        ;*      DATA AREA.
                        ;*
                        ;*****************************************************************************
                        ;--
020130                          BGNINIT
020130                                                          L$INIT::

                        ;SEE IF PROGRAM JUST STARTED, BR IF YES
                                READEF  #EF.START
020130  012700  000040                                                  MOV     #EF.START,R0
020134  104447                                                          TRAP    C$REFG
020136                          BCOMPLETE       NEWSTA
020136  103416                                                          BCS     NEWSTA

                        ;SEE IF PROGRAM JUST RESTARTED, BR IF YES
                                READEF  #EF.RESTART
020140  012700  000037                                                  MOV     #EF.RESTART,R0
020144  104447                                                          TRAP    C$REFG
020146                          BCOMPLETE       NEWRES
020146  103556                                                          BCS     NEWRES

                        ;SEE IF THIS IS A NEW PASS, BR IF YES
                                READEF  #EF.NEW
020150  012700  000035                                                  MOV     #EF.NEW,R0
020154  104447                                                          TRAP    C$REFG
020156                          BCOMPLETE       NEWPAS
020156  103555                                                          BCS     NEWPAS

                        ;SEE IF PROGRAM WAS JUST CONTINUED
                                READEF  #EF.CONTINUE
020160  012700  000036                                                  MOV     #EF.CONTINUE,R0
020164  104447                                                          TRAP    C$REFG
020166                          BNCOMPLETE      GETPRM
020166  103161                                                          BCC     GETPRM
020170  000137  020752          JMP     ENDIT
020174                  NEWSTA:
                                BRESET                  ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
020174  104433                                                          TRAP    C$RESET

                        ;+
                        ; SET UP FOR LINE TIME CLOCK INTERRUPTS.
                        ;-
                                CLOCK   L,R1            ;GET THE CLOCK PARAMETERS.
020176  012700  000114                                                  MOV     #'L,R0
020202  104462                                                          TRAP    C$CLCK
020204  010001                                                          MOV     R0,R1
020206  012137  002322          MOV     (R1)+,CLKCSR            ;STORE CLOCK CSR ADDRESS.
020212  012137  002324          MOV     (R1)+,CLKBRL            ;STORE CLOCK BUS REQ INT LEVEL.
020216  012137  002326          MOV     (R1)+,CLKVEC            ;STORE CLOCK INTERRUPT VECTOR.
020222  012137  002330          MOV     (R1)+,CLKHRZ            ;STORE CLOCK FREQUENCY.
020226  023727  002330  000062  CMP     CLKHRZ,#50.             ;TEST FOR 50HZ LINE FREQUENCY.
```

```
5226  020234  001004                          BNE     2$                      ;BRANCH IF CLOCK IS NOT 50HZ.
5227  020236  012737  000024  002342          MOV     #20.,MSTICK             ;INDICATE 20MS PER CLOCK TICK.
5228  020244  000403                          BR      4$
5229  020246  012737  000021  002342  2$:     MOV     #17.,MSTICK             ;INDICATE 17 MS PER CLOCK TICK.
5230  020254                          4$:     SETVEC  CLKVEC,#CLKINT,PRI06    ;INITIALIZE CLOCK INTERRUPT VECTOR.
5231  020254  013746  000300                                                          MOV     PRI06,-(SP)
5232  020260  012746  017572                                                          MOV     #CLKINT,-(SP)
5233  020264  013746  002326                                                          MOV     CLKVEC,-(SP)
5234  020270  012746  000003                                                          MOV     #3,-(SP)
5235  020274  104437                                                                 TRAP    C$SVEC
5236  020276  062706  000010                                                          ADD     #10,SP
5237  020302  013700  002330                          MOV     CLKHRZ,R0               ;INITIALIZE THE BREAK COUNT
5238  020306  006300                          ASL     R0                      ; TO CAUSE A BREAK
5239  020310  010037  002340                          MOV     R0,BCOUNT               ; EVERY 2 SECONDS.
5240  020314                          SETPRI  #PRI05                  ;ALLOW CLOCK INTERRUPTS DISABLE OTHERS.
5241  020314  012700  000240                                                          MOV     #PRI05,R0
5242  020320  104441                                                                 TRAP    C$SPRI
5243                                  ;+
5244                                  ; ENABLE THE LINE TIME CLOCK (LTC) CHECKING TO MAKE SURE THAT THE CSR
5245                                  ; IS ACCESSABLE.
5246                                  ;    FIRST SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:
5247                                  ;-
5248  020322  013737  000004  002314          MOV     4,TP4VEC        ;SAVE THE EXISTING 004 TRAP VECTOR.
5249  020330  012737  020016  000004          MOV     #TP4RTN,4       ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
5250                                  ;+
5251                                  ;    ENABLE LTC CHECKING FOR 004 TRAP IN CASE CSR IS NOT THERE.
5252                                  ;-
5253  020336  005037  002316                  CLR     TP4FLG          ;CLEAR THE 004 TRAP FLAG.
5254  020342  012737  000100  002320          MOV     #BIT6,WORD1     ;SET UP TO SET BIT6 OF THE LTC CSR.
5255  020350  012700  002320                  MOV     #WORD1,R0       ;SET UP WORD1 AS THE CKTRAP MOVE SOURCE.
5256  020354  013701  002322                  MOV     CLKCSR,R1       ;SET UP LTC CSR AS DESTINATION FOR CKTRAP MOVE.
5257  020360  004737  014430                  JSR     PC,CKTRAP       ;MOVE AND CHECK FOR TRAP.
5258  020364  013737  002314  000004          MOV     TP4VEC,4        ;RESTORE THE NORMAL 004 TRAP VECTOR.
5259  020372  103403                          BCS     6$              ;IF NO TRAP, LTC IS THERE SO CONTINUE.
5260  020374  005037  002330                  CLR     CLKHRZ          ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
5261  020400  000402                          BR      8$              ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
5262                                  ;+
5263                                  ; CALIBRATE THE DELAY ROUTINE MILLI-SECOND DELAY COUNT VALUE.
5264                                  ;-
5265  020402  004737  014134          6$:     JSR     PC,CALMSL
5266                                  ;+
5267                                  ; CHECK FOR MEMMORY MANAGEMENT PRESENT ON THIS MACHINE.
5268                                  ; IF MEM MGT IS PRESENT, DISABLE IT.
5269                                  ;-
5270  020406  013737  000004  002314  8$:     MOV     4,TP4VEC        ;SAVE THE EXISTING 004 TRAP VECTOR.
5271  020414  012737  020016  000004          MOV     #TP4RTN,4       ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
5272  020422  005037  002316                  CLR     TP4FLG          ;CLEAR THE 004 TRAP FLAG.
5273  020426  005037  002320                  CLR     WORD1           ;PREPARE TO CLEAR THE MEM MGT SR0 REGISTER.
5274  020432  012700  002320                  MOV     #WORD1,R0       ;SELECT CLEARED WORD AS CKTRAP RTN SOURCE.
5275  020436  013701  002346                  MOV     MMSR0,R1        ;SELECT MEM MGT SR0 REGISTER AS DESTINATION.
5276  020442  005037  002350                  CLR     MMPRES          ;INDICATE NO MEM MGT PRESENT IN CASE IT ISN'T.
5277  020446  005037  002352                  CLR     MMENAB          ;INDICATE MEM MGT IS NOT ENABLED.
5278  020452  004737  014430                  JSR     PC,CKTRAP       ;CLEAR THE MEM MCT SR0 REG AND CHECK FOR TRAP.
5279  020456  013737  002314  000004          MOV     TP4VEC,4        ;RESTORE THE NORMAL 004 TRAP VECTOR.
5280  020464  103003                          BCC     10$             ;SKIP INDICATING MEM MGT PRESENT IF IT ISN'T.
5281  020466  012737  000001  002350          MOV     #1,MMPRES       ;INDICATE THAT MEM MGT IS PRESENT.
```

```
5282  020474  005037  002302          10$:    CLR     PASCNT          ;CLR COUNTER USED IN REPORTING ROM VERSION #.
5283  020500  000137  020512                  JMP     NEWPAS          ;SKIP AROUND THE BUS RESET, IT'S BEEN DONE.
5284
5285  020504                          NEWRES: BRESET                  ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
5286  020504  104433                                                                  TRAP    C$RESET
5287  020506  005037  002302                  CLR     PASCNT          ;CLR COUNTER USED IN REPORTING ROM VERSION #.
5288  020512                          NEWPAS:
5289  020512  012737  177777  002244          MOV     #-1,UNITN       ;RESET LOGICAL DEVICE TO -1
5290                                  ;+
5291                                  ; INCREMENT THE PASS COUNTER, CORRECT FOR ANY OVERFLOW.
5292                                  ;  THIS COUNTER IS USED IN THE ROM VERSION TEST.
5293                                  ;-
5294  020520  005237  002302           INC     PASCNT          ;INCREMENT THE PASS COUNTER.
5295  020524  001002                  BNE     GETPRM          ;BRANCH IF WE HAVE NOT YET! OVERFLOWED.
5296  020526  005337  002302           DEC     PASCNT          ;SET PASS COUNT TO 177777 OCTAL.
5297
5298                                  ; GET THE HARDWARE PARAMETERS FOR THIS UNIT.
5299  020532                          GETPRM:
5300  020532  005237  002244          INC     UNITN           ;INCREMENT LOGICAL DEVICE NUMBER
5301  020536  023737  002244  002012  CMP     UNITN,L$UNIT    ;SEE IF MAXIMUM UNIT NO. EXCEEDED
5302  020544  002362                  BGE     NEWPAS          ;BR IF YES
5303
5304  020546                          GPHARD  UNITN,R1         ;GET P-TABLE POINTER INTO R1
5305  020546  013700  002244                                                  MOV     UNITN,R0
5306  020552  104442                                                          TRAP    C$GPHRD
5307  020554  010001                                                          MOV     R0,R1
5308  020556                          BCOMPLETE       30$     ;BR IF DEVICE AVAILABLE
5309  020556  103401                                                          BCS     30$
5310  020560  000764                  BR      GETPRM          ;SKIP THIS DEVICE
5311
5312
5313                                  ;************ HARDWARE PARAMETER MOVING CODE *****************
5314  020562  012137  002246  30$:    MOV     (R1)+,CSRA      ;STORE DHV-11 CSR ADDRESS IN DEV.REG.ADDRESS TABLE
5315  020566  012102                  MOV     (R1)+,R2        ;GET THE RX INTERRUPT VECTOR ADDRESS.
5316  020570  010237  002234          MOV     R2,RXVECA       ;STORE RX INT VECTOR ADDRESS.
5317  020574  062702  000004          ADD     #4,R2           ;CALCULATE TX INTERRUPT VECTOR ADDRESS.
5318  020600  010237  002236          MOV     R2,TXVECA       ;STORE TX INT VECTOR ADDRESS.
5319  020604  012137  002240          MOV     (R1)+,ACTLNS    ;STORE DHV-11 ACTIVE LINE BIT MAP
5320  020610  012702  000377          MOV     #MAPLNS,R2      ;GET THE BIT MAP FOR ALL LINES.
5321  020614  005102                  COM     R2              ;GET A BIT MAP OF NON-EXISTANT LINES.
5322  020616  040237  002240          BIC     R2,ACTLNS       ;CLEAR NON-EXISTANT LINES FROM ACTLNS.
5323  020622  112137  002242          MOVB    (R1)+,LOPBCK    ;STORE DHV-11 LOOPBACK MODE
5324  020626  112137  002243          MOVB    (R1)+,BRLEVL    ;STORE DHV-11 INTERUPT BUS REQUEST LEVEL
5325                                  ;+
5326                                  ;
5327                                  ;  CALCULTATE DEVICE REGISTER ADDRESSES,AND PUT THEM IN THE
5328                                  ;  DEVICE REGISTER ADDRESS TABLE.
5329  020632  013701  002246          ;-
5330  020636  005201                  MOV     CSRA,R1         ;COPY CSR ADDRESS
5331  020640  005201                  INC     R1              ;INCREMENT CSR ADDRESS
5332  020642  012703  000007          INC     R1              ;  COPY BY 2.
5333  020646  012702  002250          MOV     #7,R3           ;SET UP REGISTER COUNT
5334  020652  010122          12$:    MOV     #RBUFA,R2       ;GET LOCATION WHERE RBUF ADDRESS GOES IN TABLE
5335  020654  005201                  MOV     R1,(R2)+        ;STORE REGISTER ADDRESS IN TABLE
5336  020656  005201                  INC     R1              ;INCREMENT REGISTER ADDRESS
5337  020660  005303                  INC     R1              ;  BY 2,FOR THE NEXT DEVICE REGISTER.
                                      DEC     R3              ;DECREMENT REGISTER COUNT
```

```
5338  020662  001373                        BNE     12$             ;LOOP IF NOT DONE
5339
5340                                  ;+
5341                                  ; INITIALISE THE BMP CODE QUEUE.
5342                                  ;-
5343  020664  012700  002452                 MOV     #BMPCQB,R0      ;GET THE START ADDRESS OF THE QUEUE.
5344  020670  012701  002652                 MOV     #BMPCQE,R1      ;GET THE END ADDRESS OF THE QUEUE.
5345  020674  010037  002450                 MOV     R0,BMPCQP       ;SET THE POINTER TO THE START OF THE QUEUE.
5346  020700  005020                  14$:    CLR     (R0)+           ;CLEAR OUT THE CONTENTS OF THE QUEUE.
5347  020702  020001                          CMP     R0,R1           ;CHECK IF END OF QUEUE HAS BEEN REACHED.
5348  020704  103775                        BLO     14$             ;LOOP IF NOT ALL DONE.
5349
5350                                  ;+
5351                                  ; REPORT THE UNIT NUMBER IF THE SOFTWARE P-TABLE QUESTION WAS ANSWERED YES,
5352                                  ;  AND THE MAXIMUM UNIT NUMBER IS GREATER THAN 1.
5353  020706  032737  000020  002230  ;-        BIT     #BIT4,OPTION    ;CHECK IF THE QUESTION WAS ANSWERED YES.
5354  020714  001416                        BEQ     16$             ;SKIP REPORTING UNIT NUMBER IF IT IS DISABLED.
5355  020716  023727  002012  000001          CMP     L$UNIT,#1       ;CHECK MAXIMUM NUMBER OF UNITS SELECTED.
5356  020724  003412                        BLE     16$             ;DO NOT REPORT UNIT NUMBER IF MAX NUMBER < 1.
5357  020726                                PRINTF  #MFUNIT,UNITN   ;REPORT UNIT NUMBER.
5358  020726  013746  002244                                                          MOV     UNITN,-(SP)
5359  020732  012746  004162                                                          MOV     #MFUNIT,-(SP)
5360  020736  012746  000002                                                          MOV     #2,-(SP)
5361  020742  010600                                                                  MOV     SP,R0
5362  020744  104417                                                                  TRAP    C$PNTF
5363  020746  062706  000006                                                          ADD     #6,SP
5364  020752                        16$:
5365
5366  020752  005037  002270  ENDIT: CLR     CTRLCF          ;CLR THE CTRL-C TEST ABORT FLAG.
5367                                  ;+
5368                                  ; SET THE PROCESSOR PRIORITY TO ALLOW LTC INTERRUPTS BUT NOT OTHERS.
5369                                  ;-
5370  020756                                SETPRI  #PRI07                  ;SET PROCESSOR PRIORITY TO 7.
5371  020756  012700  000340                                                          MOV     #PRI07,R0
5372  020762  104441                                                                  TRAP    C$SPRI
5373  020764                                ENDINIT
5374  020764                                                          L10016:
5375  020764  104411                                                                  TRAP    C$INIT
```

```
                                .SBTTL  AUTODROP SECTION


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

020766                                  BGNAUTO
020766                                                          L$AUTO::

020766                                  ENDAUTO
020766                                                          L10017:
020766  104461                                                          TRAP    C$AUTO
```

```
5395
5396
5397                                    .SBTTL  CLEANUP CODING SECTION
5398
5399                                    ;++
5400                                    ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
5401                                    ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
5402                                    ;--
5403
5404  020770                                    BGNCLN
5405  020770                                                                          L$CLEAN::
5406
5407  020770  005737  002270                     TST     CTRLCF          ;DID WE GET HERE BY CTRL-C FROM TEST?
5408  020774  001401                            BEQ     2$              ;CTRL-C FROM TEST?  NO, SKIP BUS RESET.
5409  020776                                    BRESET                  ;YES, CLR ANY DMAS OR OUTSTANDING INTERRUPTS.
5410  020776  104433                                                                  TRAP    C$RESET
5411  021000                                2$:
5412
5413
5414  021000                                    EXIT    CLN
5415  021000  104432                                                                  TRAP    C$EXIT
5416  021002  000002                                                                  .WORD   L10020-.
5417
5418
5419                                    .EVEN
5420
5421  021004                                    ENDCLN
5422  021004                                                                  L10020:
5423  021004  104412                                                                  TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION

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

021006                                  BGNDU
021006                                                                  L$DU::
021006                                   PRINTF #DROP,R0        ;REPORT UNIT THAT HAS BEEN DROPPED.
021006  010046                                                         MOV     R0,-(SP)
021010  012746  021032                                                 MOV     #DROP,-(SP)
021014  012746  000002                                                 MOV     #2,-(SP)
021020  010600                                                         MOV     SP,R0
021022  104417                                                         TRAP    C$PNTF
021024  062706  000006                                                 ADD     #6,SP
021030  000427                           BR     EDROP          ;BRANCH AROUND THE MESSAGE.

021032  040445  052440  044516  DROP:    .ASCIZ/%A UNIT%D6%A DROPPED FROM FURTHER TESTING.%N/
021040  022524  033104  040445
021046  042040  047522  050120
021054  042105  043040  047522
021062  020115  052506  052122
021070  042510  020122  042524
021076  052123  047111  027107
021104  047045     000
        021110                  .EVEN
021110                  EDROP:

                                EXIT    DU

021110                                                          .WORD   J$JMP
021110  000167                                                  .WORD   L10021-2-.
021112  000000

021114                                  ENDDU
021114                                                  L10021:
021114  104453                                                  TRAP    C$DU
```

```
5463
5464
5465                                    .SBTTL  ADD UNIT SECTION
5466
5467                                    ;++
5468                                    ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
5469                                    ;  TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
5470                                    ;  TO THE TEST CYCLE.
5471                                    ;--
5472
5473  021116                                    BGNAU
5474  021116                                                                  L$AU::
5475  021116                                    EXIT    AU
5476  021116  000167                                                                .WORD   J$JMP
5477  021120  000000                                                                .WORD   L10022-2-.
5478
5479                                    .EVEN
5480
5481
5482  021122                                    ENDAU
5483  021122                                                                  L10022:
5484  021122  104452                                                                TRAP    C$AU
5485
```

```
                                .SBTTL  HARDWARE TEST           - ADRA -
                                ;++
                                ;*****************************************************************************
                                ;*                          - REGISTER ADDRESS TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE Q-BUS CAN READ AND WRITE TO THE DHV11
                                ;*      DEVICE REGISTERS.  IF THE DHV11 DOES NOT RESPOND TO THE ACCESS
                                ;*      ATTEMPTS (IF THE DHV11 IS AT THE WRONG ADDRESS, FOR EXAMPLE) THE
                                ;*      004 BUS TIME-OUT TRAP IS DETECTED BY THIS ROUTINE AND AN ERROR
                                ;*      IS REPORTED.
                                ;*
                                ;*****************************************************************************
                                ;--

                                        BGNTST
                                                                                T1::
                                        TNUM == 1               ;THIS TEST MUST ALWAYS BE INCLUDED AS TEST 1.
021124  012737  000001  002272          MOV     #TNUM,TSTNUM    ;SET THE TEST NUMBER TO 1.
021132  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
                                ;+
                                ; SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:
                                ;-
021140  013737  000004  002314          MOV     4,TP4VEC        ;SAVE THE EXISTING 004 TRAP VECTOR.
021146  012737  020016  000004          MOV     #TP4RTN,4       ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
021154  005005                          CLR     R5              ;CLEAR THE ERROR FLAGS.

                                ;+
                                ; SET UP FOR THE INITIAL ITERATION OF THE TEST LOOP:
                                ;-
021156  005004                          CLR     R4              ;CLEAR THE LINE COUNTER.

                                ;+
                                ; HERE BEGINS THE LOOP TO TEST THE REGISTERS FOR A LINE.
                                ;    FIRST TEST THE CSR AND SET THE IND.ADR.REG (I.A.R) FIELD.
                                ;-
021160  005037  002316          2$:     CLR     TP4FLG          ;CLEAR THE 004 TRAP FLAG.
021164  013700  002246                  MOV     CSRA,R0         ;SET UP CSR AS THE CKTRAP MOVE SOURCE.
021170  012701  021404                  MOV     #52$,R1         ;SET UP DESTINATION LOCATION FOR CKTRAP MOVE.
021174  004737  014430                  JSR     PC,CKTRAP       ;MOVE AND CHECK FOR TRAP.
021200  103402                          BCS     4$              ;IF NO TRAP, BYPASS ERROR.
021202  052705  100001                  BIS     #100001,R5      ;SET FATAL READ ERROR FLAGS.
021206  042737  000017  021404  4$:     BIC     #17,52$         ;CLEAR THE I.A.R FIELD OF THE CSR DATA.
021214  050437  021404                  BIS     R4,52$          ;OR IN THE LINE COUNTER TO THE I.A.R FIELD.
021220  010100                          MOV     R1,R0           ;USE OLD DESTINATION FOR SOURCE OF CKTRAP MOVE.
021222  013701  002246                  MOV     CSRA,R1         ;SET UP CSR AS THE CKTRAP MOVE DESTINATION.
021226  004737  014430                  JSR     PC,CKTRAP       ;MOVE AND CHECK FOR TRAP.
021232  103403                          BCS     6$              ;IF NO TRAP, BYPASS ERROR.
021234  052705  100002                  BIS     #100002,R5      ;SET FATAL WRITE ERROR FLAGS.
021240  000440                          BR      40$             ;EXIT AND REPORT FATAL ERROR.
                                ;+
                                ;    NOW, WE TEST EACH REGISTER FOR THIS LINE.
                                ;-
021242  012702  000010          6$:     MOV     #10,R2          ;INIT REGISTER COUNTER TO 8.
021246  013737  002246  021402          MOV     CSRA,50$                ;INITIALIZE THE REGISTER POINTER.
021254  012700  021402          8$:     MOV     #50$,R0         ;SET UP REGISTER AS THE SOURCE FOR CKTRAP MOVE.
021260  012701  021404                  MOV     #52$,R1         ;SET UP LOCAL STORAGE AS THE DES FOR CKTRAP.
```

```
5542  021264  004737  014430                  JSR     PC,CKTRAP       ;PERFORM THE MOVE, CHECK FOR TRAP.
5543  021270  103402                          BCS     10$             ;IF NO TRAP, BYPASS THE SETTING OF ERROR FLAGS.
5544  021272  052705  100001                    BIS     #100001,R5      ;SET FATAL READ ERROR FLAGS.
5545  021276  010100                  10$:      MOV     R1,R0           ;USE OLD DEST AS SRC FOR CKTRAP MOVE.
5546  021300  012701  021402                    MOV     #50$,R1         ;SET UP REGISTER AS THE DEST FOR CKTRAP MOVE.
5547  021304  004737  014430                  JSR     PC,CKTRAP       ;PERFORM THE MOVE, CHECK FOR TRAP.
5548  021310  103402                          BCS     12$             ;IF NO TRAP, BYPASS THE SETTING OF ERROR FLAGS.
5549  021312  052705  100002                    BIS     #100002,R5      ;SET FATAL WRITE ERROR FLAGS.
5550  021316  005237  021402          12$:      INC     50$             ;INCREMENT THE REGISTER
5551  021322  005237  021402                    INC     50$             ; POINTER BY 2.
5552  021326  005302                            DEC     R2              ;COUNT THE REGISTER.
5553  021330  001351                          BNE     8$              ;LOOP TO TEST THE NEXT REGISTER ADDRESS.
5554
5555                                  ;+
5556                                  ; NOW WE SET UP TO TEST THE NEXT LINE, OR TO EXIT IF WE ARE DONE.
5557                                  ;-
5558  021332  005204                            INC     R4              ;INCREMENT THE LINE COUNTER.
5559  021334  020427  000010                    CMP     R4,#NUMLNS      ;COMPARE LINE COUNTER AGAINST NUMBER OF LINES.
5560  021340  002707                          BLT     2$              ;LOOP TO TEST THE NEXT LINE IF WE'RE NOT DONE.
5561
5562                                  ;+
5563                                  ; DONE CHECKING DEVICE REGISTER ADDRESSES.
5564                                  ;    REPORT ANY ERRORS AND EXIT.
5565                                  ;-
5566  021342  013737  002314  000004  40$:      MOV     TP4VEC,4        ;RESTORE THE NORMAL 004 TRAP VECTOR.
5567  021350  005705                            TST     R5              ;CHECK THE ERROR FLAGS.
5568  021352  100015                          BPL     60$             ;EXIT ROUTINE IF NO ERRORS.
5569                                      ; REPORT "DEVICE REGISTER ACCESS ERRORS"
5570  021354                                    ERRDF   101,EM0103,ER0101;              >>>>> ERROR #101 <<<<<.
5571  021354  104455                                                                    TRAP    C$ERDF
5572  021356  000145                                                                    .WORD   101
5573  021360  005177                                                                    .WORD   EM0103
5574  021362  012322                                                                    .WORD   ER0101
5575
5576  021364                                  DODU    UNITN           ;DROP THIS UNIT FROM FUTHER TESTING.
5577  021364  013700  002244                                                           MOV     UNITN,R0
5578  021370  104451                                                                   TRAP    C$DODU
5579  021372  005037  002270                    CLR     CTRLCF          ;INDICATE NO CTRL-C ABORT FROM TEST.
5580  021376                                  DOCLN                   ;ABORT THIS SUB PASS.
5581  021376  104444                                                                   TRAP    C$DCLN
5582  021400  000402                          BR      60$             ;
5583                                  ;+
5584                                  ; LOCAL STORAGE.
5585                                  ;-
5586  021402  000000                  50$:    .WORD   0       ;STORAGE FOR THE SOURCE OR DEST OF THE CKTRAP MOVE.
5587  021404  000000                  52$:    .WORD   0       ;STORAGE FOR THE SOURCE OR DEST OF THE CKTRAP MOVE.
5588  021406  005037  002270          60$:     CLR    CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
5589  021412                                  ENDTST
5590  021412                                                                  L10023:
5591  021412  104401                                                                   TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - NOTXDV -
                                ;++ ************************************************************************
                                ;*                          - NO TX_DATA_VALID/NO TX_ACTION TEST -
                                ;*      THIS TEST VERIFIES THAT IF A DATA WORD IS WRITTEN WITHOUT THE
                                ;*      TX_DATA_VALID BIT SET, NO TX_ACTION WILL BE GENERATED.
                                ;*      TO ENSURE DATA IS NOT ACCIDENTALLY TRANSMITTED, THE TEST IS PERFORMED
                                ;*      IN INTERNAL LOOPBACK, AND ON ALL ACTIVE LINES.
                                ;*
                                ;-- ************************************************************************
021414                                  BGNTST
021414                                                                  T2::
021414                                  SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
021414  012700  000240                                                  MOV     #PRI05,R0
021420  104441                                                          TRAP    C$SPRI
        000002                          TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
021422  012737  000002  002272           MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (21)
021430  012737  177777  002270           MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
021436  012737  000001  004052           MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
021444  012737  004065  004054           MOV     #2101.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
021452  012737  005500  004056           MOV     #EM2101,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
021460  012737  013544  004060           MOV     #ER9101,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 2101 <<<<<.
                                ;-
021466  004737  014460                  JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
021472  103054                          BCC     60$             ;RESET FAILURE?, ABORT THIS TEST.
021474  005237  004054                    INC     ERRNBR          ;SET THE ERROR NUMBER TO 2102.
                                ;+
                                ; SET INTERNAL LOOPBACK ON ALL ACTIVE LINES.
                                ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
                                ; 2 STOP BITS.
                                ; DISABLE TRANSMITTERS ON ALL ACTIVE LINES.
                                ;-
021500  013705  002240                    MOV     ACTLNS,R5       ;PASS THE ACTIVE LINE BIT MAP.
021504  012700  000200                    MOV     #200,R0         ;PASS THE LNCTRL CONTENTS.
021510  004737  017412                  JSR     PC,WTWLNC       ;INITIALISE THE LNCTRL REGISTERS.
021514  012700  177670                    MOV     #177670,R0      ;PASS THE LPR CONTENTS.
021520  004737  017466                  JSR     PC,WTWLPR       ;INITIALSE THE LPR REGISTERS ON ALL LINES.
021524  012704  000012                    MOV     #10.,R4         ;PASS DELAY TIME OF 10 MILLI SECS.
021530  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
021534  004737  016612                  JSR     PC,TXDSBL       ;DISABLE TRANSMITTERS ON ALL ACTIVE LINES.
                                ;+
                                ; TEST ALL ACTIVE LINES INDIVIDUALLY.
                                ; WRITE A DATA WORD TO THE TXCHAR REGISTER WITH TX_DATA_VALID CLEAR.
                                ; VERIFY NO TX_ACTION IS GENERATED.
                                ;-
021540  013705  002240                    MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
021544  005004                            CLR     R4              ;CLEAR THE LINE NUMBER COUNTER.
021546  000241                  2$:       CLC                     ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
021550  006005                            ROR     R5              ;SHIFT THE BIT MAP INTO THE CARRY BIT.
021552  103020                          BCC     4$              ;DO NOT TEST THE LINE IF IT IS INACTIVE.
                                    ;+
                                    ; SELECT THE LINE UNDER TEST.
                                    ;  WRITE DATA WORD (ASCII <LF>) TO TXCHR REGISTER WITH THE MOST SIGNIFICANT
```

```
                        ;  BIT (TX_DATA_VALID) CLEAR.
                        ;-
021554  010177  160466          MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
021560  012777  000012  160462  MOV     #12,@TXCHA      ;WRITE THE DATA WORD TO THE DUT'S TXCHAR REG.
                        ;+
                        ; WAIT FOR A TX_ACTION TO BE RETURNED, REPORT ERROR IF TX_ACTION FOUND
                        ; BEFORE TIME-OUT OCCURS.
                        ;-
021566  012701  170002          MOV     #170002,R1      ;TEST BIT 15, TIMEOUT OF 2 MILLI SECS.
021572  013702  002246          MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
021576  004737  017276          JSR     PC,WAIBIS       ;WAIT FOR TX_ACTION TO COME BACK.
021602  103004                  BCC     4$              ;SKIP ERROR REPORT IF TX-ACTION NOT FOUND.

021604  010401                  MOV     R4,R1           ;PASS THE NUMBER OF CURRENT LINE UNDER TEST.
021606  012702  005543          MOV     #EM2102,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
                                                        ;''TX_ACT FOUND AFTER INVALID DATA WORD WRITTEN''
021612                          ERROR                   ;                >>>>> ERROR #4102 <<<<<.
021612  104460                                                                  TRAP    C$ERROR

                        ;+
                        ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
                        ;-
021614  005204          4$:     INC     R4              ;INCREMENT THE LINE NUMBER COUNTER.
021616  005705                  TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
021620  001352                  BNE     2$              ;YES; BRANCH TO TEST THE NEXT LINE.
021622  000400                  BR      60$             ;NO; EXIT THIS TEST.

021624  005037  002270  60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
021630                          ENDTST
021630                                                                  L10024:
021630  104401                                                                  TRAP    C$ETST
```

```
5678                                    .SBTTL  HARDWARE TEST           - TXDVAL-
5679                                    ;++ ************************************************************************
5680                                    ;*                              - TX DATA VALID/TX ACTION TEST -
5681                                    ;*      THIS TEST VERIFIES THAT IF A DATA WORD IS WRITTEN TO THE TXCHAR REGISTER
5682                                    ;*      WITH THE TX_DATA_VALID BIT SET, A CORRESPONDING TX_ACTION WILL BE
5683                                    ;*      GENERATED.
5684                                    ;*      TO ENSURE DATA IS NOT ACCIDENTALLY TRANSMITTED, THE TEST IS PERFORMED
5685                                    ;*      IN INTERNAL LOOPBACK, AND ON ALL ACTIVE LINES.
5686                                    ;*
5687                                    ;-- ************************************************************************
5688
5689   021632                                   BGNTST
5690   021632                                                                                   T3::
5691   021632                                   SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
5692   021632  012700  000240                                                                   MOV     #PRI05,R0
5693   021636  104441                                                                           TRAP    C$SPRI
5694           000003                           TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
5695   021640  012737  000003  002272          MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.          (22)
5696   021646  012737  177777  002270          MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
5697   021654  012737  000001  004052          MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
5698   021662  012737  004231  004054          MOV     #2201.,ERRNBR  ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
5699   021670  012737  005637  004056          MOV     #EM2201,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
5700   021676  012737  013544  004060          MOV     #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
5701                                   ;+
5702                                   ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
5703                                   ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
5704                                   ; THIS SUBROUTINE REPORTS ERROR >>>>> 2201 <<<<<.
5705                                   ;-
5706   021704  004737  014460                  JSR     PC,CLNRST      ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
5707   021710  103066                          BCC     60$            ;RESET FAILURE?, ABORT THIS TEST.
5708                                   ;+
5709                                   ; SET INTERNAL LOOPBACK ON ALL ACTIVE LINES.
5710                                   ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
5711                                   ; 2 STOP BITS.
5712                                   ; DISABLE TRANSMITTERS ON ALL ACTIVE LINES.
5713                                   ;-
5714   021712  013705  002240                  MOV     ACTLNS,R5      ;PASS THE ACTIVE LINE BIT MAP.
5715   021716  012700  000200                  MOV     #200,R0        ;PASS THE LNCTRL CONTENTS.
5716   021722  004737  017412                  JSR     PC,WTWLNC      ;INITIALISE THE LNCTRL REGISTERS.
5717   021726  012700  177670                  MOV     #177670,R0     ;PASS THE LPR CONTENTS.
5718   021732  004737  017466                  JSR     PC,WTWLPR      ;INITIALSE THE LPR REGISTERS ON ALL LINES.
5719   021736  012704  000012                  MOV     #10.,R4        ;PASS DELAY TIME OF 10 MILLI-SECONDS.
5720   021742  004737  014574                  JSR     PC,DELAY       ;WAIT FOR LNCTR AND LPR REGS TO BE UPDATED.
5721   021746  004737  016612                  JSR     PC,TXDSBL      ;DISABLE TRANSMITTERS ON ALL ACTIVE LINES.
5722                                   ;+
5723                                   ; TEST ALL ACTIVE LINES INDIVIDUALLY.
5724                                   ; WRITE A DATA WORD TO THE TXCHAR REGISTER WITH TX_DATA_VALID SET.
5725                                   ; VERIFY THAT A CORRESPONDING TX_ACTION IS GENERATED.
5726                                   ;-
5727   021752  013705  002240                  MOV     ACTLNS,R5      ;GET THE ACTIVE LINE BIT MAP.
5728   021756  005004                          CLR     R4             ;CLEAR THE LINE NUMBER COUNTER.
5729   021760  012737  004232  004054  2$:     MOV     #2202.,ERRNBR  ;SET THE ERROR NUMBER TO 2202.
5730   021766  000241                          CLC                    ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
5731   021770  006005                          ROR     R5             ;SHIFT THE BIT MAP INTO THE CARRY BIT.
5732   021772  103032                          BCC     8$             ;DO NOT TEST THE LINE IF IT IS INACTIVE.
5733                                      ;+
```

```
                                        ; SELECT THE LINE UNDER TEST.
                                        ;  WRITE DATA WORD (ASCII <LF>) TO TXCHR REGISTER WITH THE MOST SIGNIFICANT
                                        ;  BIT (TX_DATA_VALID) SET.
                                        ;-
021774  010477  160246                      MOV     R4,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
022000  012777  100012  160242              MOV     #100012,@TXCHA  ;WRITE THE DATA WORD TO THE DUT'S TXCHAR REG.
                                        ;+
                                        ; WAIT FOR A TX_ACTION TO BE RETURNED, REPORT ERROR IF NO TX_ACTION
                                        ; FOUND BEFORE TIME-OUT OCCURS.
                                        ;-
022006  012701  170002                      MOV     #170002,R1      ;TEST BIT 15, TIMEOUT OF 2 MILLI SECS.
022012  013702  002246                      MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
022016  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR TX_ACTION TO COME BACK.
022022  103403                            BCS     4$              ;SKIP ERROR REPORT IF TX-ACTION FOUND.
022024  012702  005674                      MOV     #EM2202,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
                                                                    ;"NO TX_ACT FOUND AFTER VALID DATA WORD TX'D".
022030  000411                            BR      6$              ;GO REPORT THE ERROR.
                                        ;+
                                        ; VERIFY TX_ACTION RETURNED FROM CORRECT LINE.
                                        ;-
022032  005237  004054          4$:       INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 2103.
022036  000302                            SWAB    R2              ;GET THE LINE NUMBER IN THE LOW BYTE.
022040  042702  177760                    BIC     #177760,R2      ;CLEAR THE UNWANTED BITS.
022044  020204                            CMP     R2,R4           ;IS IT THE CORRECT LINE NUMBER?.
022046  001404                            BEQ     8$              ;YES; SKIP THE ERROR REPORT.
022050  012702  005766                      MOV     #EM2203,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
                                                                    ; "INCORRECT LINE # RETURNED WITH TX_ACT"
022054  010401                  6$:       MOV     R4,R1           ;PASS THE NUMBER OF CURRENT LINE UNDER TEST.
022056                                    ERROR                   ;                   >>>>> ERROR <<<<<.
022056  104460                                                                              TRAP    C$ERROR

                                        ;+
                                        ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
                                        ;-
022060  005204                  8$:       INC     R4              ;INCREMENT THE LINE NUMBER COUNTER.
022062  005705                            TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
022064  001335                          BNE     2$              ;YES; BRANCH TO TEST THE NEXT LINE.

022066  005037  002270          60$:      CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
022072                                    ENDTST
022072                                                                          L10025:
022072  104401                                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - TXENBI-
                                ;++ ************************************************************************
                                ;*                      - TX ENABLE (INACTIVE) TEST -
                                ;*      THIS TEST VERIFIES THAT WHEN THE LINE UNDER TEST'S TX_ENABLE BIT IS
                                ;*      CLEAR, TRANSMISSION WILL NOT TAKE PLACE ON THAT LINE.
                                ;*      THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, AND ON ALL ACTIVE LINES.
                                ;*
                                ;-- ************************************************************************

                                        BGNTST
                                                                                T4::
                                        SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
                                                                                MOV     #PRI05,R0
                                                                                TRAP    C$SPRI
                                         TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
022102  012737  000004  002272          MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (23)
022110  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
022116  012737  000001  004052          MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
022124  012737  004375  004054          MOV     #2301.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
022132  012737  006073  004056          MOV     #EM2301,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
022140  012737  013544  004060          MOV     #ER9101,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 2301 <<<<<.
                                ;-
022146  004737  014460                  JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
022152  103110                        BCC     60$             ;RESET FAILURE?, ABORT THIS TEST.
                                ;+
                                ; SET INTERNAL LOOPBACK ON ALL ACTIVE LINES.
                                ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
                                ; 2 STOP BITS.
                                ; ENABLE TRANSMITTERS ON ALL LINES.
                                ;-
022154  013705  002240                   MOV     ACTLNS,R5       ;PASS THE ACTIVE LINE BIT MAP.
022160  012700  000200                   MOV     #200,R0         ;PASS THE LNCTRL CONTENTS.
022164  004737  017412                  JSR     PC,WTWLNC       ;INITIALISE THE LNCTRL REGISTERS.
022170  012700  177670                   MOV     #177670,R0      ;PASS THE LPR CONTENTS.
022174  004737  017466                  JSR     PC,WTWLPR       ;INITIALSE THE LPR REGISTERS ON ALL LINES.
022200  012704  000012                   MOV     #10.,R4         ;PASS DELAY TIME OF 10 MILLI-SECONDS.
022204  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LNCTR AND LPR REGS TO BE UPDATED.
022210  012705  000377                   MOV     #MAPLNS,R5      ;PASS THE BIT MAP CORRESPONDING TO ALL LINES.
022214  004737  016706                  JSR     PC,TXENBL       ;ENABLE TRANSMITTERS ON ALL LINES.
                                ;+
                                ; TEST ALL ACTIVE LINES INDIVIDUALLY.
                                ; DISABLE TRANSMISSION ON EACH ACTIVE LINE.
                                ;-
022220  012703  000001                   MOV     #1,R3           ;SET UP THE LINE BIT MAP FOR CHANNEL 0.
022224  005004                           CLR     R4              ;CLEAR THE LINE NUMBER COUNTER.
022226  012737  004376  004054  2$:      MOV     #2302.,ERRNBR   ;SET THE ERROR NUMBER TO 2302.
022234  030337  002240                   BIT     R3,ACTLNS       ;CHECK IF THE LINE IS ACTIVE.
022240  001447                          BEQ     6$              ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
                                    ;+
                                    ; CLEAR THE TX_ENABLE BIT IN TBUFFAD2 REGISTER.
                                    ; SELECT THE LINE UNDER TEST.
                                    ;   VERIFY IT IS CLEAR, REPORT ERROR IF SET.
```

```
5832                                                    ;-
5833  022242  010305                                      MOV     R3,R5           ;PASS THE BIT MAP OF THE LINE UNDER TEST.
5834  022244  004737  016612                            JSR     PC,TXDSBL       ;DISABLE TRANSMISSION ON THE LINE UNDER TEST.
5835  022250  010477  157772                             MOV     R4,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
5836  022254  005777  160002                             TST     @TXAD2A         ;VERIFY THE TX_ENABLE BIT IS SET.
5837  022260  100433                                     BMI     4$              ;GO REPORT ERROR IF TX_ENABLE BIT SET.
5838                                                    ;+
5839                                                    ; WRITE DATA WORD (ASCII <LF>) TO TXCHR REGISTER.
5840                                                    ; WAIT FOR A TX_ACTION TO BE RETURNED, REPORT ERROR IF A TX_ACTION
5841                                                    ; IS FOUND BEFORE TIME-OUT OCCURS.
5842                                                    ;-
5843  022262  012737  004377  004054                     MOV     #2303.,ERRNBR   ;SET ERROR NUMBER TO 2303.
5844  022270  012777  100012  157752                     MOV     #100012,@TXCHA  ;WRITE THE DATA WORD TO THE DUT'S TXCHAR REG.
5845  022276  012701  170002                             MOV     #170002,R1      ;TEST BIT 15, TIMEOUT OF 2 MILLI SECS.
5846  022302  013702  002246                             MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5847  022306  004737  017276                            JSR     PC,WAIBIS       ;WAIT FOR TX_ACTION TO COME BACK.
5848  022312  103016                                     BCC     4$              ;GO REPORT ERROR IF NO TX-ACTION FOUND.
5849                                                    ;+
5850                                                    ; WAIT FOR THE DATA TO APPEAR IN THE FIFO, REPORT ERROR IF DATA FOUND.
5851                                                    ;-
5852  022314  005237  004054                             INC     ERRNBR          ;SET ERROR NUMBER TO 2304.
5853  022320  012701  070012                             MOV     #70012,R1       ;TEST BIT 7, TIMEOUT OF 10 MILLI SECS.
5854  022324  013702  002246                             MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5855  022330  004737  017276                            JSR     PC,WAIBIS       ;WAIT FOR RX_DATA_AVAILABLE TO SET.
5856  022334  103405                                     BCS     4$              ;REPORT ERROR IF DATA RECEIVED IN THE FIFO.
5857  022336  005237  004054                             INC     ERRNBR          ;SET ERROR NUMBER TO 2305.
5858  022342  017702  157702                             MOV     @RBUFA,R2       ;READ THE DATA FROM THE FIFO.
5859  022346  100004                                     BPL     6$              ;SKIP ERROR REPORT IF DATA IS THERE.
5860
5861  022350  010401                                4$:     MOV     R4,R1           ;PASS THE NUMBER OF CURRENT LINE UNDER TEST.
5862  022352  012702  006131                              MOV     #EM2302,R2      ;PASS THE MESSAGE TO BE REPORTED.
5863                                                                                ;'TX_ENABLE BIT BAD ON LINE: NN'.
5864  022356                                            ERROR                     ;                >>>>> ERROR <<<<<.
5865  022356  104460                                                                                        TRAP    C$ERROR
5866                                                    ;+
5867                                                    ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
5868                                                    ;-
5869  022360  000241                                6$:     CLC                     ;CLEAR THE CARRY BIT PRIOR TO ROTATION.
5870  022362  006103                                      ROL     R3              ;SHIFT THE BIT MAP FOR THE NEXT LINE.
5871  022364  005204                                      INC     R4              ;INCREMENT THE LINE NUMBER COUNTER.
5872  022366  020427  000010                              CMP     R4,#NUMLNS      ;HAVE ALL THE LINES BEEN TESTED?.
5873  022372  002715                                     BLT     2$              ;NO; BRANCH TO TEST THE NEXT LINE.
5874
5875  022374  005037  002270                        60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
5876  022400                                            ENDTST
5877  022400                                                                                      L10026:
5878  022400  104401                                                                                        TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - TXENBA-
                                ;++ ************************************************************************
                                ;*                      - TX ENABLE (ACTIVE) TEST -
                                ;*      THIS TEST VERIFIES TRAT WHEN THE TX ENABLE BIT IS SET IN THE APPRORIATE
                                ;*      LINE REGISTER, TRANSMISSION WILL TAKE PLACE ON THAT LINE.
                                ;*      THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, AND ON ALL ACTIVE LINES.
                                ;*
                                ;-- ************************************************************************

                                        BGNTST
                                                                                T5::
                                        SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
022402  012700  000240                                                   MOV     #PRI05,R0
022406  104441                                                          TRAP    C$SPRI
        000005
                                         TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
022410  012737  000005  002272           MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.        (24)
022416  012737  177777  002270           MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
022424  012737  000001  004052           MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
022432  012737  004541  004054           MOV     #2401.,ERRNBR  ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
022440  012737  006167  004056           MOV     #EM2401,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
022446  012737  013544  004060           MOV     #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 2401 <<<<<.
                                ;-
022454  004737  014460                   JSR     PC,CLNRST      ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
022460  103127                          BCC     60$            ;RESET FAILURE?, ABORT THIS TEST.
                                ;+
                                ; SET INTERNAL LOOPBACK ON ALL ACTIVE LINES.
                                ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
                                ; 2 STOP BITS.
                                ; DISABLE TRANSMITTERS ON ALL LINES.
                                ;-
022462  013705  002240                    MOV     ACTLNS,R5      ;PASS THE ACTIVE LINE BIT MAP.
022466  012700  000200                    MOV     #200,R0        ;PASS THE LNCTRL CONTENTS.
022472  004737  017412                   JSR     PC,WTWLNC      ;INITIALISE THE LNCTRL REGISTERS.
022476  012700  177670                    MOV     #177670,R0     ;PASS THE LPR CONTENTS.
022502  004737  017466                   JSR     PC,WTWLPR      ;INITIALSE THE LPR REGISTERS ON ALL LINES.
022506  012704  000012                    MOV     #10.,R4        ;PASS DELAY TIME OF 10 MILLI-SECONDS.
022512  004737  014574                   JSR     PC,DELAY       ;WAIT FOR LNCTR AND LPR REGS TO BE UPDATED.
022516  012705  000377                    MOV     #MAPLNS,R5     ;PASS THE BIT MAP CORRESPONDING TO ALL LINES.
022522  004737  016612                   JSR     PC,TXDSBL      ;DISABLE TRANSMITTERS ON ALL LINES.
                                ;+
                                ; TEST ALL ACTIVE LINES INDIVIDUALLY.
                                ; ENABLE TRANSMISSION ON EACH ACTIVE LINE.
                                ;-
022526  012703  000001                    MOV     #1,R3          ;SET UP THE LINE BIT MAP FOR CHANNEL 0.
022532  005004                          CLR     R4             ;CLEAR THE LINE NUMBER COUNTER.
022534  012737  004542  004054  2$:      MOV     #2402.,ERRNBR  ;SET THE ERROR NUMBER TO 2402.
022542  030337  002240                   BIT     R3,ACTLNS      ;CHECK IF THE LINE IS ACTIVE.
022546  001463                          BEQ     8$             ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
                                ;+
                                ; SELECT THE LINE UNDER TEST.
                                ;  SET THE TX_ENABLE BIT IN TBUFFAD2 REGISTER.
                                ;   VERIFY IT IS SET, REPORT ERROR IF CLEAR.
```

```
5935                                            ;-
5936  022550  010305                              MOV     R3,R5           ;PASS THE BIT MAP OF THE LINE UNDER TEST.
5937  022552  004737  016706                      JSR     PC,TXENBL       ;ENABLE TRANSMISSION ON THE LINE UNDER TEST.
5938  022556  012705  000012                      MOV     #10.,R5         ;SET TXCHAR/LOOP COUNT TO 10.
5939  022562  010477  157460                      MOV     R4,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
5940  022566  005777  157470                      TST     @TXAD2A         ;VERIFY THE TX_ENABLE BIT IS SET.
5941  022572  100045                              BPL     6$              ;GO REPORT ERROR IF TX_ENABLE BIT CLEAR.
5942                                            ;+
5943                                            ; WRITE DATA WORD (ASCII <LF>) TO TXCHR REGISTER.
5944                                            ; WAIT FOR A TX_ACTION TO BE RETURNED, REPORT ERROR IF NO TX_ACTION
5945                                            ; FOUND BEFORE TIME-OUT OCCURS.
5946                                            ;-
5947  022574  012737  004543  004054  4$:        MOV     #2403.,ERRNBR   ;SET ERROR NUMBER TO 2403.
5948  022602  012777  100012  157440             MOV     #100012,@TXCHA  ;WRITE THE DATA WORD TO THE DUT'S TXCHAR REG.
5949  022610  012701  170002                     MOV     #170002,R1      ;TEST BIT 15, TIMEOUT OF 2 MILLI SECS.
5950  022614  013702  002246                     MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5951  022620  004737  017276                     JSR     PC,WAIBIS       ;WAIT FOR TX_ACTION TO COME BACK.
5952  022624  103030                             BCC     6$              ;GO REPORT ERROR IF NO TX-ACTION FOUND.
5953                                            ;+
5954                                            ; WAIT FOR THE DATA TO APPEAR IN THE FIFO, REPORT ERROR IF TIME-OUT.
5955                                            ;-
5956  022626  005237  004054                     INC     ERRNBR          ;SET ERROR NUMBER TO 2404.
5957  022632  012701  070012                     MOV     #70012,R1       ;TEST BIT 7, TIMEOUT OF 10 MILLI SECS.
5958  022636  013702  002246                     MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5959  022642  004737  017276                     JSR     PC,WAIBIS       ;WAIT FOR RX DATA AVAILABLE TO SET.
5960  022646  103017                             BCC     6$              ;REPORT ERROR IF NO DATA RECEIVED IN THE FIFO.
5961  022650  005237  004054                     INC     ERRNBR          ;SET ERROR NUMBER TO 2405.
5962  022654  017702  157370                     MOV     @RBUFA,R2       ;READ THE DATA FROM THE FIFO.
5963  022660  100012                             BPL     6$              ;GO REPORT ERROR IF THER IS'NT ANY DATA THERE.
5964  022662  005237  004054                     INC     ERRNBR          ;SET ERROR NUMBER TO 2406.
5965  022666  000302                             SWAB    R2              ;PUT THE LINE NUMBER IN THE LOW BYTE.
5966  022670  042702  177760                     BIC     #177760,R2      ;CLEAR THE UNWANTED BITS.
5967  022674  020204                             CMP     R2,R4           ;DID THE DATA COME FROM THE CORRECT LINE?.
5968  022676  001003                             BNE     6$              ;NO; GO REPORT THE ERROR.
5969  022700  005305                             DEC     R5              ;DECREMENT THE TXCHAR/LOOP COUNTER.
5970  022702  001334                             BNE     4$              ;LOOP TO TX THE NEXT CHAR.
5971  022704  000404                             BR      8$              ;GO TEST THE NEXT LINE.
5972
5973  022706  010401                     6$:     MOV     R4,R1           ;PASS THE NUMBER OF CURRENT LINE UNDER TEST.
5974  022710  012702  006131                     MOV     #EM2302,R2      ;PASS THE MESSAGE TO BE REPORTED.
5975                                             ;'TX_ENABLE BIT BAD ON LINE: NN'.
5976  022714                                     ERROR                   ;              >>>>> ERROR <<<<<.
5977  022714  104460                                                                      TRAP    C$ERROR
5978                                          ;+
5979                                          ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
5980                                          ;-
5981  022716  010305                     8$:     MOV     R3,R5           ;PASS THE BIT MAP OF THE LINE UNDER TEST.
5982  022720  004737  016612                     JSR     PC,TXDSBL       ;CLEAR THE TX ENABLE BIT ON THIS LINE.
5983  022724  000241                             CLC                     ;CLEAR THE CARRY BIT PRIOR TO ROTATION.
5984  022726  006103                             ROL     R3              ;SHIFT THE BIT MAP FOR THE NEXT LINE.
5985  022730  005204                             INC     R4              ;INCREMENT THE LINE NUMBER COUNTER.
5986  022732  020427  000010                     CMP     R4,#NUMLNS      ;HAVE ALL THE LINES BEEN TESTED?.
5987  022736  002676                             BLT     2$              ;NO; BRANCH TO TEST THE NEXT LINE.
5988
5989  022740  005037  002270             60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NO1 WITHIN A TEST.
5990  022744                                     ENDTST
```

```
 5991  022744
 5992  022744  104401                    L10027:
                                                TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - INTA -
                                ;++ ******************************************************************************
                                ;*                          - INTERRUPT TEST -
                                ;*      THIS TEST VERIFIES THAT THE DEVICE UNDER TEST (DUT) WILL GENERATE
                                ;*      RECEPTION AND TRANSMISSION INTERRUPTS CORRECTLY.  THIS TEST DOES
                                ;*      NOT DEPEND ON THE USE OF THE SERIAL LINE TRANSMISSION OR RECEPTION
                                ;*      CAPABILITIES OF THE DUT.  THE LINES ARE PUT IN INTERNAL LOOPBACK
                                ;*      TO MINIMIZE ANY EXTERNAL EFFECTS THAT COULD BE CAUSED ON DEVICES
                                ;*      ATTACHED TO THE SERIAL LINES.
                                ;*
                                ;-- ******************************************************************************

                                        BGNTST
                                                                                T6::
                                        SETPRI  #PRI05          ;ALLOW THE LTC TO INTERRUPT.
022746  012700  000240                                                    MOV     #PRI05,R0
022752  104441                                                            TRAP    C$SPRI
        000006                   TNUM == TNUM + 1               ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
022754  012737  000006  002272    MOV     #TNUM,TSTNUM           ;SET UP THE TEST NUMBER.        (26)
022762  012737  177777  002270    MOV     #-1,CTRLCF             ;INDICATE THAT WE ARE IN A TEST.
022770  012737  000001  004052    MOV     #1,ERRTYP              ;SET ERROR FATAL ERROR TYPE IN ERROR TABLE.
022776  012737  003101  004054    MOV     #1601.,ERRNBR          ;SET FIRST ERROR REPORT NUMBER IN ERROR TABLE.
023004  012737  006223  004056    MOV     #EM2601.,ERRMSG        ;SET TEST ERROR MESSAGE IN ERROR TABLE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, DO NOT REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERRORS FROM >>>>> 2601 THRU 2602 <<<<<.
                                ;-
023012  004737  016022                JSR     PC,RESETT              ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
023016  103402                    BCS     2$                     ;SKIP AROUND ABORTING TEST IF NO ERROR FOUND.
023020  000137  023764            JMP     60$                    ;ABORT TEST IF FATAL ERROR FOUND DURING RESET.
023024  012737  005053  004054  2$:       MOV     #2603.,ERRNBR          ;SET THE ERROR REPORT NUMBER TO 2603.
                                ;+
                                ; ENABLE TRANSMITTERS ON ALL LINES.
                                ;-
023032  012705  000377          4$:       MOV     #MAPLNS,R5             ;PASS ACTIVE LINE BIT MAP.
023036  004737  016706                    JSR     PC,TXENBL              ;ENABLE TRANSMISSION ON ALL LINES.
                                ;+
                                ; TEST RECEPTION INTERRUPTS.
                                ;    SET UP FOR RX AND TX INTERRUPTS:
                                ;    RX INTERRUPT SERVICE ROUTINE INPUTS A CHAR AND COUNTS THE INTERRUPT.
                                ;    TX INTERRUPT SERVICE ROUTINE COUNTS TX INTERRUPTS.
                                ;-
023042  005037  002304                CLR     RXINTC                 ;CLEAR THE RX INTERRUPT COUNTER.
023046  005037  002306                CLR     RXINTF                 ;CLEAR THE RX INTERRUPT FLAGS.
023052  005037  002310                CLR     TXINTC                 ;CLEAR THE TX INTERRUPT COUNTER.
023056  012737  002712  002266        MOV     #BUFBAS,BUFPTR         ;LOAD THE BUFFER PTR WITH THE BUFFER BASE ADR.
023064                              SETVEC  RXVECA,#RXINPT,#PRI05 ;SET UP INTERRUPT VECTOR TO CATCH RX INT.
023064  012746  000240                                                    MOV     #PRI05,-(SP)
023070  012746  017732                                                    MOV     #RXINPT,-(SP)
023074  013746  002234                                                    MOV     RXVECA,-(SP)
023100  012746  000003                                                    MOV     #3,-(SP)
023104  104437                                                            TRAP    C$SVEC
023106  062706  000010                                                    ADD     #10,SP
023112                              SETVEC  TXVECA,#CACHTX,#PRI05 ;SET UP INTERRUPT VECTOR TO CATCH TX INT.
023112  012746  000240                                                    MOV     #PRI05,-(SP)
```

```
6049  023116  012746  017544                                                  MOV     #CACHTX,-(SP)
6050  023122  013746  002236                                                  MOV     TXVECA,-(SP)
6051  023126  012746  000003                                                  MOV     #3,-(SP)
6052  023132  104437                                                          TRAP    C$SVEC
6053  023134  062706  000010                                                  ADD     #10,SP
6054  023140
6055  023140  012700  000140                  SETPRI  #PRI03          ;ALLOW DEVICE INTERRUPTS.
                                                                              MOV     #PRI03,R0
6056  023144  104441                                                          TRAP    C$SPRI
6057
6058                                   ;+
6059                                   ;ENABLE RECEPTION INTERRUPTS.
6060                                   ;DELAY 4 MS TO ALLOW TIME FOR THE INTERRUPTS TO TAKE PLACE.
6061                                   ;DISABLE RECEPTION INTERRUPTS.
6062  023146  004737  016174          ;-      JSR     PC,RXIE1        ;ENABLE THE RECEPTION INTERRUPTS.
6063  023152  012704  000004                  MOV     #4,R4           ;PASS 4 MS COUNT TO THE DELAY ROUTINE.
6064  023156  004737  014574                  JSR     PC,DELAY        ;DELAY 4 MILLI-SECONDS.
6065  023162  004737  016134                  JSR     PC,RXIE0        ;DISABLE RECEPTION INTERRUPTS.
6066                                   ;+
6067                                   ; VERIFY THAT THE CORRECT INTERRUPTS TOOK PLACE.
6068                                   ;    TEST THE INT COUNTER TO VERIFY THAT INTERRUPTS TOOK PLACE.
6069                                   ;-
6070  023166  005737  002304                  TST     RXINTC          ;CHECK THE RX INTERRUPT COUNT.
6071  023172  001017                          BNE     6$              ;SKIP THE FOLLOWING ERRORS IF COUNT <> 0.
6072                                   ;+
6073                                   ; DETERMINE REASON FOR NO RX INTERRUPTS AND PRINT PROPER ERROR MESSAGE.
6074                                   ;-
6075  023174  012701  006432                  MOV     #EM2604,R1      ;SET UP MSG IN CASE 'RX.DATA.AVAIL IS CLR".
6076  023200  032777  000200  157040          BIT     #BIT7,@CSRA     ;TEST THE RX.DATA.AVAIL BIT OF THE CSR.
6077  023206  001416                          BEQ     8$              ;GO REPORT ERROR IF RX.DATA.AVAIL IS CLR.
6078  023210  012701  006344                  MOV     #EM2603,R1      ;SET UP MSG IN CASE 'DATA.VALID IS CLEAR".
6079  023214  032777  100000  157026          BIT     #BIT15,@RBUFA   ;TEST THE DATA.VALID BIT OF THE FIFO.
6080  023222  001410                          BEQ     8$              ;GO REPORT ERROR IF DATA.VALID IS CLEAR.
6081  023224  012701  006253                  MOV     #EM2602,R1      ;SET UP MSG,'DATA.VALID IS SET".
6082  023230  000405                          BR      8$              ;GO REPORT THE ERROR.
6083                                   ;+
6084                                   ; IF RX INTS OCCURRED WITH RX.DATA.AVAIL CLEAR, REPORT THE ERROR.
6085                                   ;-
6086  023232  005737  002306          6$:     TST     RXINTF          ;CHECK THE RX INTERRUPT FLAGS.
6087  023236  100006                          BPL     10$             ;SKIP THE ERROR IF FLAG IS CLEAR.
6088  023240  012701  006526                  MOV     #EM2605,R1      ;SET UP THE PROPER MESSAGE.
6089                                   ;+
6090                                   ; REPORT THE ERROR WHICH HAS BEEN FOUND.
6091                                   ;-
6092  023244                          8$:     ERRDF   2603,EM2601,ER0503;             >>>>> ERROR #2603 <<<<<.
6093  023244  104455                                                                  TRAP    C$ERDF
6094  023246  005053                                                                  .WORD   2603
6095  023250  006223                                                                  .WORD   EM2601
6096  023252  012640                                                                  .WORD   ER0503
6097                                   ;+
6098                                   ; VERIFY THAT NO TX INTERRUPTS HAVE BEEN GENERATED SO FAR IN THIS TEST.
6099                                   ;-
6100  023254  013702  002310          10$:    MOV     TXINTC,R2       ;LOAD # OF TX INTERRUPTS FOR ER0504 RTN.
6101  023260  001406                          BEQ     12$             ;SKIP ERROR IF NO TX INTERRUPTS.
6102                                   ;REPORT "TX INTERRUPTS(S) RECEIVED WITH TX INTERRUPTS DISABLED."
6103  023262  012701  005325                  MOV     #EM0526,R1      ;SET UP MESSAGE ADR FOR INDIRECT PRINT.
6104  023266                                  ERRDF   2604,EM2601,ER0504;             >>>>> ERROR #2604 <<<<<.
```

```
                                                                                TRAP    C$ERDF
                                                                                .WORD   2604
                                                                                .WORD   EM2601
                                                                                .WORD   ER0504

                        ;+
                        ; CLEAN OUT THE INTERRUPT VECTORS USED IN THIS TEST.
                        ;-
023276                  12$:    SETPRI  #PRI05          ;DISABLE DEVICE INTERRUPTS.
023276  012700  000240                                                          MOV     #PRI05,R0
023302  104441                                                                 TRAP    C$SPRI
023304                          CLRVEC  RXVECA          ;RETURN RX INT VECTOR TO UNUSED POOL.
023304  013700  002234                                                          MOV     RXVECA,R0
023310  104436                                                                 TRAP    C$CVEC
023312                          CLRVEC  TXVECA          ;RETURN TX INT VECTOR TO UNUSED POOL.
023312  013700  002236                                                          MOV     TXVECA,R0
023316  104436                                                                 TRAP    C$CVEC

                        ;+
                        ; TEST TRANSMISSION INTERRUPTS.
                        ;    SET UP FOR RX AND TX INTERRUPTS:
                        ;    RX INTERRUPT SERVICE ROUTINE COUNTS RX INTERRUPTS.
                        ;    TX INTERRUPT SERVICE ROUTINE COUNTS THE INTERRUPT AND SETS FLAGS.
                        ;-
023320  005037  002304          CLR     RXINTC          ;CLEAR THE RX INTERRUPT COUNTER.
023324  005037  002310          CLR     TXINTC          ;CLEAR THE TX INTERRUPT COUNTER.
023330  005037  002312          CLR     TXINTF          ;CLEAR THE RX INTERRUPT FLAGS.
023334                          SETVEC  RXVECA,#CACHRX,#PRI05 ;SET UP INTERRUPT VECTOR TO CATCH RX INT.
023334  012746  000240                                                          MOV     #PRI05,-(SP)
023340  012746  017516                                                          MOV     #CACHRX,-(SP)
023344  013746  002234                                                          MOV     RXVECA,-(SP)
023350  012746  000003                                                          MOV     #3,-(SP)
023354  104437                                                                 TRAP    C$SVEC
023356  062706  000010                                                          ADD     #10,SP
023362                          SETVEC  TXVECA,#TXINTR,#PRI05 ;SET UP INT VECTOR TO TX INT ROUTINE.
023362  012746  000240                                                          MOV     #PRI05,-(SP)
023366  012746  020040                                                          MOV     #TXINTR,-(SP)
023372  013746  002236                                                          MOV     TXVECA,-(SP)
023376  012746  000003                                                          MOV     #3,-(SP)
023402  104437                                                                 TRAP    C$SVEC
023404  062706  000010                                                          ADD     #10,SP
023410                          SETPRI  #PRI03          ;ALLOW DEVICE INTERRUPTS.
023410  012700  000140                                                          MOV     #PRI03,R0
023414  104441                                                                 TRAP    C$SPRI

                        ;+
                        ; VERIFY THAT THE TX_ACTION BIT IS CLEAR.
                        ;-
023416  012705  000022          MOV     #18.,R5         ;INITIALIZE THE LOOP COUNTER.
023422  012701  000144          MOV     #100.,R1        ;SET 100 MS TIME-OUT.
023426  012702  100000          MOV     #BIT15,R2       ;SELECT TX_ACTION BIT TO TEST.
023432  013704  002246          MOV     CSRA,R4         ;PASS OUT CSR AS THE WORD TO TEST.
023436  012703  100000  14$:    MOV     #BIT15,R3       ;WAIT FOR TX_ACTION TO BE SET.
023442  004737  015332          JSR     PC,MSLOOP       ;WAIT UP TO 100 MS FOR TX_ACTION SET.
023446  103020                  BCC     20$             ;IF TIME-OUT, CONSIDER TX_ACTION CLEAR.
023450  005003                   CLR    R3              ;NOW, WAIT FOR TX_ACTION CLEAR.
023452  004737  015332          JSR     PC,MSLOOP       ;WAIT UP TO 100 MS FOR TX_ACTION CLEAR.
023456  103005                  BCC     16$             ;IF TIME-OUT, REPORT TX_ACTION WON'T CLEAR.
023460  005305                   DEC    R5              ;DECREMENT THE TX_ACTION SET COUNTER.
```

```
6161  023462  001365                        BNE     14$             ;LOOP IF NOT TOO MANY TX_ACTIONS FOUND.
6162                                    ;REPORT "TX_ACTION SET REPEATEDLY AFTER RESET, NO DATA SENT."
6163  023464  012701  006650                MOV     #EM2607,R1      ;SELECT ERROR MESSAGE.
6164  023470  000402                        BR      18$             ;GO TO REPORT THE ERROR.
6165  023472  012701  006744          16$:   MOV     #EM2608,R1      ;SELECT TX_ACTION STUCK SET MSG.
6166  023476                          18$:   ERRDF   2605,EM2606,ER0503;          >>>>> ERROR #2605 <<<<<.
6167  023476  104455                                                                TRAP    C$ERDF
6168  023500  005055                                                                .WORD   2605
6169  023502  006611                                                                .WORD   EM2606
6170  023504  012640                                                                .WORD   ER0503
6171  023506  000424                        BR      24$             ;GO TO TEST WITH TX_ACTION SET.
6172                                        ;+
6173                                        ; VERIFY THAT NO INTERRUPTS OCCUR WITH TX_ACTION CLEAR.
6174                                        ;-
6175  023510  004737  017042          20$:   JSR     PC,TXIE1        ;ENABLE TX_INTERRUPTS.
6176  023514  012704  000062                MOV     #50.,R4         ;PASS 50 MS TIME TO THE DELAY ROUTINE.
6177  023520  004737  014574                JSR     PC,DELAY        ;DELAY 50 MILLI-SECONDS TO ALLOW INTS TO OCCUR.
6178  023524  005737  002310                TST     TXINTC          ;TEST THE TX INTERRUPT COUNT.
6179  023530  001413                        BEQ     24$             ;SKIP THE ERROR IF NO TX INTERRUPTS.
6180  023532  012701  006650                MOV     #EM2607,R1      ;SELECT MESSAGE IN CASE TX INT FLAG CLEAR.
6181  023536  005737  002312                TST     TXINTF          ;TEST THE TX INTERRUPT FLAGS.
6182  023542  100002                        BPL     22$             ;GO REPORT ERROR IF TX FLAG IS CLEAR.
6183  023544  012701  007015                MOV     #EM2609,R1      ;TX FLAG IS SET, SELECT PROPER ERROR MESSAGE.
6184                                    ;REPORT "TRANSMIT INTERRUPT TEST ERROR:..."
6185  023550                          22$:   ERRDF   2606,EM2606,ER0503;          >>>>> ERROR #2606 <<<<<.
6186  023550  104455                                                                TRAP    C$ERDF
6187  023552  005056                                                                .WORD   2606
6188  023554  006611                                                                .WORD   EM2606
6189  023556  012640                                                                .WORD   ER0503
6190                                        ;+
6191                                        ; PREPARE TX INTERRUPT COUNTER AND FLAGS.
6192                                        ;-
6193  023560  005037  002310          24$:   CLR     TXINTC          ;CLEAR THE TX INTERRUPT COUNT.
6194  023564  005037  002312                CLR     TXINTF          ;CLEAR THE TX INTERRUPT FLAGS.
6195                                        ;+
6196                                        ; SET UP LINE PARAMETERS FOR TRANSMISSION.
6197                                        ;-
6198  023570  012705  000377                MOV     #MAPLNS,R5      ;PASS ACTIVE LINES BIT MAP.
6199  023574  012700  000200                MOV     #200,R0         ;PASS INERT STATE, INTERNAL LOOPBACK.
6200  023600  004737  017412                JSR     PC,WTWLNC       ;DISABLE RECPTION AND DMA, ETC. ON DUT.
6201  023604  012700  156430                MOV     #156430,R0      ;SPECIFY 9600BPS,1STOP,NO PARITY,8BITS/CHAR.
6202  023610  004737  017466                JSR     PC,WTWLPR       ;WRITE TO ALL LPR REGISTERS.
6203                                        ;+
6204                                        ; SEND A NULL CHAR TO EACH LINE.
6205                                        ;-
6206  023614  013701  002250                MOV     TXCHA,R1        ;SET UP TXCHAR REGISTER ADDRESS.
6207  023620  012702  100000                MOV     #100000,R2      ;SET CHARACTER TO BE TRANSMITTED = NULL.
6208  023624  004737  017442                JSR     PC,WTWLNS       ;SEND NULL CHAR TO EACH LINE.
6209                                        ;+
6210                                        ; DELAY 250 MILLI-SECONDS TO ALLOW INTERRUPTS TO OCCUR.
6211                                        ;-
6212  023630  012704  000372                MOV     #250.,R4        ;SET UP FOR 250 MS DELAY.
6213  023634  004737  014574                JSR     PC,DELAY        ;WAIT 250 MS.
6214                                        ;+
6215                                        ; VERIFY THAT TX INTERRUPTS OCCURRED.
6216                                        ;-
```

```
6217  023640  005737  002310                  TST     TXINTC          ;CHECK THE TX INTERRUPT COUNTER.
6218  023644  001007                          BNE     26$             ;SKIP THE FOLLOWING ERROR IF WE GOT TX INTS.
6219                                          ;+
6220                                          ; DETERMINE THE REASON THAT WE RECEIVED NO INTERRUPTS.
6221                                          ;-
6222  023646  012701  007074                  MOV     #EM2610,R1      ;SET UP MSG IN CASE 'TX_ACTION IS SET'.
6223  023652  005777  156370                  TST     @CSRA           ;CHECK THE DUT CSR.
6224  023656  100407                          BMI     28$             ;GO TO REPORT ERROR IF TX_ACTION IS SET.
6225  023660  012701  007166                  MOV     #EM2611,R1      ;SET UP 'TX_ACTION NOT SET' MESSAGE.
6226                                          ;+
6227                                          ; CHECK TO VERIFY THAT TX_ACTION WAS SET FOR EACH INTERRUPT.
6228                                          ;-
6229  023664  005737  002312          26$:    TST     TXINTF          ;CHECK THE TX INTERRUPT FLAGS.
6230  023670  100006                          BPL     30$             ;SKIP ERROR IF TX_ACTION CLR FLAG IS CLEAR.
6231  023672  012701  007015                  MOV     #EM2609,R1      ;SET UP TX INT WITH 'TX_ACTION CLR' MSG.
6232                                          ;+
6233                                          ; REPORT 'TRANSMIT INTERRUPT TEST ERROR:....'
6234                                          ;-
6235  023676                          28$:    ERRDF   2607,EM2606,ER0503;             >>>>> ERROR #2607 <<<<<.
6236  023676  104455                                                                          TRAP    C$ERDF
6237  023700  005057                                                                          .WORD   2607
6238  023702  006611                                                                          .WORD   EM2606
6239  023704  012640                                                                          .WORD   ER0503
6240                                          ;+
6241                                          ; VERIFY THAT NO TX INTERRUPTS HAVE BEEN GENERATED SO FAR IN THIS TEST.
6242                                          ;-
6243  023706  013702  002304          30$:    MOV     RXINTC,R2       ;LOAD # OF RX INTERRUPTS FOR ER0504 RTN.
6244  023712  001406                          BEQ     32$             ;SKIP ERROR IF NO RX INTERRUPTS.
6245  023714  012701  005235                  MOV     #EM0525,R1      ;SET UP MESSAGE ADR FOR INDIRECT PRINT.
6246                                      ;REPORT 'RX INTERRUPTS(S) RECEIVED WITH RX INTERRUPTS DISABLED.'
6247  023720                                  ERRDF   2608,EM2606,ER0504;             >>>>> ERROR #2608 <<<<<.
6248  023720  104455                                                                          TRAP    C$ERDF
6249  023722  005060                                                                          .WORD   2608
6250  023724  006611                                                                          .WORD   EM2606
6251  023726  012664                                                                          .WORD   ER0504
6252                                          ;+
6253                                          ; DISABLE INTERRUPTS AND CLEAN OUT THE INTERRUPT VECTORS USED IN THIS TEST.
6254                                          ;-
6255  023730  005001          32$:    CLR     R1              ;CLEAR BOTH TRANSMITTER
6256  023732  004737  017002          JSR     PC,TXIE0        ; INTERRUPT ENABLE AND RECEIVER
6257  023736  004737  016134          JSR     PC,RXIE0        ; INTERRUPT ENABLE BITS IN THE DUT CSR.
6258  023742                          SETPRI  #PRI05          ;DISABLE DEVICE INTERRUPTS.
6259  023742  012700  000240                                                          MOV     #PRI05,R0
6260  023746  104441                                                                  TRAP    C$SPRI
6261  023750                          CLRVEC  RXVECA          ;RETURN RX INT VECTOR TO UNUSED POOL.
6262  023750  013700  002234                                                          MOV     RXVECA,R0
6263  023754  104436                                                                  TRAP    C$CVEC
6264  023756                          CLRVEC  TXVECA          ;RETURN TX INT VECTOR TO UNUSED POOL.
6265  023756  013700  002236                                                          MOV     TXVECA,R0
6266  023762  104436                                                                  TRAP    C$CVEC
6267
6268  023764  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6269  023770                          SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
6270  023770  012700  000340                                                          MOV     #PRI07,R0
6271  023774  104441                                                                  TRAP    C$SPRI
6272  023776                          ENDTST
```

```
 6273  023776
 6274  023776  104401
 6275
                                                        L10030:
                                                                TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST           - BRLEVA -
                        ;++ ************************************************************************
                        ;*                      - BR LEVEL TEST B -
                        ;*      THIS TEST VERIFIES THAT THE DEVICE UNDER TEST (DUT) WILL GENERATE
                        ;*      RECEPTION AND TRANSMISSION INTERRUPTS AT THE CORRECT BR LEVEL.
                        ;*      THIS TEST DOES NOT DEPEND ON THE USE OF THE SERIAL LINE TRANSMISSION
                        ;*      OR RECEPTION CAPABILITIES OF THE DUT.  THE LINES ARE PUT IN INTERNAL
                        ;*      LOOPBACK TO MINIMIZE ANY EXTERNAL EFFECTS THAT COULD BE CAUSED ON
                        ;*      DEVICES ATTACHED TO THE SERIAL LINES.
                        ;*
                        ;-- ************************************************************************

                            BGNTST
                                                                        T7::
                            SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
024000                                                                  MOV     #PRI05,R0
024000  012700  000240                                                  TRAP    C$SPRI
024004  104441
                         TNUM == TNUM + 1           ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
000007
024006  012737  000007  002272  MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (30)
024014  012737  177777  002270  MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
024022  012737  000001  004052  MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
024030  012737  005671  004054  MOV     #3001.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
024036  012737  007251  004056  MOV     #EM3001,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
024044  005037  002446          CLR     ERSMRF          ;INITIALIZE THE 'REPORT ERROR SUMMARY'' FLAGS.
                        ;+
                        ; RESET THE DUT TO A KNOWN STATE, DO NOT REMOVE THE STATUS CODES FROM THE FIFO.
                        ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                        ; THIS SUBROUTINE REPORTS ERRORS FROM >>>>> 3001 THRU 3002 <<<<<.
                        ;-
024050  004737  016022          JSR     PC,RESETT       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
024054  103402                  BCS     2$              ;SKIP AROUND ABORTING TEST IF NO ERROR FOUND.
024056  000137  024654          JMP     60$             ;ABORT TEST IF FATAL ERROR FOUND DURING RESET.
024062  012737  005673  004054  2$:     MOV     #3003.,ERRNBR   ;SET THE ERROR REPORT NUMBER TO 3003.
                        ;+
                        ; ENABLE TRANSMITTERS ON ALL LINES.
                        ;-
024070  012705  000377  4$:     MOV     #MAPLNS,R5      ;PASS ACTIVE LINE BIT MAP.
024074  004737  016706          JSR     PC,TXENBL       ;ENABLE TRANSMISSION ON ALL LINES.
                        ;+
                        ; GENERATE A TRANSMISSION INTERRUPT REQUEST.
                        ; PROCESSOR PRIORITY SHOULD BE AT 7 DISABLING INTS.
                        ;-
                            SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
024100                                                                  MOV     #PRI07,R0
024100  012700  000340                                                  TRAP    C$SPRI
024104  104441
                            SETVEC  TXVECA,#TXINTR,#PRI07 ;SET UP INTERRUPT VECTOR TO CATCH TX INT.
024106                                                                  MOV     #PRI07,-(SP)
024106  012746  000340                                                  MOV     #TXINTR,-(SP)
024112  012746  020040                                                  MOV     TXVECA,-(SP)
024116  013746  002236                                                  MOV     #3,-(SP)
024122  012746  000003                                                  TRAP    C$SVEC
024126  104437                                                          ADD     #10,SP
024130  062706  000010
                        ;+
                        ; SET UP DUT FOR TRANSMISSION INTERRUPTS:
                        ;  SET UP INTERNAL LOOPBACK.
                        ;  SET UP LINE PARAMETERS FOR TRANSMISSION.
```

```
6332
6333  024134  012705  000377          ;-
6334  024140  012700  000200              MOV     #MAPLNS,R5      ;PASS ACTIVE LINES BIT MASK.
6335  024144  004737  017412              MOV     #200,R0         ;PASS INERT STATE, INTERNAL LOOPBACK.
6336  024150  012700  156430             JSR     PC,WTWLNC       ;DISABLE RECPTION AND DMA, ETC. ON DUT.
6337  024154  004737  017466              MOV     #156430,R0      ;SPECIFY 9600BPS,1STOP,NO PARITY,8BITS/CHAR.
6338                                      JSR     PC,WTWLPR       ;WRITE INTO ALL LPR REGISTERS.
6339                                  ;+
6340                                  ; SEND A NULL CHAR TO EACH LINE.
6341  024160  013701  002250          ;-
6342  024164  012702  100000              MOV     TXCHA,R1        ;SET UP TXCHAR REGISTER ADDRESS.
6343  024170  004737  017442              MOV     #100000,R2      ;SET CHARACTER TO BE TRANSMITTED = NULL.
6344                                     JSR     PC,WTWLNS       ;SEND NULL CHAR TO EACH LINE.
6345                                  ;+
6346                                  ; DELAY 50 MS TO ALLOW TIME FOR THE INTERRUPT TO BE GENERATED.
6347  024174  012704  000062          ;-
6348  024200  004737  014574              MOV     #50.,R4         ;PASS 50 MS TIME TO THE DELAY ROUTINE.
6349                                     JSR     PC,DELAY        ;DELAY 50 MILLI-SECONDS.
6350                                ;+
6351                                ; GENERATE A RECEPTION INTERRUPT REQUEST.
6352  024204                        ;-
6353  024204  012746  000340           SETVEC  RXVECA,#RXBRRT,#PRI07 ;SET UP INTERRUPT VECTOR TO CATCH RX INT.
6354  024210  012746  017642                                                 MOV     #PRI07,-(SP)
6355  024214  013746  002234                                                 MOV     #RXBRRT,-(SP)
6356  024220  012746  000003                                                 MOV     RXVECA,-(SP)
6357  024224  104437                                                         MOV     #3,-(SP)
6358  024226  062706  000010                                                 TRAP    C$SVEC
6359                                                                         ADD     #10,SP
6360                                ;+
6361                                ; SET UP FOR THE LOOP WHICH TESTS THE INTERRUPT BR LEVELS.
6362  024232  012705  000340        ;-
6363  024236  005003                     MOV     #340,R5         ;SET UP THE PRIORITY LEVEL TO 7.
6364  024240  005002                     CLR     R3              ;CLEAR THE RX PRIORITY STORE AND FLAGS.
6365                                      CLR     R2              ;CLEAR THE TX PRIORITY STORE AND FLAGS.
6366                                  ;+
6367                                  ; ENABLE TX AND RX INTERRUPTS.
6368                                  ; PROCESSOR PRIORITY SHOULD BE AT 7 DISABLING THE INTERRUPTS.
6369  024242  004737  016174          ;-
6370  024246  004737  017042              JSR     PC,RXIE1        ;ENABLE RECEIVER INTERRUPTS.
6371                                     JSR     PC,TXIE1        ;ENABLE TRANSMITTER INTERRUPTS.
6372                                ;+
6373                                ; LOOP, LOWERING THE PROCESSOR PRIORITY UNTIL THE DUT INTERRUPTS ON RX AND TX.
6374  024252  005037  002310        ;-
6375  024256  005037  002312  6$:       CLR     TXINTC          ;CLEAR THE TX INTERRUPT COUNTER.
6376  024262  005037  002304             CLR     TXINTF          ;CLEAR THE TX INTERRUPT FLAGS.
6377  024266  005037  002306             CLR     RXINTC          ;CLEAR THE RX INTERRUPT COUNTER.
6378  024272                             CLR     RXINTF          ;CLEAR THE RX INTERRUPT FLAGS.
6379  024272  010500                    SETPRI  R5              ;SET PROCESSOR PRIORITY TO THE SELECTED VALUE.
6380  024274  104441                                                    MOV     R5,R0
6381  024276  012704  000001                                           TRAP    C$SPRI
6382  024302  004737  014574             MOV     #1,R4           ;PASS 1 MS COUNT TO THE DELAY ROUTINE.
6383                                     JSR     PC,DELAY        ;DELAY 1 MS TO ALLOW INTERRUPTS TO OCCUR.
6384                                 ;+
6385                                 ; DETERMINE IF ANY RX DUT INTERRUPTS OCCURRED.
6386                                 ; LOG THE PROCESSOR PRIORITY FOR THE RX INTERRUPT IF FIRST RX INT.
6387  024306  005737  002304        ;-
                                         TST     RXINTC          ;CHECK THE RECEIVE INTERRUPT COUNTER.
```

```
6388  024312  001412                          BEQ     8$              ;SKIP THE PRIORITY LOG IF NO RX INT OCCURRED.
6389                                          ;+
6390                                          ; IF THIS IS THE FIRST RX INTERRUPT, LOG THE PRIORITY.
6391                                          ;-
6392  024314  005703                           TST     R3              ;CHECK THE RX PRIORITY STORE AND FLAGS.
6393  024316  001010                          BNE     8$              ;GOTO TEST FOR TX INTS IF NOT THE FIRST RX INT.
6394  024320  010503                           MOV     R5,R3           ;LOG THE PRESENT PRIORITY IN THE RX PRIO STORE.
6395  024322  052703  100000                   BIS     #BIT15,R3       ;SET THE RX INT HAS OCCURRED FLAG.
6396  024326  013700  002306                   MOV     RXINTF,R0       ;GET THE RX INTERRUPT ROUTINE FLAGS.
6397  024332  042700  137777                   BIC     #137777,R0      ;CLEAR ALL BUT THE TX INT ERROR FLAG.
6398  024336  050003                           BIS     R0,R3           ;IF TX INT ERROR, SET BIT 14 OF THE PRIO FLAGS.
6399                                      ;+
6400                                      ; DETERMINE IF ANY TX DUT INTERRUPTS HAVE OCCURRED.
6401                                      ; LOG THE PRESENT PROCESSOR PRIORITY IF THIS IS THE FIRST TX INTERRUPT.
6402                                      ;-
6403  024340  005737  002310          8$:     TST     TXINTC          ;CHECK THE TRANSMIT INTERRUPT COUNTER.
6404  024344  001405                          BEQ     10$             ;SKIP THE PRIORITY LOG IF NO TX INT OCCURRED.
6405                                          ;+
6406                                          ; IF THIS IS THE FIRST TX INTERRUPT, LOG THE PRIORITY.
6407                                          ;-
6408  024346  005702                           TST     R2              ;CHECK THE TX PRIORITY STORE AND FLAGS.
6409  024350  100403                          BMI     10$             ;SKIP THE LOGGING IF NOT FIRST TX INTERRUPT.
6410  024352  010502                           MOV     R5,R2           ;LOG THE PRESENT PRIORITY IN THE TX PRIO STORE.
6411  024354  052702  100000                   BIS     #BIT15,R2       ;SET THE TX INT HAS OCCURRED FLAG.
6412                                      ;+
6413                                      ; SELECT NEXT PROCESSOR PRIORITY.
6414                                      ; TEST FOR BOTH RX AND TX INTERRUPTS HAVING OCCURRED, LOOP IF NOT.
6415                                      ;-
6416  024360  162705  000040         10$:     SUB     #40,R5          ;DECREMENT PRIORITY LEVEL BY ONE.
6417  024364  002402                          BLT     12$             ;GOTO CHECK FOR ERRORS IF BELOW PRIORITY ZERO.
6418  024366  030203                           BIT     R2,R3           ;AND PRIO FLAGS TOGETHER, ALTER NONE OF THEM.
6419  024370  100330                          BPL     6$              ;LOOP IF RX AND TX INTS HAVEN'T BOTH OCCURRED.
6420                                      ;+
6421                                      ; DISABLE INTERRUPTS AND CLEAR INTERRUPT VECTORS.
6422                                      ;-
6423  024372                          12$:    SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
6424  024372  012700  000340                                                          MOV     #PRI07,R0
6425  024376  104441                                                                  TRAP    C$SPRI
6426  024400                                  CLRVEC  RXVECA          ;RETURN RX INT VECTOR TO UNUSED POOL.
6427  024400  013700  002234                                                          MOV     RXVECA,R0
6428  024404  104436                                                                  TRAP    C$CVEC
6429  024406                                  CLRVEC  TXVECA          ;RETURN TX INT VECTOR TO UNUSED POOL.
6430  024406  013700  002236                                                          MOV     TXVECA,R0
6431  024412  104436                                                                  TRAP    C$CVEC
6432                                      ;+
6433                                      ; VERIFY THAT RX AND TX INTERRUPTS OCCURRED,
6434                                      ; AT THE PROPER BR LEVEL, AND
6435                                      ; IN THE PROPER ORDER.
6436                                      ;    DETERMINE IF TX INTERRUPT OCCURRED.
6437                                      ;-
6438  024414  005702                           TST     R2              ;DETERMINE WHETHER TX INT OCCURRED OR NOT.
6439  024416  100414                          BMI     16$             ;SKIP THESE ERRORS IF TX INT OCCURRED.
6440                                      ;+
6441                                      ; DETERMINE REASON THAT NO TX INT OCCURRED.
6442                                      ;-
6443  024420  012701  007074                   MOV     #EM2610,R1      ;SELECT "NO TX INT FROM TX.ACTION" MESSAGE.
```

```
6444  024424  005777  155616                  TST     @CSRA           ;CHECK THE TX.ACTION BIT OF THE DUT CSR.
6445  024430  100402                          BMI     14$             ;SKIP TX.ACTION CLR MSG SELECTION IF IT IS SET.
6446  024432  012701  007166                  MOV     #EM2611,R1      ;SELECT "TX.ACTION CLEAR AFTER CHARS SENT" MSG.
6447                                      ;REPORT "INTERRUPT BR LEVEL TEST ERROR:"
6448  024436                            14$:  ERRDF   3003,EM3001,ER0503;           >>>>> ERROR #3003 <<<<<.
6449  024436  104455                                                                TRAP    C$ERDF
6450  024440  005673                                                                .WORD   3003
6451  024442  007251                                                                .WORD   EM3001
6452  024444  012640                                                                .WORD   ER0503
6453  024446  000423                          BR      18$             ;SKIP THE BR LEVEL CHECK, NO TX INT OCCURRED.
6454                                          ;+
6455                                          ; VERIFY THAT THE TX INTERRUPT WAS AT THE PROPER BR LEVEL.
6456                                          ;-
6457  024450  010204                    16$:  MOV     R2,R4           ;CALCULATE THE BR LEVEL
6458  024452  042704  177400                 BIC     #177400,R4      ; THAT THE TRANMSIT
6459  024456  006204                          ASR     R4              ; INTERRUPT WAS
6460  024460  006204                          ASR     R4              ; REQUESTED AT, WHICH
6461  024462  006204                          ASR     R4              ; IS ONE GREATER THAN
6462  024464  006204                          ASR     R4              ; THE PROCESSOR PRIORITY
6463  024466  006204                          ASR     R4              ; LEVEL AT WHICH THE
6464  024470  005204                          INC     R4              ; TRANSMIT INTERRUPT OCCURRED.
6465  024472  113705  002243                  MOVB    BRLEVL,R5       ;GET THE EXPECTED INTERRUPT BR LEVEL.
6466  024476  120405                          CMPB    R4,R5           ;COMARE THE INTERRUPT BR LEVEL WITH EXPECTED.
6467  024500  001406                          BEQ     18$             ;SKIP THE ERROR IF BR LEVEL IS CORRECT.
6468                                      ;REPORT "TX INTERRUPT GENERATED AT WRONG BR LEVEL: ..."
6469  024502  012701  007376                  MOV     #EM3003,R1      ;SELECT THE ERROR MESSAGE FOR THE ERROR CALL.
6470  024506                                  ERRDF   3004,EM3001,ER3001;           >>>>> ERROR #3004 <<<<<.
6471  024506  104455                                                                TRAP    C$ERDF
6472  024510  005674                                                                .WORD   3004
6473  024512  007251                                                                .WORD   EM3001
6474  024514  013012                                                                .WORD   ER3001
6475                                          ;+
6476                                          ; DETERMINE IF RX INTERRUPT OCCURRED.
6477                                          ;-
6478  024516  005703                    18$:  TST     R3              ;CHECK THE RX INT OCCURRED FLAG.
6479  024520  100415                          BMI     22$             ;SKIP THESE ERRORS IF RX INT OCCURRED.
6480                                          ;+
6481                                          ; DETERMINE REASON THAT NO RX INT OCCURRED.
6482                                          ;-
6483  024522  012701  006253                  MOV     #EM2602,R1      ;SELECT "NO RX INT FROM TX.ACTION" MSG.
6484  024526  032777  000200  155512          BIT     #BIT7,@CSRA     ;CHECK THE RX.DATA.AVAIL BIT OF THE DUT CSR.
6485  024534  001002                          BNE     20$             ;SKIP RX.DATA.AVAIL CLR MSG IF BIT IS SET.
6486  024536  012701  007302                  MOV     #EM3002,R1      ;SELECT "NO RX.DATA.AVAIL AFTER RESET" MSG.
6487                                      ;REPORT "INTERRUPT BR LEVEL TEST ERROR:"
6488  024542                            20$:  ERRDF   3005,EM3001,ER0503;           >>>>> ERROR #3005 <<<<<.
6489  024542  104455                                                                TRAP    C$ERDF
6490  024544  005675                                                                .WORD   3005
6491  024546  007251                                                                .WORD   EM3001
6492  024550  012640                                                                .WORD   ER0503
6493  024552  000423                          BR      24$             ;SKIP THE BR CHECK IF NO RX INT OCCURRED.
6494                                          ;+
6495                                          ; VERIFY THAT THE RX INTERRUPT WAS AT THE PROPER BR LEVEL.
6496                                          ;-
6497  024554  010304                    22$:  MOV     R3,R4           ;CALCULATE THE BR LEVEL
6498  024556  042704  177400                 BIC     #177400,R4      ; THAT THE RECEIVE
6499  024562  006204                          ASR     R4              ; INTERRUPT WAS
```

```
6500  024564  006204                          ASR     R4              ; REQUESTED AT, WHICH
6501  024566  006204                          ASR     R4              ; IS ONE GREATER THAN
6502  024570  006204                          ASR     R4              ; THE PROCESSOR PRIORITY
6503  024572  006204                          ASR     R4              ; LEVEL AT WHICH THE
6504  024574  005204                          INC     R4              ; RECEIVE INTERRUPT OCCURRED.
6505  024576  113705  002243                  MOVB    BRLEVL,R5       ;GET THE EXPECTED INTERRUPT BR LEVEL.
6506  024602  120405                          CMPB    R4,R5           ;COMARE THE INTERRUPT BR LEVEL WITH EXPECTED.
6507  024604  001406                        BEQ     24$             ;SKIP THE ERROR IF BR LEVEL IS CORRECT.
6508                                      ;REPORT "RX INTERRUPT GENERATED AT WRONG BR LEVEL: ..."
6509  024606  012701  007452                    MOV     #EM3004,R1      ;SELECT ERROR MESSAGE FOR THE ERROR CALL.
6510  024612                                  ERRDF   3006,EM3001,ER3001;             >>>>> ERROR #3006 <<<<<.
6511  024612  104455                                                                          TRAP    C$ERDF
6512  024614  005676                                                                          .WORD   3006
6513  024616  007251                                                                          .WORD   EM3001
6514  024620  013012                                                                          .WORD   ER3001
6515
6516                                        ;+
6517                                        ; TEST FOR INTERRUPTS OCCURING IN THE PROPER ORDER.
6518                                        ;-
6519  024622  032703  040000          24$:     BIT     #BIT14,R3       ;CHECK THE IMPROPER INT ORDER ERROR FLAG.
6520  024626  001406                        BEQ     26$             ;SKIP ERROR REPORT IF ERROR DID NOT OCCUR.
6521                                      ;REPORT "TX INTERRUPT GIVEN PRECEDENCE OVER SIMULTANEOUS RX INT."
6522  024630  012701  007526                    MOV     #EM3005,R1      ;SELECT THE ERROR MESSAGE FOR INDIRECT PRINT.
6523  024634                                  ERRDF   3007,EM3001,ER0503;             >>>>> ERROR #3007 <<<<<.
6524  024634  104455                                                                          TRAP    C$ERDF
6525  024636  005677                                                                          .WORD   3007
6526  024640  007251                                                                          .WORD   EM3001
6527  024642  012640                                                                          .WORD   ER0503
6528
6529                                        ;+
6530                                        ; CLEAN UP, EXIT THE TEST.
6531  024644  004737  017002                ;-
6532  024650  004737  016134          26$:     JSR     PC,TXIE0        ;CLEAR TRANSMITTER INTERRUPTS.
6533  024654  005037  002270                JSR     PC,RXIE0        ;CLEAR RECEIVER INTERRUPTS.
6534  024660                          60$:     CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6535  024660  012700  000340                SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
6536  024664  104441                                                                          MOV     #PRI07,R0
6537  024666                                                                                  TRAP    C$SPRI
6538  024666                                  ENDTST
6539  024666  104401                                                                  L10031:
6540                                                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - DIABMP -
                                ;++ ************************************************************************
                                ;*                      - DIAGNOSTIC FIELD (BMP) TEST -
                                ;*      THIS TEST VERIFIES THAT A REQUEST TO THE DUT TO REPORT BMP STATUS
                                ;*      CODES IS COMPLIED WITH, WITHIN THE SPECIFIED TIME.
                                ;*      ALL ACTIVE LINES ARE TESTED.
                                ;-- ************************************************************************

024670                                  BGNTST
024670                                                                          T8::
024670                                  SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
024670  012700  000240                                                        MOV     #PRI05,R0
024674  104441                                                                TRAP    C$SPRI
        000010
024676  012737  000010  002272           TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
                                        MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (31)
024704  012737  177777  002270           MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
024712  012737  000001  004052           MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
024720  012737  006035  004054           MOV     #3101.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
024726  012737  007620  004056           MOV     #EM3101,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
024734  012737  013544  004060           MOV     #ER9101,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 3101 <<<<<.
                                ;-
024742  004737  014460                  JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
024746  103077                        BCC     60$             ;RESET FAILURE?, ABORT THIS TEST.
                                ;+
                                ; TEST ALL ACTIVE LINES INDIVIDUALLY.
                                ; WRITE THE REQUEST CODE TO THE DIAGNOSTIC FIELD IN THE LPR REGISTER.
                                ; VERIFY THAT A BMP CODE IS RETURNED WITHIN THE CORRECT TIME.
                                ;-
024750  013705  002240                  MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
024754  005004                        CLR     R4              ;CLEAR THE LINE NUMBER COUNTER.
024756  013703  002246                  MOV     CSRA,R3         ;GET THE ADDRESS OF THE DUT'S CSR.
024762  000241                2$:     CLC                     ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
024764  006005                        ROR     R5              ;SHIFT THE BIT MAP INTO THE CARRY BIT.
024766  103064                      BCC     8$                ;DO NOT TEST THE LINE IF IT IS INACTIVE.
                                ;+
                                ; SELECT THE LINE UNDER TEST.
                                ;  WRITE THE BMP REQUEST CODE TO THE DIAG FIELD IN THE LPR REGISTER.
                                ;-
024770  012737  006036  004054          MOV     #3102.,ERRNBR   ;SET THE ERROR NUMBER TO 3102.
024776  010413                        MOV     R4,(R3)         ;SELECT THE LINE CURRENTLY UNDER TEST.
025000  052777  000002  155244        BIS     #2,@LPRA        ;WRITE THE BMP REQUEST CODE TO THE LPR.
                                ;+
                                ; WAIT FOR BMP REQUEST CODE TO BE CLEARED, REPORT ERROR IF TIME-OUT
                                ; OCCURS.
                                ;-
025006  012701  010764                  MOV     #10764,R1       ;TEST BIT 1, TIMEOUT OF 500 MILLI SECS.
025012  013702  002252                  MOV     LPRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
025016  004737  017222                JSR     PC,WAIBIC         ;WAIT FOR REQUEST CODE TO CLEAR.
025022  103042                      BCC     6$                ;GO REPORT ERROR IF CODE DID NOT CLEAR IN TIME.
                                ;+
                                ; WAIT FOR BMP CODE TO APPEAR IN THE FIFO, REPORT ERROR IF TIME-OUT
                                ; OCCURS.
```

```
6597
6598  025024  005237  004054                ;-
6599  025030  012701  070012                 INC     ERRNBR          ;SET ERROR NUMBER TO 3103.
6600  025034  013702  002246                 MOV     #70012,R1       ;TEST BIT 7, TIMEOUT OF 10 MILLI SECS.
6601  025040  004737  017276                 MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
6602  025044  103031                        JSR     PC,WAIBIS       ;WAIT FOR RX DATA AVAILABLE TO SET.
6603                                         BCC     6$              ;GO REPORT ERROR IF CODE DID NOT CLEAR IN TIME.
6604                                        ;+
6605                                        ; READ THE BMP CODE (IF IT IS THERE) FROM THE RBUF REGISTER.
6606                                        ; DETERMINE IF IT IS A VALID BMP CODE,
6607                                        ; VERIFY THE BMP CODE WAS RECEIVED FROM THE CORRECT CHANNEL.
6608                                        ; IF THE BMP CODE DOES NOT INDICATE DUT RUNNING OK, THEN SAVE IT ON
6609                                        ; THE QUEUE TO BE REPORTED IN A LATER TEST.
6610  025046  005237  004054                ;-
6611  025052  017702  155172                 INC     ERRNBR          ;SET ERROR NUMBER TO 3104.
6612  025056  100024                         MOV     @RBUFA,R2       ;GET THE BMP CODE FROM THE FIFO.
6613  025060  005237  004054                BPL     6$              ;GO REPORT ERROR IF NO BMP CODE FOUND.
6614  025064  012700  170301                 INC     ERRNBR          ;SET ERROR NUMBER TO 3105.
6615  025070  040200                         MOV     #170301,R0      ;SET-UP A BMP CODE MASK.
6616  025072  001016                         BIC     R2,R0           ;TRY TO CLEAR THE BMP MASK.
6617  025074  005237  004054                BNE     6$              ;GO REPORT ERROR IF IT IS NOT A VALID BMP CODE.
6618  025100  010200                         INC     ERRNBR          ;SET THE ERROR NUMBER TO 3106.
6619  025102  000300                         MOV     R2,R0           ;COPY THE BMP CODE.
6620  025104  042700  177760                 SWAB    R0              ;PUT THE LINE NUMBER IN THE LOW BYTE.
6621  025110  120400                         BIC     #177760,R0      ;CLEAR THE UNWANTED BITS.
6622  025112  001006                         CMPB    R4,R0           ;DID THE BMP CODE COME FROM THE CORRECT LINE?.
6623  025114  120227  000305                BNE     6$              ;NO; GO REPORT ERROR.
6624  025120  001407                         CMPB    R2,#305         ;IS THE BMP CODE A "GOOD ONE"?.
6625  025122  004737  016220                BEQ     8$              ;YES; SKIP SAVING THE BMP CODE ON THE QUEUE.
6626  025126  000404                         JSR     PC,SAVBMP       ;SAVE THE BMP CODE ON THE QUEUE.
6627                                         BR      8$              ;GO SEE IF THERE ARE ANY MORE LINE TO TEST.
6628  025130  010401              6$:         MOV     R4,R1           ;PASS THE LINE NUMBER TO BE REPORTED.
6629  025132  012702  007654                 MOV     #EM3102,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
6630                                                                 ;"BMP REQUEST BIT BAD ON LINE:"
6631  025136                                 ERROR                   ;                >>>>> ERROR <<<<<.
6632  025136  104460                                                                    TRAP    C$ERROR
6633                                        ;+
6634                                        ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
6635                                        ;-
6636  025140  005204              8$:         INC     R4              ;INCREMENT THE LINE NUMBER COUNTER.
6637  025142  005705                         TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
6638  025144  001306                         BNE     2$              ;YES; BRANCH TO TEST THE NEXT LINE.
6639  025146  005037  002270        60$:      CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6640  025152                                 ENDTST
6641  025152                                                                     L10032:
6642  025152  104401                                                                    TRAP    C$ETST
```

```
6643
6644                                    .SBTTL  HARDWARE TEST           - DMASTA -
6645                                    ;++ ************************************************************************
6646                                    ;*                      - DMA START BIT TEST -
6647                                    ;*      THIS TEST VERIFIES THAT THE DMA START BIT IN THE DUT'S LINE CONTROL
6648                                    ;*      REGISTERS WILL INITIATE DMA TRANSMISSION ON THE SELECTED LINE.
6649                                    ;*      THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, ON ALL ACTIVE LINES.
6650                                    ;*
6651                                    ;-- ************************************************************************
6652  025154                                    BGNTST
6653  025154                                                                              T9::
6654  025154                                    SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
6655  025154  012700  000240                                                            MOV     #PRI05,R0
6656  025160  104441                                                                    TRAP    C$SPRI
6657           000011                     TNUM == TNUM + 1            ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
6658  025162  012737  000011  002272      MOV     #TNUM,TSTNUM        ;SET UP THE TEST NUMBER.        (40)
6659  025170  012737  177777  002270      MOV     #-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
6660  025176  012737  000001  004052      MOV     #1,ERRTYP           ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
6661  025204  012737  007641  004054      MOV     #4001.,ERRNBR       ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
6662  025212  012737  007715  004056      MOV     #EM4001,ERRMSG      ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
6663  025220  012737  013544  004060      MOV     #ER9101,ERRBLK      ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
6664                                    ;+
6665                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
6666                                    ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
6667                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 4001 <<<<<.
6668                                    ;-
6669  025226  004737  014460                   JSR     PC,CLNRST           ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
6670  025232  103143                           BCC     60$                 ;RESET FAILURE?, ABORT THIS TEST.
6671
6672  025234  004737  015070                   JSR     PC,INDATP           ;INITIALSE THE 256 BYTE DATA PATTERN.
6673                                    ;+
6674                                    ; SET INTERNAL LOOPBACK,ENABLE RECEIVER FUNCTIONS ON ALL ACTIVE LINES.
6675                                    ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
6676                                    ; 2 STOP BITS.
6677                                    ; ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
6678                                    ;-
6679  025240  013705  002240                   MOV     ACTLNS,R5           ;PASS THE ACTIVE LINE BIT MAP.
6680  025244  012700  000204                   MOV     #204,R0             ;PASS THE LNCTRL CONTENTS.
6681  025250  004737  017412                   JSR     PC,WTWLNC           ;INITIALISE THE LNCTRL REGISTERS.
6682  025254  012700  177670                   MOV     #177670,R0          ;PASS THE LPR CONTENTS.
6683  025260  004737  017466                   JSR     PC,WTWLPR           ;INITIALSE THE LPR REGISTERS ON ALL LINES.
6684  025264  004737  016706                   JSR     PC,TXENBL           ;ENABLE TRANSMITTERS ON ALL LINES.
6685                                    ;+
6686                                    ; SET-UP OUTER LOOP TO TEST THE DMA_START BIT ON ALL ACTIVE LINES.
6687                                    ;-
6688  025270  013705  002240                   MOV     ACTLNS,R5           ;GET THE ACTIVE LINE BIT MAP.
6689  025274  005001                           CLR     R1                  ;CLEAR THE LINE NUMBER COUNTER.
6690  025276  012737  007642  004054  2$:      MOV     #4002.,ERRNBR       ;SET THE ERROR NUMBER TO 4002.
6691  025304  000241                           CLC                         ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
6692  025306  006005                           ROR     R5                  ;SHIFT THE BIT MAP INTO THE CARRY BIT.
6693  025310  103106                          BCC     14$                 ;DO NOT TEST THE LINE IF IT IS INACTIVE.
6694  025312  004737  015656                   JSR     PC,PUFIFO           ;PURGE THE FIFO.
6695  025316  103107                          BCC     50$                 ;GO REPORT ERROR IF FIFO WILL NOT CLEAR.
6696                                    ;+
6697                                    ; PERFORM DMA_START BIT TESTING ON EACH LINE INDIVIDUALLY.
6698                                    ; TEST EACH DMA_START BIT BEFORE TX'ING DATA PATTERN, REPORT ERROR IF SET.
```

```
                                        ; SET DMA_START BIT ON LUT, VERIFY IT IS SET, REPORT ERROR IF CLEAR.
                                        ; WAIT FOR DMA TO COMPLETE.
                                        ; VERIFY DMA_START BIT IS CLEAR, REPORT ERROR IF SET.
                                        ; VERIFY CORRECT NUMBER OF CHARS WERE RECEIVED, REPORT ERROR IF < EXPECTED.
                                        ;-
025320  005237  004054                      INC     ERRNBR          ;SET ERROR NUMBER TO 4003.
025324  012702  002712                      MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
025330  012703  000144                      MOV     #100.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
025334  004737  014634                    JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
025340  103067                            BCC     12$             ;GO REPORT ERROR IF DMA_START BIT SET.
                                        ;+
                                        ; TEST THE STATE OF THE DMA_START BIT ON THE LINE UNDER TEST.
                                        ;  REPORT ERROR IF DMA_START BIT IS CLEAR.
                                        ;-
025342  005237  004054                      INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4004.
025346  010177  154674                      MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
025352  105777  154704                      TSTB    @TXAD2A         ;TEST THE STATE OF THE DMA_START BIT.
025356  100060                            BPL     12$             ;GO REPORT ERROR IF BIT IS CLEAR.
                                        ;+
                                        ; WAIT FOR DMA TRANSMISSION TO COMPLETE.
                                        ;-
025360  005237  004054          4$:         INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4005.
025364  010103                              MOV     R1,R3           ;SAVE THE LINE NUMBER.
025366  012701  170144                      MOV     #170144,R1      ;TEST BIT 15, TIMEOUT OF 100 MILLI SECS.
025372  013702  002246                      MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
025376  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
025402  103045                            BCC     10$             ;GO REPORT ERROR IF TIMEOUT OCCURRED.
025404  012704  000005                      MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
025410  004737  014574                    JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
025414  010301                              MOV     R3,R1           ;RESTORE THE CURRENT LINE NUMBER.
                                        ;+
                                        ; TEST THE STATE OF THE DMA_START BIT ON THE LINE UNDER TEST.
                                        ;  REPORT ERROR IF DMA_START BIT IS SET.
                                        ;-
025416  005237  004054                      INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4006.
025422  010177  154620                      MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
025426  105777  154630                      TSTB    @TXAD2A         ;TEST THE STATE OF THE DMA_START BIT.
025432  100432                            BMI     12$             ;GO REPORT ERROR IF BIT IS STILL SET.
                                        ;+
                                        ; VERIFY THE NUMBER OF CHARS RECEIVED = NUMBER OF CHARS EXPECTED.
                                        ; REPORT ERROR IF COUNT IS INCORRECT.
                                        ; IF MORE THAN 128 BMP CODES ARE FOUND THEN REPORT ERROR AND EXIT TEST.
                                        ;-
025434  005003                            CLR     R3              ;CLEAR THE READ COUNTER.
025436  012704  000200                      MOV     #128.,R4        ;SET UP MAX BMP CODE READ COUNT.
025442  012737  007647  004054  6$:        MOV     #4007.,ERRNBR   ;SET ERROR NUMBER TO 4007.
025450  017702  154574                      MOV     @RBUFA,R2       ;READ THE CHARACTER FROM THE FIFO.
025454  100021                            BPL     12$             ;GO REPORT ERROR IF FIFO EMPTY TOO SOON.
025456  012700  170301                      MOV     #170301,R0      ;SET-UP BIT MASK OF A BMP CODE.
025462  040200                            BIC     R2,R0           ;TRY TO CLEAR THE BMP CODE MASK.
025464  001007                            BNE     8$              ;BRANCH IF NOT A BMP CODE.
025466  005237  004054                      INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4008.
025472  004737  016220                    JSR     PC,SAVBMP       ;SAVE THE BMP CODE ON THE QUEUE.
025476  005304                            DEC     R4              ;DECREMENT MAX BMP CODE READ COUNT.
025500  001416                            BEQ     50$             ;GO REPORT ERROR IF TOO MANY BMP CODES FOUND.
025502  000757                            BR      6$              ;DO NOT COUNT THE BMP CODE AS A VALID CHAR.
```

```
6755  025504  005203                  8$:     INC     R3              ;COUNT THIS CHARACTER.
6756  025506  020327  000144                  CMP     R3,#100.        ;HAVE WE RECIEVED 100 CHARACTERS?.
6757  025512  002753                          BLT     6$              ;LOOP UNTIL 100 (NON-BMP) CHARS ARE READ.
6758  025514  000404                          BR      14$             ;SKIP AROUND THE ERROR REPORT.
6759
6760                                          ;+
6761                                          ; REPORT ERROR, SKIP FURTHER TESTING ON THIS LINE.
6762                                          ;-
6763  025516  010301                  10$:    MOV     R3,R1           ;RESTORE THE CURRENT LINE NUMBER.
6764
6765  025520  012702  007740          12$:    MOV     #EM4002,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
6766                                                                  ; 'DMA_START BIT BAD ON LINE NN'.
6767  025524                                  ERROR                   ;                >>>>> ERROR <<<<<.
6768  025524  104460                                                                          TRAP    C$ERROR
6769
6770  025526  005201                  14$:    INC     R1              ;INCREMENT THE LINE NUMBER COUNTER.
6771  025530  005705                          TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
6772  025532  001261                          BNE     2$              ;YES; BRANCH TO TEST THE NEXT LINE.
6773  025534  000402                          BR      60$             ;NO; EXIT THIS TEST.
6774
6775  025536  004737  016456          50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
6776  025542  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6777
6778  025546                                  ENDTST
6779  025546                                                                  L10033:
6780  025546  104401                                                                          TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST           - DMABRT -
                        ;++ ************************************************************************************
                        ;*                          - DMA ABORT/RESTART TEST -
                        ;*      THIS TEST VERIFIES THAT EACH DMA_ABORT BIT WILL CORRECTLY HALT
                        ;*      A DMA TRANSMISSION, AND RETURN A TX_ACTION.
                        ;*      IT WILL ALSO VERIFY THAT THE ABORTED DMA TRANSMISSION CAN BE RESUMMED,
                        ;*      AND THAT A TX_ACTION IS RETURNED UPON COMPLETION.
                        ;*      THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, ON ALL ACTIVE LINES.
                        ;*
                        ;-- ************************************************************************************
025550                          BGNTST
025550                                                                  T10::
025550                          SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
025550  012700  000240                                                  MOV     #PRI05,R0
025554  104441                                                          TRAP    C$SPRI
        000012           TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
025556  012737  000012  002272        MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (41)
025564  012737  177777  002270        MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
025572  012737  000001  004052        MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
025600  012737  010005  004054        MOV     #4101.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
025606  012737  007774  004056        MOV     #EM4101,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
025614  012737  013544  004060        MOV     #ER9101,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                        ;+
                        ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                        ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                        ; THIS SUBROUTINE REPORTS ERROR >>>>> 4101 <<<<<.
                        ;-
025622  004737  014460                JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
025626  103160                      BCC     60$             ;RESET FAILURE?, ABORT THIS TEST.
                        
025630  004737  015070                JSR     PC,INDATP       ;INITIALISE 256 BYTE DATA PATTERN.
                        ;+
                        ; SET INTERNAL LOOPBACK,ENABLE RECEIVER FUNCTIONS ON ALL ACTIVE LINES.
                        ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
                        ; 2 STOP BITS.
                        ; ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
                        ;-
025634  013705  002240                MOV     ACTLNS,R5       ;PASS THE ACTIVE LINE BIT MAP.
025640  012700  000204                MOV     #204,R0         ;PASS THE LNCTRL CONTENTS.
025644  004737  017412                JSR     PC,WTWLNC       ;INITIALISE THE LNCTRL REGISTERS.
025650  012700  177670                MOV     #177670,R0      ;PASS THE LPR CONTENTS.
025654  004737  017466                JSR     PC,WTWLPR       ;INITIALSE THE LPR REGISTERS ON ALL LINES.
025660  004737  016706                JSR     PC,TXENBL       ;ENABLE TRANSMITTERS ON ALL LINES.
                        ;+
                        ; PERFORM DMA_ABORT BIT TESTING ON EACH INDIVIDUAL (ACTIVE) LINE.
                        ;-
025664  013705  002240                MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
025670  005001                        CLR     R1              ;CLEAR THE LINE NUMBER COUNTER.
025672  012737  010006  004054  2$:   MOV     #4102.,ERRNBR   ;SET THE ERROR NUMBER TO 4102.
025700  000241                        CLC                     ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
025702  006005                        ROR     R5              ;SHIFT THE BIT MAP INTO THE CARRY BIT.
025704  103123                      BCC     10$             ;DO NOT TEST THE LINE IF IT IS INACTIVE.
025706  004737  015656                JSR     PC,PUFIFO       ;PURGE THE FIFO.
025712  103124                      BCC     50$             ;GO REPORT ERROR IF FIFO WILL NOT CLEAR.
                        ;+
                        ;  CHECK THE DMA_ABORT BIT BEFORE ENABLING DMA, REPORT ERROR IF SET.
```

```
6837                                            ;-
6838  025714  005237  004054                        INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4103.
6839  025720  010177  154322                        MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
6840  025724  032777  000002  154324                BIT     #BIT1,@LNCTRA   ;TEST THE STATE OF THE DMA_ABORT BIT.
6841  025732  001105                              BNE     6$              ;GO REPORT ERROR IF BIT IS SET.
6842                                            ;+
6843                                            ; ENABLE DMA TX ON SELECTED LINE, WAIT FOR DMA TO TX APPROX 1/4 OF DATA.
6844                                            ; ABORT THE DMA TRANSMISSION. WAIT FOR TX_ACTION TO BE RETURNED.
6845                                            ;-
6846  025734  005237  004054                        INC     ERRNBR          ;SET ERROR NUMBER TO 4104.
6847  025740  012702  002712                        MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
6848  025744  012703  000400                        MOV     #256.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
6849  025750  004737  014634                      JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
6850  025754  103103                              BCC     50$             ;GO REPORT ERROR IF THERE ARE TX PROBLEMS.
6851                                            ;+
6852                                            ; WAIT FOR DMA TO TRANSMIT 1/4 OF THE DATA BEFORE ABORTING.
6853                                            ;-
6854  025756  010177  154264                        MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
6855  025762  012704  000050                        MOV     #40.,R4         ;PASS THE DELAY TIME OF 40 MILLI SECONDS.
6856  025766  004737  014574                      JSR     PC,DELAY        ;WAIT FOR APPROX 1/4 OF DATA TO BE TX'D.
6857  025772  052777  000001  154256              BIS     #BIT0,@LNCTRA   ;ABORT THE DMA TRANSMISSION.
6858                                            ;+
6859                                            ; WAIT FOR TX_ACTION TO BE RETURNED, REPORT ERROR IF TIME-OUT OCCURS.
6860                                            ;-
6861  026000  005237  004054                        INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4105.
6862  026004  010103                              MOV     R1,R3           ;SAVE THE LINE NUMBER.
6863  026006  012701  170012                      MOV     #170012,R1      ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
6864  026012  013702  002246                      MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
6865  026016  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
6866  026022  103050                            BCC     4$              ;GO REPORT ERROR IF TIMEOUT OCCURRED.
6867  026024  010301                              MOV     R3,R1           ;RESTORE THE CURRENT LINE NUMBER.
6868                                            ;+
6869                                            ; VERIFY DMA_START BIT CLEAR, REPORT ERROR IF SET.
6870                                            ;-
6871  026026  005237  004054                        INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4106.
6872  026032  012702  010053                        MOV     #EM4103,R2      ;SELECT MESSAGE TO BE REPORTED.
6873                                                                          ; "DMA_START BIT FOUND SET AFTER DMA ABORTED".
6874  026036  010177  154204                        MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
6875  026042  105777  154214                        TSTB    @TXAD2A         ;TEST THE STATE OF THE DMA_START BIT.
6876  026046  100441                              BMI     8$              ;GO REPORT ERROR IF IT IS SET.
6877                                            ;+
6878                                            ; RESUME DMA TRANSMISSION BY CLEARING DMA_ABORT AND SETTING DMA_START.
6879                                            ;-
6880  026050  042777  000002  154200                BIC     #BIT1,@LNCTRA   ;CLEAR THE DMA_ABORT BIT.
6881  026056  052777  000200  154176                BIS     #BIT7,@TXAD2A   ;SET THE DMA_START BIT.
6882                                            ;+
6883                                            ; WAIT FOR DMA TRANSMISSION TO COMPLETE.
6884                                            ;-
6885  026064  005237  004054                        INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4107.
6886  026070  010103                              MOV     R1,R3           ;SAVE THE LINE NUMBER.
6887  026072  012701  170226                      MOV     #170226,R1      ;TEST BIT 15, TIMEOUT OF 150 MILLI SECS.
6888  026076  013702  002246                      MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
6889  026102  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
6890  026106  103016                            BCC     4$              ;GO REPORT ERROR IF TIMEOUT OCCURRED.
6891  026110  012704  000002                      MOV     #2,R4           ;PASS TIME-OUT OF 2 MILLI SECS.
6892  026114  004737  014574                    JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
```

```
                        MOV     R3,R1           ;RESTORE THE CURRENT LINE NUMBER.
                    ;+
                    ; TEST THE STATE OF THE DMA_ABORT BIT ON THE LINE UNDER TEST.
                    ;  REPORT ERROR IF DMA_ABORT BIT IS SET.
                    ;-
                        INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4108.
                        MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
                        BIT     #BIT1,@LNCTRA   ;TEST THE STATE OF THE DMA_ABORT BIT.
                      BNE       6$              ;GO REPORT ERROR IF BIT IS SET.
                      BR        10$             ;BRANCH TO CHECK FOR ANY MORE LINES TO TEST.
                    ;+
                    ; REPORT ERROR, SKIP FURTHER TESTING ON THIS LINE.
                    ;-
                4$:   MOV     R3,R1           ;RESTORE THE CURRENT LINE NUMBER.

                6$:   MOV     #EM4102,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
                                              ; "DMA_ABORT BIT BAD ON LINE NN".
                8$:  ERROR                    ;                >>>>> ERROR <<<<<.
                                                                        TRAP    C$ERROR
                    ;+
                    ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
                    ;-
               10$:   INC     R1              ;INCREMENT THE LINE NUMBER COUNTER.
                      TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
                    BNE       2$              ;YES; BRANCH TO TEST THE NEXT LINE.
                    BR        60$             ;NO; EXIT THIS TEST.

               50$: JSR       PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
               60$:  CLR      CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.

                    ENDTST
                                                                L10034:
                                                                        TRAP    C$ETST
```

```
                                             .SBTTL  HARDWARE TEST          - OAUTOI -
                                             ;+**********************************************************************
                                             ;*                       - OAUTO BIT INACTIVE TEST -
                                             ;*
                                             ;*        THIS TEST VERIFIES THAT THE DUT'S OAUTO FUNCTION BEHAVES CORRECTLY
                                             ;*        WHEN INACTIVE, IE OAUTO BIT CLEAR.
                                             ;*        THIS TEST WILL ONLY EXECUTE IF STAGGERED LOOPBACK MODE IS SELECTED.
                                             ;*        THE SPECIAL STAGGERED LOOPBACK CONNECTOR MUST BE FITTED.
                                             ;*
                                             ;-**********************************************************************

026176                                               BGNTST
026176                                                                                        T11::
026176  123727  002242  000002                        CMPB    LOPBCK,#2           ;CHECK MODE SELECTED.
026204  001402                                       BEQ     .+6                 ;DO NOT EXIT IF STAGGERD LOPBCK MODE SELECTED.
026206  000137  026736                               JMP     60$                 ;EXIT THIS TEST.
026212                                               SETPRI  #PRI05              ;ALLOW LTC INTERRUPTS.
026212  012700  000240                                                                                MOV     #PRI05,R0
026216  104441                                                                                        TRAP    C$SPRI
        000013                                        TNUM == TNUM + 1           ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
026220  012737  000013  002272                        MOV     #TNUM,TSTNUM       ;SET UP THE TEST NUMBER.     (49)
026226  012737  177777  002270                        MOV     #-1,CTRLCF         ;INDICATE THAT WE ARE IN A TEST.
026234  012737  000001  004052                        MOV     #1,ERRTYP          ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
026242  012737  011445  004054                        MOV     #4901.,ERRNBR      ;SET ERROR NUMBER TO 4901.
026250  012737  010137  004056                        MOV     #EM4901,ERRMSG     ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
026256  012737  013544  004060                        MOV     #ER9101,ERRBLK     ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                             ;+
                                             ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                             ;   CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                             ; THIS SUBROUTINE REPORTS ERROR >>>>> 4901 <<<<<.
                                             ;-
026264  004737  014460                               JSR     PC,CLNRST           ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
026270  103402                                       BCS     .+6                 ;DO NOT EXIT IF RESET WAS SUCCESSFUL.
026272  000137  026736                               JMP     60$                 ;EXIT THIS TEST.
                                             ;+
                                             ; SET-UP THE ASSOCIATED TX/RX LINE NUMBER TABLES.
                                             ;-
026276  004737  014024                               JSR     PC,ASLNTL           ;INITIALISE THE ASSOCIATED TX/RX TABLES.
                                             ;+
                                             ; SET EXTERNAL LOOPBACK, DISABLE OAUTO AND ENABLE RECEIVER ON ALL ACTIVE LINES.
                                             ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
                                             ; 2 STOP BITS.
                                             ; ENABLE TRANSMITTERS ON ALL LINES.
                                             ;-
026302  013705  002240                                MOV     ACTLNS,R5          ;PASS THE ACTIVE LINE BIT MAP.
026306  012700  000004                                MOV     #4,R0              ;PASS THE LNCTRL CONTENTS.
026312  004737  017412                               JSR     PC,WTWLNC           ;INITIALISE THE LNCTRL REGISTERS.
026316  012705  000377                                MOV     #MAPLNS,R5         ;PASS BIT MAP OF ALL LINES.
026322  012700  177670                                MOV     #177670,R0         ;PASS THE LPR CONTENTS.
026326  004737  017466                               JSR     PC,WTWLPR           ;INITIALSE THE LPR REGISTERS ON ALL LINES.
026332  004737  016706                               JSR     PC,TXENBL           ;ENABLE TRANSMITTERS ON ALL LINES.
                                             ;+
                                             ; SET UP OUTER LOOP FOR TESTING ACTIVE LINES IN BOTH LINE GROUPS.
                                             ;-
026336  012703  100000                                MOV     #100000,R3         ;SET-UP LOOP CONTROL FLAG.
026342  013705  002240                                MOV     ACTLNS,R5          ;GET THE ACTIVE LINE BIT MAP.
```

```
 6982  026346  043705  002300                BIC     LGRP2M,R5       ;REMOVE LINES IN GROUP 2.
 6983  026352  010537  026730          2$:   MOV     R5,45$          ;SAVE THE CURRENT LINE GROUP.
 6984  026356  005037  026726                CLR     40$             ;CLEAR THE LINE NUMBER COUNTER.
 6985  026362  013701  026726          4$:   MOV     40$,R1          ;COPY THE LINE NUMBER.
 6986  026366  000241                        CLC                     ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
 6987  026370  006005                        ROR     R5              ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
 6988  026372  103054                      BCC     8$              ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
 6989                                    ;+
 6990                                    ; TEST THE STATE OF THE OAUTO BIT ON THE LINE UNDER TEST.
 6991                                    ; REPORT ERROR IF IT IS FOUND SET, AND SKIP FURTHER TESTING OF THAT LINE.
 6992                                    ;-
 6993  026374  012737  011446  004054            MOV     #4902.,ERRNBR   ;SET THE ERROR NUMBER TO 4902.
 6994  026402  010177  153640                    MOV     R1,@CSRA        ;SELECT THE LINE TO BE TESTED.
 6995  026406  032777  000020  153642            BIT     #BIT4,@LNCTRA   ;TEST THE STATE OF THE OAUTO BIT.
 6996  026414  001404                          BEQ     6$              ;SKIP ERROR REPORT IF OAUTO BIT IS CLEAR.
 6997  026416  012702  010171                    MOV     #EM4902,R2      ;PASS THE ERROR MESSAGE.
 6998                                                                    ; "OAUTO BIT BAD ON LINE NN"
 6999  026422                                    ERROR                   ;                 >>>>> ERROR #4902 <<<<<.
 7000  026422  104460                                                                              TRAP    C$ERROR
 7001  026424  000437                            BR      8$              ;SKIP FURTHER TESTING OF THIS LINE.
 7002                                    ;+
 7003                                    ; TRANSMIT THE XOFF (ASCII DC3) ON THE ASSOCIATED LINE.
 7004                                    ;-
 7005  026426  116177  004012  153612  6$:       MOVB    TXRLNB(R1),@CSRA ;SELECT THE ASSOCIATED TX LINE.
 7006  026434  012777  100023  153606            MOV     #100023,@RBUFA  ;TRANSMIT THE XOFF CHARACTER TO THE LUT.
 7007                                    ;+
 7008                                    ; WAIT FOR TRANSMISSION TO COMPLETE.
 7009                                    ;-
 7010  026442  005237  004054                    INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4903.
 7011  026446  012701  170012                    MOV     #170012,R1      ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
 7012  026452  013702  002246                    MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
 7013  026456  004737  017276                  JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
 7014  026462  103123                          BCC     50$             ;ABORT TEST IF TIMEOUT OCCURRED.
 7015  026464  012704  000005                    MOV     #5,R4           ;PASS TIME-OUT OF 5 MILLI SECS.
 7016  026470  004737  014574                  JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
 7017                                    ;+
 7018                                    ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
 7019                                    ; REPORT ERROR IF TX_ENABLE BIT IS CLEAR.
 7020                                    ;-
 7021  026474  005237  004054                    INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4904.
 7022  026500  013701  026726                    MOV     40$,R1          ;GET THE NUMBER OF THE LINE TEST.
 7023  026504  010177  153536                    MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
 7024  026510  005777  153546                    TST     @TXAD2A         ;TEST THE STATE OF THE TX_ENABLE BIT.
 7025  026514  100403                          BMI     8$              ;SKIP ERROR REPORT IF BIT IS SET.
 7026  026516  012702  010171                    MOV     #EM4902,R2      ;PASS THE MESSAGE TO BE REPORTED.
 7027                                                                    ; 'OAUTO BIT BAD ON LINE NN'.
 7028  026522                                    ERROR                   ;                 >>>>> ERROR #4904 <<<<<.
 7029  026522  104460                                                                              TRAP    C$ERROR
 7030
 7031  026524  005237  026726          8$:       INC     40$             ;INCREMENT THE LINE NUMBER,
 7032  026530  005705                            TST     R5              ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
 7033  026532  001313                          BNE     4$              ;
 7034                                    ;+
 7035                                    ; DISABLE TRANSMITTERS ON THE SELECTED LINES IN THE CURRENT LINE GROUP.
 7036                                    ;-
 7037  026534  013705  026730                    MOV     45$,R5          ;RESTORE THE CURRENT LINE ACTIVE LINE GROUP.
```

```
 7038  026540  004737  016612                    JSR     PC,TXDSBL       ;DISABLE TRANSMITTERS ON THE SELECTED LINES.
 7039  026544  013705  026730                     MOV     45$,R5          ;GET THE CURRENT ACTIVE LINE GROUP AGAIN.
 7040  026550  005037  026726                     CLR     40$             ;CLEAR THE LINE COUNTER.
 7041  026554  012737  011451  004054  10$:       MOV     #4905.,ERRNBR   ;SET ERROR NUMBER TO 4905.
 7042  026562  013701  026726                     MOV     40$,R1          ;COPY THE LINE NUMBER.
 7043  026566  000241                             CLC                     ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
 7044  026570  006005                             ROR     R5              ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
 7045  026572  103035                            BCC     12$             ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
 7046                                          ;+
 7047                                          ; TRANSMIT THE XON (ASCII DC1) ON THE ASSOCIATED LINE.
 7048                                          ;-
 7049  026574  116177  004012  153444             MOVB    TXRLNB(R1),@CSRA ;SELECT THE ASSOCIATED TX LINE.
 7050  026602  012777  100021  153440             MOV     #100021,@RBUFA  ;TRANSMIT THE XON CHARACTER TO THE LUT.
 7051                                          ;+
 7052                                          ; WAIT FOR TRANSMISSION TO COMPLETE.
 7053                                          ;-
 7054  026610  012701  170012                     MOV     #170012,R1      ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
 7055  026614  013702  002246                     MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
 7056  026620  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
 7057  026624  103042                            BCC     50$             ;ABORT TEST IF TIMEOUT OCCURRED.
 7058  026626  012704  000005                     MOV     #5,R4           ;PASS TIME-OUT OF 5 MILLI SECS.
 7059  026632  004737  014574                    JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
 7060                                          ;+
 7061                                          ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
 7062                                          ;  REPORT ERROR IF TX_ENABLE BIT IS SET.
 7063                                          ;-
 7064  026636  005237  004054                     INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 4906.
 7065  026642  013701  026726                     MOV     40$,R1          ;GET THE NUMBER OF THE LINE UNDER TEST.
 7066  026646  010177  153374                     MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
 7067  026652  005777  153404                     TST     @TXAD2A         ;TEST THE STATE OF THE TX_ENABLE BIT.
 7068  026656  100003                            BPL     12$             ;SKIP ERROR REPORT IF BIT IS CLEAR.
 7069  026660  012702  010171                     MOV     #EM4902,R2      ;PASS THE MESSAGE TO BE REPORTED.
 7070                                                                    ; 'OAUTO BIT BAD ON LINE NN'.
 7071  026664                                     ERROR                   ;                >>>>> ERROR #4906 <<<<<.
 7072  026664  104460                                                                            TRAP    C$ERROR
 7073
 7074  026666  005237  026726          12$:       INC     40$             ;INCREMENT THE LINE NUMBER,
 7075  026672  005705                             TST     R5              ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
 7076  026674  001327                            BNE     10$             ;
 7077                                          ;+
 7078                                          ; CHECK LOOP CONTROL FLAG TO DETERMINE IF BOTH SETS OF LINES HAVE BEEN TESTED
 7079                                          ;  IF THIS IS THE FIST TIME AROUND, RE-ENABLE TX ON ALL LINES, GENERATE ACTIVE
 7080                                          ;  BIT MAP FOR SECOND LINE GROUP.
 7081                                          ;-
 7082  026676  005703                             TST     R3              ;HAVE BOTH LINE GROUPS BEEN TESTED?.
 7083  026700  001416                            BEQ     60$             ;YES; THEN EXIT THIS TEST.
 7084  026702  005003                             CLR     R3              ;NO; CLEAR THE LOOP CONTROL FLAG,
 7085  026704  012705  000377                     MOV     #MAPLNS,R5      ;PASS THE BIT MAP OF ALL AVAILABLE LINE.
 7086  026710  004737  016706                    JSR     PC,TXENB'       ;RE-ENABLE TRANSMISSION ON ALL LINFS.
 7087  026714  013705  002240                     MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
 7088  026720  043705  002276                     BIC     LGRP1M,R5       ;REMOVE ALL ACTIVE LINES IN GROUP 1.
 7089  026724  000612                            BR      2$              ;ONCE MORE AROUND AND WE ARE DONE.
 7090
 7091  026726  000000          40$:       .WORD   0               ;STORAGE FOR CURRENT LINE NUMBER.
 7092  026730  000000          45$:       .WORD   0               ;STORAGE FOR CURRENT ACTIVE LINE BIT MAP.
 7093  026732  004737  016456  50$:       JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
```

```
 7094  026736  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
 7095
 7096  026742                                  ENDTST
 7097  026742                                                          L10035:
 7098  026742  104401                                                          TRAP    C$ETST
```

```
7099
7100                                    .SBTTL  HARDWARE TEST           - OAUTOA -
7101                                    ;++*************************************************************************
7102                                    ;*                      - OAUTO BIT ACTIVE TEST -
7103                                    ;*
7104                                    ;*      THIS TEST VERIFIES THAT THE DUT'S OAUTO FUNCTION BEHAVES CORRECTLY
7105                                    ;*      WHEN ACTIVE, IE OAUTO BIT ASSERTED HIGH.
7106                                    ;*      THIS TEST WILL ONLY EXECUTE IF THE STAGGERED LOOPBACK MODE IS SELECTED.
7107                                    ;*      THE SPECIAL STAGGERED LOOPBACK CONNECTOR MUST BE FITTED.
7108                                    ;*
7109                                    ;--*************************************************************************
7110
7111  026744                                    BGNTST
7112  026744                                                                                    T12::
7113  026744  123727  002242  000002            CMPB    LOPBCK,#2       ;CHECK MODE SELECTED.
7114  026752  001402                    BEQ     .+6             ;DO NOT EXIT IF STAGGERD LOPBCK MODE SELECTED.
7115  026754  000137  027504            JMP     60$             ;EXIT THIS TEST.
7116  026760                            SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
7117  026760  012700  000240                                                            MOV     #PRI05,R0
7118  026764  104441                                                                    TRAP    C$SPRI
7119           000014                    TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
7120  026766  012737  000014  002272    MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (50)
7121  026774  012737  177777  002270    MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
7122  027002  012737  000001  004052    MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
7123  027010  012737  011611  004054    MOV     #5001.,ERRNBR   ;SET ERROR NUMBER TO 5001.
7124  027016  012737  010223  004056    MOV     #EM5001,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
7125  027024  012737  013544  004060    MOV     #ER9101,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
7126                                    ;+
7127                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
7128                                    ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
7129                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 5001 <<<<<.
7130                                    ;-
7131  027032  004737  014460                    JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
7132  027036  103402                    BCS     .+6             ;DO NOT EXIT IF RESET WAS SUCCESSFUL.
7133  027040  000137  027504            JMP     60$             ;EXIT THIS TEST.
7134                                    ;+
7135                                    ; SET-UP THE ASSOCIATED TX/RX LINE NUMBER TABLES.
7136                                    ;-
7137  027044  004737  014024                    JSR     PC,ASLNTL       ;INITIALISE THE ASSOCIATED TX/RX TABLES.
7138                                    ;+
7139                                    ; SET EXTERNAL LOOPBACK,ENABLE OAUTO AND RECEIVER FUNCTIONS ON ALL ACTIVE LINES
7140                                    ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
7141                                    ; 2 STOP BITS.
7142                                    ; ENABLE TRANSMITTERS ON ALL LINES.
7143                                    ;-
7144  027050  013705  002240                    MOV     ACTLNS,R5       ;PASS THE ACTIVE LINE BIT MAP.
7145  027054  012700  000024                    MOV     #24,R0          ;PASS THE LNCTRL CONTENTS.
7146  027060  004737  017412                   JSR     PC,WTWLNC       ;INITIALISE THE LNCTRL REGISTERS.
7147  027064  012705  000377                    MOV     #MAPLNS,R5      ;PASS BIT MAP OF ALL LINES.
7148  027070  012700  177670                    MOV     #177670,R0      ;PASS THE LPR CONTENTS.
7149  027074  004737  017466                   JSR     PC,WTWLPR       ;INITIALSE THE LPR REGISTERS ON ALL LINES.
7150  027100  004737  016706                   JSR     PC,TXENBL       ;ENABLE TRANSMITTERS ON ALL LINES.
7151                                    ;+
7152                                    ; SET UP OUTER LOOP FOR TESTING ACTIVE LINES IN BOTH LINE GROUPS.
7153                                    ;-
7154  027104  012703  100000                    MOV     #100000,R3      ;SET-UP LOOP CONTROL FLAG.
```

```
7155  027110  013705  002240          MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
7156  027114  043705  002300          BIC     LGRP2M,R5       ;REMOVE LINES IN GROUP 2.
7157  027120  010537  027476  2$:     MOV     R5,45$          ;SAVE THE CURRENT LINE GROUP.
7158  027124  005037  027474          CLR     40$             ;CLEAR THE LINE NUMBER COUNTER.
7159  027130  013701  027474  4$:     MOV     40$,R1          ;COPY THE LINE NUMBER.
7160  027134  000241                  CLC                     ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
7161  027136  006005                  ROR     R5              ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
7162  027140  103054                  BCC     8$              ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
7163                                ;+
7164                                ; TEST THE STATE OF THE OAUTO BIT ON THE LINE UNDER TEST.
7165                                ;  REPORT ERROR IF IT IS FOUND CLEAR, AND SKIP FURTHER TESTING OF THAT LINE.
7166                                ;-
7167  027142  012737  011612  004054      MOV     #5002.,ERRNBR   ;SET THE ERROR NUMBER TO 5002.
7168  027150  010177  153072              MOV     R1,@CSRA        ;SELECT THE LINE TO BE TESTED.
7169  027154  032777  000020  153074      BIT     #BIT4,@LNCTRA   ;TEST THE STATE OF THE OAUTO BIT.
7170  027162  001004                      BNE     6$              ;SKIP ERROR REPORT IF OAUTO BIT IS SET.
7171  027164  012702  010171              MOV     #EM4902,R2      ;PASS THE ERROR MESSAGE.
7172                                                              ; "OAUTO BIT BAD ON LINE NN"
7173  027170                              ERROR                   ;                >>>>> ERROR #5002 <<<<<.
7174  027170  104460                                                                          TRAP    C$ERROR
7175  027172  000437                      BR      8$              ;SKIP FURTHER TESTING OF THIS LINE.
7176                                    ;+
7177                                    ; TRANSMIT THE XOFF (ASCII DC3) ON THE ASSOCIATED LINE.
7178                                    ;-
7179  027174  116177  004012  153044  6$:     MOVB    TXRLNB(R1),@CSRA ;SELECT THE ASSOCIATED TX LINE.
7180  027202  012777  100023  153040          MOV     #100023,@RBUFA  ;TRANSMIT THE XOFF CHARACTER TO THE LUT.
7181                                    ;+
7182                                    ; WAIT FOR TRANSMISSION TO COMPLETE.
7183                                    ;-
7184  027210  005237  004054              INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 5003.
7185  027214  012701  170012              MOV     #170012,R1      ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
7186  027220  013702  002246              MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7187  027224  004737  017276              JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
7188  027230  103123                      BCC     50$             ;ABORT TEST IF TIMEOUT OCCURRED.
7189  027232  012704  000005              MOV     #5,R4           ;PASS TIME-OUT OF 5 MILLI SECS.
7190  027236  004737  014574              JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
7191                                    ;+
7192                                    ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
7193                                    ;  REPORT ERROR IF TX_ENABLE BIT IS SET.
7194                                    ;-
7195  027242  005237  004054              INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 5004.
7196  027246  013701  027474              MOV     40$,R1          ;GET THE NUMBER OF THE LINE TEST.
7197  027252  010177  152770              MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
7198  027256  005777  153000              TST     @TXAD2A         ;TEST THE STATE OF THE TX_ENABLE BIT.
7199  027262  100003                      BPL     8$              ;SKIP ERROR REPORT IF BIT IS CLEAR.
7200  027264  012702  010171              MOV     #EM4902,R2      ;PASS THE MESSAGE TO BE REPORTED.
7201                                                              ; "OAUTO BIT BAD ON LINE NN".
7202  027270                              ERROR                   ;                >>>>> ERROR #5004 <<<<<.
7203  027270  104460                                                                          TRAP    C$ERROR
7204
7205  027272  005237  027474  8$:     INC     40$             ;INCREMENT THE LINE NUMBER,
7206  027276  005705                  TST     R5              ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
7207  027300  001313                  BNE     4$              ;
7208                                ;+
7209                                ; DISABLE TRANSMITTERS ON THE SELECTED LINES IN THE CURRENT LINE GROUP.
7210                                ;-
```

```
7211  027302  013705  027476                    MOV     45$,R5          ;RESTORE THE CURRENT LINE ACTIVE LINE GROUP.
7212  027306  004737  016612                    JSR     PC,TXDSBL       ;DISABLE TRANSMITTERS ON THE SELECTED LINES.
7213  027312  013705  027476                    MOV     45$,R5          ;GET THE CURRENT LINE ACTIVE LINE GROUP AGAIN.
7214  027316  005037  027474                    CLR     40$             ;CLEAR THE LINE COUNTER.
7215  027322  012737  011615  004054  10$:      MOV     #5005.,ERRNBR   ;SET ERROR NUMBER TO 5005.
7216  027330  013701  027474                    MOV     40$,R1          ;COPY THE LINE NUMBER.
7217  027334  000241                            CLC                     ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
7218  027336  006005                            ROR     R5              ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
7219  027340  103035                           BCC     12$             ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
7220                                          ;+
7221                                          ; TRANSMIT THE XON (ASCII DC1) ON THE ASSOCIATED LINE.
7222                                          ;-
7223  027342  116177  004012  152676            MOVB    TXRLNB(R1),@CSRA ;SELECT THE ASSOCIATED TX LINE.
7224  027350  012777  100021  152672            MOV     #100021,@RBUFA  ;TRANSMIT THE XON CHARACTER TO THE LUT.
7225                                          ;+
7226                                          ; WAIT FOR TRANSMISSION TO COMPLETE.
7227                                          ;-
7228  027356  012701  170012                    MOV     #170012,R1      ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
7229  027362  013702  002246                    MOV     CSRA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7230  027366  004737  017276                   JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE.
7231  027372  103042                           BCC     50$             ;ABORT TEST IF TIMEOUT OCCURRED.
7232  027374  012704  000005                    MOV     #5,R4           ;PASS TIME-OUT OF 5 MILLI SECS.
7233  027400  004737  014574                   JSR     PC,DELAY        ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
7234                                          ;+
7235                                          ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
7236                                          ;  REPORT ERROR IF TX_ENABLE BIT IS CLEAR.
7237                                          ;-
7238  027404  005237  004054                    INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 5006.
7239  027410  013701  027474                    MOV     40$,R1          ;GET THE NUMBER OF THE LINE UNDER TEST.
7240  027414  010177  152626                    MOV     R1,@CSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
7241  027420  005777  152636                   TST     @TXAD2A         ;TEST THE STATE OF THE TX_ENABLE BIT.
7242  027424  100403                           BMI     12$             ;SKIP ERROR REPORT IF BIT IS CLEAR.
7243  027426  012702  010171                    MOV     #EM4902,R2      ;PASS THE MESSAGE TO BE REPORTED.
7244                                                                  ; ''0AUTO BIT BAD ON LINE NN''.
7245  027432                                  ERROR                   ;                >>>>> ERROR #5006 <<<<<.
7246  027432  104460                                                                          TRAP    C$ERROR
7247
7248  027434  005237  027474          12$:      INC     40$             ;INCREMENT THE LINE NUMBER,
7249  027440  005705                            TST     R5              ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
7250  027442  001327                           BNE     10$             ;
7251                                          ;+
7252                                          ; CHECK LOOP CONTROL FLAG TO DETERMINE IF BOTH SETS OF LINES HAVE BEEN TESTED
7253                                          ;  IF THIS IS THE FIST TIME AROUND, RE-ENABLE TX ON ALL LINES, GENERATE ACTIVE
7254                                          ;  BIT MAP FOR SECOND LINE GROUP.
7255                                          ;-
7256  027444  005703                            TST     R3              ;HAVE BOTH LINE GROUPS BEEN TESTED?.
7257  027446  001416                           BEQ     60$             ;YES; THEN EXIT THIS TEST.
7258  027450  005003                            CLR     R3              ;NO; CLEAR THE LOOP CONTROL FLAG,
7259  027452  012705  000377                    MOV     #MAPLNS,R5      ;PASS THE BIT MAP OF ALL AVAILABLE LINE.
7260  027456  004737  016706                   JSR     PC,TXENBL       ;RE-ENABLE TRANSMISSION ON ALL LINES.
7261  027462  013705  002240                    MOV     ACTLNS,R5       ;GET THE ACTIVE LINE BIT MAP.
7262  027466  043705  002276                    BIC     LGRP1M,R5       ;REMOVE ALL ACTIVE LINES IN GROUP 1.
7263  027472  000612                           BR      2$              ;ONCE MORE AROUND AND WE ARE DONE.
7264
7265  027474  000000                  40$:      .WORD   0               ;STORAGE FOR CURRENT LINE NUMBER.
7266  027476  000000                  45$:      .WORD   0               ;STORAGE FOR CURRENT ACTIVE LINE BIT MAP.
```

```
 7267  027500  004737  016456          50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
 7268  027504  005037  002270          60$:     CLR    CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
 7269
 7270  027510                                  ENDTST
 7271  027510                                                                  L10036:
 7272  027510  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST          - IAUTOI -
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*                      - IAUTO BIT INACTIVE TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE DUT'S IAUTO FUNCTION BEHAVES CORRECTLY
                                ;*      WHEN INACTIVE, IE. IAUTO BIT CLEAR.
                                ;*      ALL ACTIVE LINES ARE TESTED INDIVIDUALLY BY FILLING THE FIFO
                                ;*      THEN READING THE RECEIVED DATA CHECKING FOR THE PRESENCE OF
                                ;*      XOFF(ASCII DC3) OR XON (ASCII DC1) CHARACTERS.
                                ;*      IF ANY ARE FOUND THEN APPROPRIATE ERRORS ARE REPORTED.
                                ;*      ANY BMP CODES THAT ARE FOUND WILL BE PLACED ON THE BMP CODE QUEUE,
                                ;*      TO BE REPORTED LATER.
                                ;*      THE CHARACTERS ARE TRANSMITTED ON ALL ACTIVE LINES, IN INTERNAL
                                ;*      LOOPBACK MODE.
                                ;*
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

027512                                  BGNTST
027512                                                                  T13::
027512                                  SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
027512  012700  000240                                                  MOV     #PRI05,R0
027516  104441                                                          TRAP    C$SPRI
        000015                           TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
027520  012737  000015  002272           MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.        (51)
027526  012737  177777  002270           MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
027534  012737  000001  004052           MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
027542  012737  011755  004054           MOV     #5101.,ERRNBR  ;SET ERROR NUMBER TO 5101.
027550  012737  010253  004056           MOV     #EM5101,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
027556  012737  013544  004060           MOV     #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 5101 <<<<<.
                                ;-
027564  004737  014460                   JSR     PC,CLNRST      ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
027570  103146                           BCC     60$            ;EXIT TEST IF FATAL ERROR FOUND.

                                ;+
                                ; INITIALIZE THE 256 BYTE DATA PATTERN.
                                ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                                ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                                ;-
027572  004737  015120                   JSR     PC,INDTPX      ;INITIALISE DATA PATTERN.
                                ;+
                                ; SET INTERNAL LOOPBACK, DISABLE IAUTO, ENABLE RECEIVER ON THE SELECTED LINE.
                                ; SET LPR TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                                ;-
027576  013705  002240                    MOV     ACTLNS,R5      ;PASS THE ACTIVE LINE BIT MAP.
027602  012700  000204                    MOV     #204,R0        ;PASS INT'L LOPBCK, ENABLE RX, DISABLE IAUTO.
027606  004737  017412                   JSR     PC,WTWLNC      ;INITIALISE THE LINE CONTROL REGISTER.
027612  012700  177670                    MOV     #177670,R0     ;PASS THE LPR CONTENTS.
027616  004737  017466                   JSR     PC,WTWLPR      ;SET THE LPR CONTENTS TO 38.4K BAUD.
027622  013704  000012                    MOV     10.,R4         ;PASS DELAY TIME OF 10 MILLI SECONDS.
027626  004737  014574                   JSR     PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
```

```
                                        ;+
                                        ; SET UP LOOP FOR ALL ACTIVE LINES.
                                        ; TEST THE STATE OF THE IAUTO BIT PRIOR TO TRANSMITTING THE DATA PATTERN.
                                        ;  IF THE BIT IS SET, THEN REPORT THE ERROR AND SKIP TRANSMITTING
                                        ;  THE DATA PATTERN ON THE SELECTED LINE.
                                        ; TRANSMIT A 256 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                                        ; EMPTY THE FIFO, AND VERIFY NO XOFF OR XON CHARS WERE FOUND.
                                        ;-
027632  005001                            CLR     R1              ;CLEAR THE LINE NUMBER COUNTER.
027634  005037  030104                    CLR     55$             ;CLEAR STORAGE FOR LINE NUMBER.
027640  012737  011756  004054  2$:       MOV     #5102.,ERRNBR   ;SET THE ERROR NUMBER TO 5102.
027646  004737  015656                    JSR     PC,PUFIFO       ;PURGE THE FIFO.
027652  103111                            BCC     50$             ;GO REPORT ERROR IF FIFO DID NOT PURGE.
027654  000241                            CLC                     ;CLEAR CARRY PRIOR TO ROTATING BIT MAP.
027656  006005                            ROR     R5              ;ROTATE THE BIT MAP INTO THE CARRY BIT.
027660  103077                            BCC     12$             ;BRANCH IF LINE IS INACTIVE.
                                        ;+
                                        ; TEST THE IAUTO BIT ON THE SELECTED ACTIVE LINE.
                                        ; REPORT ERROR IF IT IS SET.
                                        ;  DO NOT TRANSMIT THE DATA PATTERN ON THE SELECTED LINE.
                                        ;-
027662  005237  004054                    INC     ERRNBR          ;SET ERROR NUMBER TO 5103.
027666  010177  152354                    MOV     R1,@CSRA        ;SELECT LINE TO TEST.
027672  032777  000002  152356            BIT     #BIT1,@LNCTRA   ;TEST THE STATE OF THE IAUTO BIT ON THIS LINE.
027700  001404                            BEQ     4$              ;SKIP ERROR IF IAUTO BIT CLEAR.
027702  012702  010301                    MOV     #EM5102,R2      ;PASS THE CORRECT ERROR MESSAGE.
027706                                    ERROR                   ;                >>>>> ERROR <<<<<.
027706  104460                                                                            TRAP    C$ERROR
027710  000463                            BR      12$             ;SKIP TRANSMITTING DATA PATTERN.

                                        ;+
                                        ; TRANSMIT DATA PATTERN OF 256 CHARS.
                                        ;-
027712  005237  004054  4$:               INC     ERRNBR          ;SET ERROR NUMBER TO 5104.
027716  012702  002712                    MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
027722  012703  000400                    MOV     #256.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
027726  004737  014634                    JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
027732  103061                            BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.

                                        ;+
                                        ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER PLUS XOFF
                                        ; TO ARRIVE IN THE FIFO.
                                        ;-
027734  005237  004054                    INC     ERRNBR          ;SET ERROR NUMBER TO 5105.
027740  012701  170454                    MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
027744  013702  002246                    MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
027750  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
027754  103050                            BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
027756  012704  000012                    MOV     #10.,R4           ASS DELAY OF 10 MILLI SECS.
027762  004737  014574                    JSR     PC,DELAY        ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.

                                        ;+
                                        ; READ 256 CHARS FROM THE FIFO.  REPORT ERROR IF ANY XOFF'S OR XON'S
                                        ; ARE FOUND.
                                        ;-
027766  005237  004054                    INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 5106.
```

```
7385  027772  012701  000400          MOV     #256.,R1        ;INITIALISE THE READ COUNTER.
7386  027776  017702  152246  6$:     MOV     @RBUFA,R2       ;READ CHAR FROM THE FIFO.
7387  030002  100035                  BPL     50$             ;GO REPORT ERROR IF FIFO EMPTY.
7388                                  ;+
7389                                  ; CHECK FOR BMP CODE IN THE FIFO.  SAVE ANY FOUND ON THE QUEUE.
7390                                  ;-
7391  030004  012700  170301          MOV     #170301,R0      ;SET UP BMP BIT MASK.
7392  030010  040200                  BIC     R2,R0           ;TRY TO CLEAR ALL THE BMP BITS.
7393  030012  001002                  BNE     8$              ;SKIP BMPSAV IF NOT A BMP CODE.
7394  030014  004737  016220          JSR     PC,SAVBMP       ;SAVE THE BMP CODE ON THE QUEUE.
7395                                  ;+
7396                                  ; CHECK FOR XOFF AND XON CHARACTERS.
7397                                  ;-
7398  030020  120227  000023  8$:     CMPB    R2,#23          ;IS IT AN XOFF CHARACTER?.
7399  030024  001406                  BEQ     10$             ;YES; GO REPORT ERROR.
7400  030026  120227  000021          CMPB    R2,#21          ;NO; IS IT AN XON CHARACTER?.
7401  030032  001403                  BEQ     10$             ;YES; GO REPORT ERROR.
7402  030034  005301                  DEC     R1              ;DECREMENT THE READ COUNT.
7403  030036  001357                  BNE     6$              ;LOOP TO READ THE NEXT CHAR.
7404  030040  000407                  BR      12$             ;GO CHECK FOR ANY UNTESTED ACTIVE LINES.
7405
7406  030042  005237  004054  10$:    INC     ERRNBR          ;SET ERROR NUMBER TO 5107.
7407  030046  013701  030104          MOV     55$,R1          ;PASS THE LINE NUMBER TO BE REPORTED.
7408  030052  012702  010337          MOV     #EM5103,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
7409  030056                          ERROR                   ;                >>>>> ERROR <<<<<.
7410  030056  104460                                                              TRAP    C$ERROR
7411                                  ;+
7412                                  ; CHECK IF ALL ACTIVE LINES HAVE BEEN TESTED.
7413                                  ;-
7414  030060  005237  030104  12$:    INC     55$             ;INCREMENT LINE NUMBER.
7415  030064  013701  030104          MOV     55$,R1          ;GET NUMBER OF THE NEXT LINE TO TEST.
7416  030070  005705                  TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
7417  030072  001262                  BNE     2$              ;LOOP TO CHECK NEXT LINE.
7418  030074  000404                  BR      60$             ;EXIT TEST.
7419
7420  030076  004737  016456  50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
7421  030102  000401                  BR      [illegible]     ;EXIT THIS TEST.
7422  030104  000000  55$:    .WORD   .                       ;STORAGE FOR LINE NUMBER.
7423  030106  005037  002270  60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
7424
7425  030112                          ENDTST
7426  030112                                                  L10037:
7427  030112  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - IAUTOA -
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*                      - IAUTO BIT ACTIVE TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE DUT'S IAUTO FUNCTION BEHAVES CORRECTLY
                                ;*      WHEN ACTIVE, IE IAUTO ASSERTED HIGH.
                                ;*      ALL ACTIVE LINES ARE TESTED INDIVIDUALLY BY FILLING THE FIFO, AND
                                ;*      CHECKING FOR THE PRESENCE OF AT LEAST ONE XOFF(ASCII DC3) CHARACTER
                                ;*      AND ONE XON (ASCII DC1) CHARACTER.
                                ;*      ANY BMP CODES THAT ARE FOUND WILL BE PLACED ON THE BMP CODE QUEUE,
                                ;*      TO BE REPORTED LATER.
                                ;*      THE CHARACTERS ARE TRANSMITTED ON ALL ACTIVE LINES, IN INTERNAL
                                ;*      LOOPBACK MODE.
                                ;*
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                        BGNTST
                                                                                T14::
030114                                  SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
030114                                                                          MOV     #PRI05,R0
030114  012700  000240                                                          TRAP    C$SPRI
030120  104441
                000016                   TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
030122  012737  000016  002272           MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.        (52)
030130  012737  177777  002270           MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
030136  012737  000001  004052           MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
030144  012737  012121  004054           MOV     #5201.,ERRNBR  ;SET ERROR NUMBER TO 5201.
030152  012737  010367  004056           MOV     #EM5201,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
030160  012737  013544  004060           MOV     #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 5201 <<<<<.
                                ;-
030166  004737  014460                   JSR     PC,CLNRST      ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
030172  103156                         BCC     60$            ;EXIT TEST IF FATAL ERROR FOUND.

                                ;+
                                ; INITIALIZE THE 256 BYTE DATA PATTERN.
                                ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                                ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                                ;-
030174  004737  015120                   JSR     PC,INDTPX      ;INITIALISE DATA PATTERN.

                                ;+
                                ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
                                ; SET LPR TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                                ;-
030200  013705  002240                    MOV     ACTLNS,R5      ;PASS THE ACTIVE LINE BIT MAP.
030204  012700  000206                    MOV     #206,R0        ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
030210  004737  017412                   JSR     PC,WTWLNC      ;INITIALISE THE LINE CONTROL REGISTER.
030214  012700  177670                    MOV     #177670,R0     ;PASS THE LPR CONTENTS.
030220  004737  017466                   JSR     PC,WTWLPR      ;SET THE LPR CONTENTS TO 38.4K BAUD.
030224  013704  000012                    MOV     10.,R4         ;PASS DELAY TIME OF 10 MILLI SECONDS.
030230  004737  014574                   JSR     PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.

                                ;+
                                ; SET UP LOOP FOR ALL ACTIVE LINES.
```

```
                                ; TEST THE STATE OF THE OAUTO BIT PRIOR TO TRANSMITTING THE DATA PATTERN.
                                ;  IF THE BIT IS CLEAR, THEN REPORT THE ERROR AND SKIP TRANSMITTING
                                ;  THE DATA PATTERN ON THE SELECTED LINE.
                                ; TRANSMIT A 224 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                                ; EMPTY THE FIFO, AND COUNT THE XOFF AND AN XON CHARS FOUND.
                                ;-
030234  005001                          CLR     R1              ;CLEAR THE LINE NUMBER COUNTER.
030236  005037  030526                   CLR     55$             ;CLEAR STORAGE FOR LINE NUMBER.
030242  012737  012122  004054  2$:      MOV     #5202.,ERRNBR   ;SET THE ERROR NUMBER TO 5202.
030250  004737  015656                   JSR     PC,PUFIFO       ;PURGE THE FIFO.
030254  103121                          BCC     50$             ;GO REPORT ERROR IF FIFO DID NOT PURGE.
030256  000241                           CLC                    ;CLEAR CARRY PRIOR TO ROTATING BIT MAP.
030260  006005                           ROR     R5              ;ROTATE THE BIT MAP INTO THE CARRY BIT.
030262  103107                          BCC     16$             ;BRANCH IF LINE IS INACTIVE.
                                ;+
                                ; TEST THE IAUTO BIT ON THE SELECTED ACTIVE LINE.
                                ; REPORT ERROR IF IT IS CLEAR.
                                ;  DO NOT TRANSMIT THE DATA PATTERN ON THE SELECTED LINE.
                                ;-
030264  005237  004054                    INC     ERRNBR          ;SET ERROR NUMBER TO 5203.
030270  010177  151752                    MOV     R1,@CSRA        ;SELECT LINE TO TEST.
030274  032777  000002  151754            BIT     #BIT1,@LNCTRA   ;TEST THE STATE OF THE IAUTO BIT ON THIS LINE.
030302  001004                           BNE     4$              ;SKIP ERROR IF IAUTO BIT SET.
030304  012702  010413                    MOV     #EM5202,R2      ;PASS THE CORRECT ERROR MESSAGE.
                                                                ; "IAUTO BIT FOUND CLEAR ON LINE NN"
030310                                    ERROR                   ;               >>>>> ERROR <<<<<.
030310  104460                                                                    TRAP    C$ERROR
030312  000473                           BR      16$             ;SKIP TRANSMITTING DATA PATTERN.
                                ;+
                                ; TRANSMIT DATA PATTERN TO FILL THE FIFO, 223 CHARS + 32 XOFF'S + XON.
                                ;-
030314  005237  004054  4$:       INC     ERRNBR          ;SET ERROR NUMBER TO 5204.
030320  012702  002712           MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
030324  012703  000337           MOV     #223.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
030330  004737  014634          JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
030334  103071                   BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
                                ;+
                                ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER PLUS XOFF
                                ; TO ARRIVE IN THE FIFO.
                                ;-
030336  005237  004054           INC     ERRNBR          ;SET ERROR NUMBER TO 5205.
030342  012701  170454           MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
030346  013702  002246           MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
030352  004737  017276          JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
030356  103060                  BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
030360  012704  000012           MOV     #10.,R4         ;PASS DELAY OF 10 MILLI SECS.
030364  004737  014574          JSR     PC,DELAY        ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
                                ;+
                                ; READ 256 CHARS FROM THE FIFO, COUNT ANY XOFF OR XON CHARS FOUND.
                                ;-
030370  005003                   CLR     R3              ;CLEAR XOFF COUNTER.
030372  005004                   CLR     R4              ;CLEAR XON COUNTER.
030374  005237  004054           INC     ERRNBR          ;INCREMENT ERROR NUMBER TO 5206.
```

```
7540  030400  012701  000400          MOV     #256.,R1        ;INITIALISE THE READ COUNTER.
7541  030404  017702  151640  6$:     MOV     @RBUFA,R2       ;READ CHAR FROM THE FIFO.
7542  030410  100043                  BPL     50$             ;GO REPORT ERROR IF FIFO EMPTY.
7543                                  ;+
7544                                  ; CHECK FOR BMP CODE IN THE FIFO.  SAVE ANY FOUND ON THE QUEUE.
7545                                  ;-
7546  030412  012700  170301          MOV     #170301,R0      ;SET UP BMP BIT MASK.
7547  030416  040200                  BIC     R2,R0           ;TRY TO CLEAR ALL THE BMP BITS.
7548  030420  001002                  BNE     8$              ;SKIP BMPSAV IF NOT A BMP CODE.
7549  030422  004737  016220          JSR     PC,SAVBMP       ;SAVE THE BMP CODE ON THE QUEUE.
7550                                  ;+
7551                                  ; CHECK FOR XOFF AND XON CHARACTERS.
7552                                  ;-
7553  030426  120227  000023  8$:     CMPB    R2,#23          ;IS IT AN XOFF CHARACTER?.
7554  030432  001001                  BNE     10$             ;NO, BRANCH TO SEE IF IT IS AN XON.
7555  030434  005203                  INC     R3              ;COUNT THE XOFF CHAR.
7556  030436  120227  000021  10$:    CMPB    R2,#21          ;IS IT AN XON CHARACTER?.
7557  030442  001001                  BNE     12$             ;NO, SKIP THE NEXT INSTRUCTION.
7558  030444  005204                  INC     R4              ;COUNT THE XON.
7559  030446  005301          12$:    DEC     R1              ;DECREMENT THE READ COUNT.
7560  030450  001355                  BNE     6$              ;LOOP TO READ THE NEXT CHAR.
7561                                  ;+
7562                                  ; VERIFY THAN AT LEAST 1 XOFF AND 1 XON WAS FOUND IN THE FIFO.
7563                                  ; REPORT ERROR IF NONE WERE FOUND.
7564                                  ;-
7565  030452  005703                  TST     R3              ;CHECK XOFF COUNT.
7566  030454  001403                  BEQ     14$             ;GO REPORT ERROR IF NONE FOUND.
7567  030456  020427  000001          CMP     R4,#1           ;CHECK XON COUNT = 1.
7568  030462  001407                  BEQ     16$             ;SKIP THE ERROR REPORT IF ONE XON WAS FOUND.
7569  030464  005237  004054  14$:    INC     ERRNBR          ;SET ERROR NUMBER TO 5207.
7570  030470  013701  030526          MOV     55$,R1          ;PASS THE LINE NUMBER TO BE REPORTED.
7571  030474  012702  010337          MOV     #EM5103,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
7572                                                          ; "IAUTO BIT BAD ON LINE NN".
7573  030500                          ERROR                   ;                >>>>> ERROR <<<<<.
7574  030500  104460                                                          TRAP    C$ERROR
7575                                  ;+
7576                                  ; CHECK IF ALL ACTIVE LINES HAVE BEEN TESTED.
7577                                  ;-
7578  030502  005237  030526  16$:    INC     55$             ;INCREMENT LINE NUMBER.
7579  030506  013701  030526          MOV     55$,R1          ;GET NUMBER OF THE NEXT LINE TO TEST.
7580  030512  005705                  TST     R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
7581  030514  001252                  BNE     2$              ;LOOP TO CHECK NEXT LINE.
7582  030516  000404                  BR      60$             ;EXIT TEST.
7583
7584  030520  004737  016456  50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
7585  030524  000401                  BR      60$             ;EXIT THIS TEST.
7586  030526  000000          55$:    .WORD   0               ;STORAGE FOR LINE NUMBER.
7587  030530  005037  002270  60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
7588
7589  030534                          ENDTST
7590  030534                                                          L10040:
7591  030534  104401                                                          TRAP    C$ETST
```

```
                                    .SBTTL  HARDWARE TEST           - FIFDAT -
                                    ;++*********************************************************************
                                    ;*                          - FIFO VALID DATA TEST -
                                    ;*
                                    ;*          THIS TEST VERIFIES THAT THE DUT IS CAPABLE OF HOLDING 256 VALID
                                    ;*          CHARACTERS IN ITS FIFO.
                                    ;*          THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
                                    ;*          INTERNAL LOOPBACK MODE.
                                    ;*          THE DATA FOUND IN THE FIFO IS COMPARED WITH THE EXPECTED DATA, AND ANY
                                    ;*          DISCREPANCIES ARE REPORTED.
                                    ;*          ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
                                    ;*          HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
                                    ;*          REPORTED LATER.
                                    ;*
                                    ;--*********************************************************************

 030536                                     BGNTST
 030536                                                                     T15::
 030536                                     SETPRI  #PRIO5          ;ALLOW LTC INTERRUPTS.
 030536  012700  000240                                                          MOV     #PRIO5,R0
 030542  104441                                                                  TRAP    C$SPRI
         000017                              TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
 030544  012737  000017  002272              MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.        (53)
 030552  012737  177777  002270              MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
 030560  012737  000001  004052              MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
 030566  012737  012265  004054              MOV     #5301.,ERRNBR  ;SET ERROR NUMBER TO 5301.
 030574  012737  010451  004056              MOV     #EM5301,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                                    ;+
                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                    ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 5301 <<<<<.
                                    ;-
 030602  004737  014460                     JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
 030606  103107                             BCC     60$             ;EXIT TEST IF FATAL ERROR FOUND.
                                    ;+
                                    ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
                                    ; INITIALISE 256 BYTE DATA PATTERN.
                                    ;-
 030610  004737  015010                     JSR     PC,FINACT       ;FIND AN ACTIVE LINE.
 030614  103104                             BCC     60$             ;EXIT IF NO ACTIVE LINES FOUND.
 030616  004737  015070                     JSR     PC,INDATP       ;INITIALISE THE DATA PATTERN.
                                    ;+
                                    ; TRANSMIT A 265 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                                    ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                                    ;-
                                            ;+
                                            ; SET INTERNAL LOOPBACK ON THE SELECTED LINE.
                                            ; TRANSMIT THE DATA PATTERN ON THE FIRST AVAILABLE ACTIVE LINE.
                                            ;-
 030622  012700  000204                      MOV    #204,R0         ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
 030626  004737  017412                     JSR     PC,WTWLNC       ;INITILAISE THE LINE CONTROL REGISTER.
 030632  012700  177670                      MOV    #177670,R0      ;PASS THE LPR CONTENTS.
 030636  004737  017466                     JSR     PC,WTWLPR       ;SET THE LPR CONTENTS TO 38.4K BAUD.
 030642  013704  000012                      MOV    10.,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
 030646  004737  014574                     JSR     PC,DELAY        ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
 030652  012702  002712                      MOV    #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
```

```
7648  030656  012703  000400                    MOV     #BUFMID-BUFBAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN.
7649  030662  005237  004054                    INC     ERRNBR            ;SET ERROR NUMBER TO 5302.
7650  030666  004737  014634                    JSR     PC,DODMA          ;TRANSMIT THE DATA PATTERN.
7651  030672  103053                            BCC     50$               ;ABORT TEST IF ERROR FOUND DURING DMA TX.
7652                                          ;+
7653                                          ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
7654                                          ; THE FIFO.
7655                                          ;-
7656  030674  005237  004054                    INC     ERRNBR            ;SET ERROR NUMBER TO 5303.
7657  030700  010103                            MOV     R1,R3             ;SAVE THE NUMBER OF THE SELECTED ACTIVE LINE.
7658  030702  012701  170454                    MOV     #170454,R1        ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
7659  030706  013702  002246                    MOV     CSRA,R2           ;PASS THE ADDRESS OF THE CSR.
7660  030712  004737  017276                    JSR     PC,WAIBIS         ;WAIT FOR DMA TO COMPLETE, TX ACTION SET.
7661  030716  103041                            BCC     50$               ;BRANCH IF FIFO EMPTY, ABORT THE TEST.
7662  030720  012704  000005                    MOV     #5,R4             ;PASS DELAY OF 5 MILLI SECS.
7663  030724  004737  014574                    JSR     PC,DELAY          ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
7664                                        ;+
7665                                        ; READ THE FIFO CHECKING FOR DATA CORRUPTION, REPORT ANY ERRORS FOUND.
7666                                        ; ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
7667                                        ;-
7668  030730  006303                            ASL     R3                ;MULTIPLY BY 2.
7669  030732  005004                            CLR     R4                ;INITIALISE THE EXPECTED DATA.
7670  030734  013705  002250                    MOV     RBUFA,R5          ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
7671  030740  012737  012270  004054  2$:       MOV     #5304.,ERRNBR     ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
7672  030746  011502                            MOV     (R5),R2           ;GET THE ACTUAL DATA FROM THE FIFO.
7673  030750  100024                            BPL     50$               ;ABORT THE TEST IF THE FIFO IS EMPTY.
7674                                        ;+
7675                                        ; CHECK IF THE READ CHARACTER IS A BMP CODE.
7676                                        ;  IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
7677                                        ;  ABORT THE TEST.
7678                                        ;-
7679  030752  005237  004054                    INC     ERRNBR            ;SET ERROR NUMBER TO 5305.
7680  030756  004737  014360                    JSR     PC,CHKBMP         ;CHECK IF CHARACTER IS A BMP CODE.
7681  030762  103002                            BCC     4$                ;BRANCH IF NOT A BMP CODE.
7682  030764                                    ERROR                     ;                >>>>> ERROR 5305 <<<<<.
7683  030764  104460                                                                                TRAP    C$ERROR
7684  030766  000417                            BR      60$               ;ABORT THIS TEST.
7685
7686  030770  005237  004054            4$:     INC     ERRNBR            ;SET ERROR NUMBER TO 5306.
7687  030774  120402                            CMPB    R4,R2             ;COMPARE THE EXPECTED WITH THE ACTUAL DATA.
7688  030776  001406                            BEQ     8$                ;SKIP ERROR REPORT IF DATA IS OK.
7689  031000  012737  013376  004060             MOV     #ER9002,ERRBLK    ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
7690  031006  012701  010476                     MOV     #EM5302,R1        ;PASS THE MESSAGE TO BE REPORTED.
7691                                            ;REPORT THE ERROR ''FIFO BAD, DATA FIELD CORRUPTED''
7692  031012                            6$:     ERROR                     ;                >>>>> ERROR 5306 <<<<<.
7693  031012  104460                                                                                TRAP    C$ERROR
7694  031014  105204                    8$:     INCB    R4                ;INCREMENT THE EXPECTED DATA.
7695  031016  001350                            BNE     2$                ;LOOP IF NOT DONE.
7696  031020  000402                            BR      60$               ;EXIT
7697
7698  031022  004737  016456            50$:    JSR     PC,TSABRT         ;ABORT THE TEST, REASON SHOWN BY ERROR NUMBER.
7699  031026  005037  002270            60$:    CLR     CTRLCF            ;INDICATE THAT WE ARE NOT WITHIN A TEST.
7700
7701  031032                                    ENDTST
7702  031032                                                                      L10041:
7703  031032  104401                                                                        TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - FI3QLI -
                                ;++*************************************************************************
                                ;*                      - FIFO 3/4 LEVEL INACTIVE TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE DUT'S FIFO 3/4 LEVEL ALARM SYSTEM
                                ;*      REMAINS INACTIVE WHILE IT CONTAINS 191 CHARACTERS OR LESS.
                                ;*      THE TEST LOOKS FOR AN XOFF (ASCII DC3) CHARACTER IN THE FIFO.
                                ;*      IF ANY XOFF'S ARE FOUND AN ERROR WILL BE REPORTED AND THE TEST ABORTED.
                                ;*      ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
                                ;*      HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
                                ;*      REPORTED LATER.
                                ;*      THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
                                ;*      INTERNAL LOOPBACK MODE.
                                ;*
                                ;--*************************************************************************

                                        BGNTST
                                                                                T16::
                                        SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
                                                                                MOV     #PRI05,R0
031034  012700  000240                                                          TRAP    C$SPRI
031040  104441
        000020                   TNUM == TNUM + 1               ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
031042  012737  000020  002272   MOV     #TNUM,TSTNUM           ;SET UP THE TEST NUMBER.        (54)
031050  012737  177777  002270   MOV     #-1,CTRLCF             ;INDICATE THAT WE ARE IN A TEST.
031056  012737  000001  004052   MOV     #1,ERRTYP              ;SET FATAL ERROR TYPE IN ERROR TABLE.
031064  012737  012431  004054   MOV     #5401.,ERRNBR          ;SET ERROR NUMBER TO 5401.
031072  012737  010626  004056   MOV     #EM5401,ERRMSG         ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
031100  012737  012640  004060   MOV     #ER0503,ERRBLK         ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 5401 <<<<<.
                                ;-
031106  004737  014460                JSR     PC,CLNRST              ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
031112  103111                      BCC     60$                    ;EXIT TEST IF FATAL ERROR FOUND.
                                ;+
                                ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
                                ;-
031114  004737  015010                JSR     PC,FINACT              ;FIND THE NUMBER OF THE FIRST ACTIVE LINE.
031120  103106                      BCC     60$                    ;EXIT IF NO LINES ARE AVAILABLE.

                                ;+
                                ; INITIALIZE THE 256 BYTE DATA PATTERN.
                                ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                                ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                                ;-
031122  004737  015120                JSR     PC,INDTPX              ;INITIALISE THE DATA PATTERN.
                                ;+
                                ; TRANSMIT A 191 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                                ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                                ;-
                                    ;+
                                    ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RX ON THE SELECTED LINE.
                                    ; TRANSMIT THE DATA PATTERN ON THE FIRST AVAILABLE ACTIVE LINE.
                                    ;-
031126  012700  000206                  MOV     #206,R0                ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
```

```
7760  031132  004737  017412                  JSR     PC,WTWLNC       ;INITILAISE THE LINE CONTROL REGISTER.
7761  031136  012700  177670                   MOV     #177670,R0      ;PASS THE LPR CONTENTS.
7762  031142  004737  017466                  JSR     PC,WTWLPR       ;SET THE LPR CONTENTS TO 38.4K BAUD.
7763  031146  013704  000012                   MOV     10.,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
7764  031152  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
7765  031156  012702  002712                   MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
7766  031162  012703  000277                   MOV     #191.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
7767  031166  004737  014634                  JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
7768  031172  103057                           BCC     50$             ;IF ERROR FOUND DURING DMA THEN ABORT TEST.
7769
7770                                          ;+
7771                                          ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
7772                                          ; THE FIFO.
7773                                          ;-
7774  031174  005237  004054                   INC     ERRNBR          ;SET ERROR NUMBER TO 5402.
7775  031200  012701  170454                   MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
7776  031204  013702  002246                   MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
7777  031210  004737  017276                  JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
7778  031214  103046                          BCC     50$             ;IF FIFO EMPTY, REPORT ERROR, ABORT THE TEST.
7779  031216  012704  000005                   MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
7780  031222  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
7781                                      ;+
7782                                      ; READ THE CONTENTS OF THE FIFO.  IF ANY OF THE FOLLOWING CONDITIONS OCCUR
7783                                      ; REPORT THE ERROR AND ABORT THE TEST;
7784                                      ;  FIFO EMPTY TOO SOON.
7785                                      ;  BMP CODE FOUND.
7786                                      ;  XOFF CODE FOUND.
7787                                      ;  EXTRA (192) CHARACTER FOUND IN FIFO.
7788                                      ;-
7789  031226  005004                           CLR     R4              ;CLEAR THE CHARACTER COUNT.
7790  031230  013705  002250                   MOV     RBUFA,R5        ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
7791  031234  012737  012267  004054  2$:      MOV     #5303.,ERRNBR   ;SET ERROR NUMBER TO 5403.
7792  031242  011502                           MOV     (R5),R2         ;GET THE ACTUAL DATA FROM THE FIFO.
7793  031244  100032                          BPL     50$             ;FIFO EMPTY, ABORT TEST.
7794  031246  005204                           INC     R4              ;COUNT THE CHARACTER.
7795                                    ;+
7796                                    ; CHECK IF THE READ CHARACTER IS A BMP CODE.
7797                                    ;  IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
7798                                    ;  ABORT THE TEST.
7799                                    ;-
7800  031250  005237  004054                   INC     ERRNBR          ;SET ERROR NUMBER TO 5404.
7801  031254  004737  014360                  JSR     PC,CHKBMP       ;CHECK IF CHARACTER IS A BMP CODE.
7802  031260  103001                          BCC     4$              ;BRANCH IF NOT A BMP CODE.
7803                                          ;REPORT ERROR "BMP CODE FOUND IN FIFO, TEST INVALIDATED".
7804  031262  000421                          BR      8$              ;REPORT THE ERROR AND ABORT THE TEST.
7805                                    ;+
7806                                    ; CHECK IF THE CHARACTER IS AN XOFF.  REPORT THE ERROR IF ONE IS FOUND.
7807                                    ;-
7808
7809  031264  005237  004054  4$:       INC     ERRNBR          ;SET ERROR NUMBER TO 5405.
7810  031270  122702  000023                   CMPB    #23,R2          ;CHECK IF THE READ DATA IS AN XOFF.
7811  031274  001003                          BNE     6$              ;BRANCH IF NOT AN XOFF.
7812  031276  012701  010665                   MOV     #EM5402,R1      ;PASS THE MESSAGE TO BE REPORTED.
7813                                          ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
7814  031302  000411                          BR      8$              ;GO REPORT THE ERROR AND ABORT THE TEST.
7815
```

```
7816  031304  005237  004054          6$:     INC     ERRNBR          ;SET ERROR NUMBER TO 5406.
7817  031310  020427  000277                  CMP     R4,#191.        ;CHECK IF WE HAVE READ ALL THE CHARACTERS.
7818  031314  001347                  BNE     2$              ;LOOP BACK TO GET THE NEXT CHARACTER.
7819  031316  011502                  MOV     (R5),R2         ;TRY TO READ AN EXTRA CHARACTER FROM THE FIFO.
7820  031320  100006                  BPL     60$             ;EXIT IF NON FOUND.
7821  031322  012701  010665          MOV     #EM5402,R1      ;PASS THE MESSAGE TO BE REPORTED.
7822                                  ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
7823
7824  031326                  8$:     ERROR                   ;       >>>>> ERRORS 5304 THRU 5306 <<<<<.
                                                                                        TRAP    C$ERROR
7825  031326  104460
7826  031330  000402                  BR      60$             ;EXIT THE TEST.
7827
7828                                  ;       >>>>> ERRORS 5402 AND 5403 <<<<<.
7829  031332  004737  016456          50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
7830  031336  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
7831
7832  031342                          ENDTST
7833  031342                                                                  L10042:
7834  031342  104401                                                                  TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST           - FI3QLA -
                        ;++*************************************************************************
                        ;*                      - FIFO 3/4 LEVEL ACTIVE TEST -
                        ;*
                        ;*      THIS TEST VERIFIES THAT THE DUT'S FIFO 3/4 LEVEL ALARM SYSTEM
                        ;*      BECOMES ACTIVE WHEN THE FIFO CONTAINS > 192 CHARACTERS.
                        ;*      THE TEST COMPARES THE ACTUAL NUMBER OF XOFF (ASCII DC3)
                        ;*      CHARACTERS THAT ARE FOUND IN THE FIFO WITH THE EXPECTED NUMBER.
                        ;*      AN ERROR WILL BE REPORTED, IF THE COUNTS ARE FOUND TO DIFFER.
                        ;*      ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
                        ;*      HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
                        ;*      REPORTED LATER.
                        ;*      THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
                        ;*      INTERNAL LOOPBACK MODE.
                        ;*
                        ;--*************************************************************************

                                BGNTST
                                                                        T17::
                                SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
031344  012700  000240                                                  MOV     #PRI05,R0
031350  104441                                                         TRAP    C$SPRI
        000021
                                 TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
031352  012737  000021  002272    MOV     #TNUM,TSTNUM   ;SET UP THE TEST NUMBER.        (55)
031360  012737  177777  002270    MOV     #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
031366  012737  000001  004052    MOV     #1,ERRTYP      ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
031374  012737  012575  004054    MOV     #5501.,ERRNBR  ;SET ERROR NUMBER TO 5501.
031402  012737  010726  004056    MOV     #EM5501,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.

                        ;+
                        ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                        ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                        ; THIS SUBROUTINE REPORTS ERROR >>>>> 5501 <<<<<.
                        ;-
031410  004737  014460          JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
031414  103402                BCS     .+6             ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
031416  000137  032036          JMP     60$             ;EXIT TEST FATAL ERROR FOUND.

                        ;+
                        ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
                        ;-
031422  004737  015010          JSR     PC,FINACT       ;FIND AN ACTIVE LINE.
031426  103402                BCS     .+6             ;SKIP EXIT OF TEST IF ACTIVE LINE FOUND.
031430  000137  032036          JMP     60$             ;EXIT TEST.

                        ;+
                        ; INITIALIZE THE 256 BYTE DATA PATTERN.
                        ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                        ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                        ;-
031434  004737  015120          JSR     PC,INDTPX       ;INITIALISE DATA PATTERN.

                        ;+
                        ; TRANSMIT A 256 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                        ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                        ;-

                            ;+
                            ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
                            ; TRANSMIT THE FIRST 191 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
                            ;-
```

```
 7891  031440  005237  004054          2$:     INC     ERRNBR          ;SET ERROR NUMBER TO 5502.
 7892  031444  012700  000206                  MOV     #206,R0         ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
 7893  031450  004737  017412                  JSR     PC,WTWLNC       ;INITIALISE THE LINE CONTROL REGISTER.
 7894  031454  012700  177670                  MOV     #177670,R0      ;PASS THE LPR CONTENTS.
 7895  031460  004737  017466                  JSR     PC,WTWLPR       ;SET THE LPR CONTENTS TO 38.4K BAUD.
 7896  031464  013704  000012                  MOV     10.,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
 7897  031470  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
 7898  031474  010105                          MOV     R1,R5           ;COPY THE LINE NUMBER.
 7899  031476  012702  002712                  MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
 7900  031502  012703  000277                  MOV     #191.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
 7901  031506  004737  014634                  JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
 7902  031512  103147                          BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
 7903
 7904                                  ;+
 7905                                  ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
 7906                                  ; THE FIFO.
 7907                                  ;-
 7908  031514  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5503.
 7909  031520  012701  170454                  MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
 7910  031524  013702  002246                  MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
 7911  031530  004737  017276                  JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
 7912  031534  103136                          BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
 7913  031536  012704  000005                  MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
 7914  031542  004737  014574                  JSR     PC,DELAY        ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
 7915
 7916                                  ;+
 7917                                  ; TRANSMIT A NULL CHARACTER WHICH WILL CAUSE AN XOFF TO BE GENERATED.
 7918  031546  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5504.
 7919  031552  010501                          MOV     R5,R1           ;PASS THE LINE NUMBER.
 7920  031554  012702  002712                  MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
 7921  031560  012703  000001                  MOV     #1,R3           ;PASS THE NUMBER OF
 7922  031564  004737  014634                  JSR     PC,DODMA        ;TX A NULL CHARACTER TO CAUSE AN XOFF.
 7923  031570  103120                          BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
 7924
 7925                                  ;+
 7926                                  ; WAIT FOR THE XOFF TO BE RECEIVED BEFORE TX THE NEXT 42 CHARACTERS
 7927                                  ; WHICH WILL CAUSE A FURTHER 21 XOFF'S TO BE GENERATED.
 7928                                  ;-
 7929  031572  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5505.
 7930  031576  012701  170012                  MOV     #170012,R1      ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
 7931  031602  013702  002246                  MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
 7932  031606  004737  017276                  JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
 7933  031612  103107                          BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
 7934  031614  012704  000005                  MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
 7935  031620  004737  014574                  JSR     PC,DELAY        ;WAIT FOR XOFF TO GET INTO THE FIFO.
 7936
 7937                                  ;+
 7938                                  ; INITIALISE THE 256 BYTE DATA PATTERN TO ALL NULLS.
 7939  031624  012702  002712                  ;-
 7940  031630  105022                  4$:     MOV     #BUFBAS,R2      ;INITIALIZE THE DATA PATTERN TO BE
 7941  031632  020227  003312                  CLRB    (R2)+           ;  ALL NULLS.
 7942  031636  103774                          CMP     R2,#BUFMID      ;
 7943                                          BLO     4$              ;
 7944
 7945                                  ;+
 7946                                  ; TRANSMIT A FURTHER 31 NULL CHARACTERS WHICH WILL CAUSE 31 XOFF'S TO BE
                                       ; GENERATED.
```

```
                                        ;-
031640  005237  004054                      INC     ERRNBR          ;SET ERROR NUMBER TO 5506.
031644  010501                              MOV     R5,R1           ;PASS THE LINE NUMBER.
031646  012702  002712                      MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
031652  012703  000037                      MOV     #31.,R3         ;PASS THE LENGTH OF THE DATA PATTERN.
031656  004737  014634                    JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
031662  103063                             BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
                                        ;+
                                        ; WAIT FOR THE XOFF'S AND THE NULL CHARACTERS  TO BE RECEIVED.
                                        ;-
031664  005237  004054                      INC     ERRNBR          ;SET ERROR NUMBER TO 5507.
031670  012701  170454                      MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
031674  013702  002246                      MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
031700  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
031704  103052                           BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
031706  012704  000005                      MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
031712  004737  014574                    JSR     PC,DELAY        ;WAIT FOR XOFF TO GET INTO THE FIFO.
                                        ;+
                                        ; READ THE FIFO UNTIL EMPTY, COUNTING THE NUMBER OF XOFF CHARACTERS
                                        ; THAT ARE FOUND.
                                        ;-
031716  005004                              CLR     R4              ;CLEAR CHARACTER COUNTER.
031720  005003                              CLR     R3              ;CLEAR THE XOFF FOUND COUNTER.
031722  012701  170001                      MOV     #170001,R1      ;INDICATE TO TEST DATA.VALID BIT, TIME-OUT 1MS.
031726  012737  012604  004054  6$:         MOV     #5508.,ERRNBR   ;SET UP ERROR NUMBER EACH TIME AROUND THE LOOP.
031734  013702  002250                      MOV     RBUFA,R2        ;INDICATE TO CHECK RECEIVE BUFFER REGISTER.
031740  004737  017276                    JSR     PC,WAIBIS       ;WAIT FOR RECEIVED CHAR OR TIME-OUT.
031744  103032                           BCC     50$             ;GO REPORT ERROR IF FIFO EMPTY.
031746  005204                              INC     R4              ;COUNT THE CHARACTER.
                                        ;+
                                        ; CHECK IF FOR BMP CODES IN THE FIFO, ABORT THE TEST IF ANY ARE FOUND.
                                        ;  SAVE THE BMP CODE ON THE QUEUE TO BE REPORTED LATER.
                                        ;-
031750  005237  004054                          INC     ERRNBR          ;SET ERROR NUMBER TO 5509.
031754  004737  014360                        JSR     PC,CHKBMP       ;CHECK IF WE HAVE GOT A BMP CODE.
031760  103422                               BCS     12$             ;GO REPORT THE ERROR IF WE FOUND A BMP CODE.
                                        ;+
                                        ; CHECK FOR XOFF CHARACTER.
                                        ;-
031762  122702  000023                  8$:     CMPB    #23,R2          ;CHECK IF THE RECEIVED CHARACTER WAS AN XOFF.
031766  001001                               BNE     10$             ;BRANCH IF CHARACTER WAS NOT AN XOFF.
031770  005203                               INC     R3              ;INCREMENT XOFF FOUND COUNT.
                                        ;+
                                        ; CHECK IF ALL THE CHARACTERS INCLUDING THE XON HAVE BEEN REMOVED.
                                        ;-
031772  020427  000400                  10$:    CMP     R4,#256.        ;CHECK IF WE HAVE REMOVED ALL THE CHARACTERS.
031776  002753                               BLT     6$              ;GO GET THE NEXT CHAR IF WE HAVE NOT FINISHED.

                                        ;+
                                        ; CHECK IF THE CORRECT NUMBER OF XOFF'S WERE FOUND IN THE FIFO,
                                        ; REPORT ERROR IF COUNT IS INCORRECT.
                                        ;-
032000  013737  012606  004054                  MOV     5510.,ERRNBR    ;SET UP THE ERROR NUMBER TO 5510.
032006  022703  000040                       CMP     #32.,R3         ;COMPARE EXPECTED XOFF COUNT WITH ACTUAL COUNT.
032012  001411                               BEQ     60$             ;EXIT TEST IF SUCCESS.
032014  012737  012640  004060                 MOV     #ER0503,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
```

```
 8003  032022  012701  010665                  MOV     #EM5402,R1      ;PASS THE MESSAGE TO BE REPORTED.
 8004                                          ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
 8005  032026                          12$:    ERROR                   ;               >>>>> ERROR <<<<<.
 8006  032026  104460                                                                          TRAP    C$ERROR
 8007  032030  000402                          BR      60$             ;ABORT THE TEST.
 8008
 8009  032032  004737  016456          50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED.  ERROR # SHOWS REASON.
 8010  032036  005037  002270          60$:     CLR    CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
 8011
 8012  032042                          ENDTST
 8013  032042                                                                        L10043:
 8014  032042  104401                                                                          TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST          - FI3QAI -
                        ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                        ;*                  - FIFO 3/4 ALARM LEVEL ACTIVE/INACTIVE TEST -
                        ;*
                        ;*      THIS TEST VERIFIES THAT THE DUT'S FIFO 3/4 LEVEL ALARM SYSTEM
                        ;*      BECOMES ACTIVE AND INACTIVE AT THE CORRECT LEVELS.
                        ;*      ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
                        ;*      HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
                        ;*      REPORTED LATER.
                        ;*      THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
                        ;*      INTERNAL LOOPBACK MODE.
                        ;*
                        ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                BGNTST
                                                                        T18::
                                SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
032044  012700  000240                                                  MOV     #PRI05,R0
032050  104441                                                          TRAP    C$SPRI
                                 TNUM == TNUM + 1       ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
032052  012737  000022  002272    MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (56)
032060  012737  177777  002270    MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
032066  012737  000001  004052    MOV     #1,ERRTYP       ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
032074  012737  012741  004054    MOV     #5601.,ERRNBR   ;SET ERROR NUMBER TO 5601.
032102  012737  010763  004056    MOV     #EM5601,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.

                        ;+
                        ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
                        ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
                        ; THIS SUBROUTINE REPORTS ERROR >>>>> 5601 <<<<<.
                        ;-
032110  004737  014460            JSR     PC,CLNRST       ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
032114  103402                    BCS     2$              ;SKIP EXITING TEST A SUCCESSFUL RESET.
032116  000137  032534            JMP     60$             ;EXIT THIS TEST.
032122                  2$:
                        ;+
                        ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
                        ;-
032122  004737  015010            JSR     PC,FINACT       ;FIND AN ACTIVE LINE.
032126  103402                    BCS     .+6             ;SKIP EXIT OF TEST IF ACTIVE LINE FOUND.
032130  000137  032534            JMP     60$             ;EXIT TEST.
                        ;+
                        ; INITIALIZE THE 256 BYTE DATA PATTERN.
                        ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
                        ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
                        ;-
032134  004737  015120            JSR     PC,INDTPX       ;INITIALISE THE DATA PATTERN.
                        ;+
                        ; TRANSMIT A 256 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
                        ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
                        ;-
                            ;+
                            ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
                            ; TRANSMIT THE FIRST 191 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
                            ;-
032140  005237  004054                INC     ERRNBR          ;SET ERROR NUMBER TO 5602.
032144  012700  000206                MOV     #206,R0         ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
```

```
8071  032150  004737  017412          JSR     PC,WTWLNC       ;INITILAISE THE LINE CONTROL REGISTER.
8072  032154  012700  177670           MOV     #177670,R0      ;PASS THE LPR CONTENTS.
8073  032160  004737  017466          JSR     PC,WTWLPR       ;SET THE LPR CONTENTS TO 38.4K BAUD.
8074  032164  013704  000012           MOV     10.,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
8075  032170  004737  014574          JSR     PC,DELAY        ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
8076  032174  010105                   MOV     R1,R5           ;COPY THE LINE NUMBER.
8077  032176  012702  002712           MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
8078  032202  012703  000277           MOV     #191.,R3        ;PASS THE LENGTH OF THE DATA PATTERN.
8079  032206  004737  014634          JSR     PC,DODMA        ;TRANSMIT THE DATA PATTERN.
8080  032212  103146                   BCC     50$             ;EXIT IF ERROR FOUND DURING DMA TX.
8081                                  ;+
8082                                  ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
8083                                  ; THE FIFO.
8084                                  ;-
8085  032214  005237  004054              INC     ERRNBR          ;SET ERROR NUMBER TO 5603.
8086  032220  012701  170454           MOV     #170454,R1      ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
8087  032224  013702  002246           MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
8088  032230  004737  017276          JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
8089  032234  103135                  BCC     50$             ;BRANCH IF FIFO EMPTY, ABORT THE TEST.
8090  032236  012704  000005           MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
8091  032242  004737  014574          JSR     PC,DELAY        ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
8092
8093                                  ;+
8094                                  ; TRANSMIT A NULL CHARACTER WHICH WILL CAUSE AN XOFF TO BE GENERATED.
8095                                  ;-
8096  032246  005237  004054              INC     ERRNBR          ;SET ERROR NUMBER TO 5604.
8097  032252  010501                   MOV     R5,R1           ;PASS THE LINE NUMBER.
8098  032254  012702  002712           MOV     #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
8099  032260  012703  000001           MOV     #1,R3           ;PASS THE NUMBER OF
8100  032264  004737  014634          JSR     PC,DODMA        ;TX A NULL CHARACTER TO CAUSE AN XOFF.
8101  032270  103117                   BCC     50$             ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
8102                                  ;+
8103                                  ; WAIT FOR THE XOFF TO BE RECEIVED BEFORE CONTINUING THE TEST.
8104                                  ;-
8105  032272  005237  004054              INC     ERRNBR          ;SET ERROR NUMBER TO 5605.
8106  032276  012701  170012           MOV     #170012,R1      ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
8107  032302  013702  002246           MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
8108  032306  004737  017276          JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
8109  032312  103106                  BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
8110  032314  012704  000005           MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
8111  032320  004737  014574          JSR     PC,DELAY        ;WAIT FOR XOFF TO GET INTO THE FIFO.
8112
8113  032324  010577  147716              MOV     R5,@CSRA        ;SELECT THE LINE READY FOR TRANSMISSION.
8114                                  ;+
8115                                  ; READ THREE CHARACTERS, TRANSMIT ONE CHARACTER UNTIL THE FIRST 192 CHARACTERS
8116                                  ; HAVE BEEN READ FROM THE FIFO, IE UNTIL THE HALF LEVEL IS REACHED.
8117                                  ;  THEN READ THE FIFO UNTIL EMPTY.
8118                                  ; COUNT ALL XOFF'S THAT ARE DETECTED.
8119                                  ;-
8120  032330  005005                   CLR     R5              ;CLEAR THE TX FLAG.
8121  032332  005004                   CLR     R4              ;CLEAR THE CHARACTER COUNTER.
8122  032334  012703  000300           MOV     #192.,R3        ;SET UP READ COUNTER FOR THE FIRST 192 CHARS.
8123
8124  032340  012700  000003  4$:      MOV     #3,R0           ;SET READ COUNTER.
8125  032344  012701  170005  6$:      MOV     #170005,R1      ;INDICATE TO TEST DATA.VALID BIT, TIME-OUT 5MS.
8126  032350  013702  002250           MOV     RBUFA,R2        ;INDICATE TO CHECK RECEIVE BUFFER REGISTER.
```

```
 8127  032354  004737  017276                JSR     PC,WAIBIS       ;WAIT FOR RECEIVED CHAR OR TIME-OUT.
 8128  032360  103046                        BCC     14$             ;EXIT LOOP IF TIME-OUT, FIFO EMPTY.
 8129  032362  005300                         DEC     R0              ;DECREMENT READ COUNTER.
 8130  032364  005303                         DEC     R3              ;DECREMENT CHAR COUNTER.
 8131  032366  003002                         BGT     8$              ;SKIP DISBL'G TX IF FIRST 192 CHARS NOT READ.
 8132  032370  052705  100000                 BIS     #BIT15,R5       ;DISABLE ANY FURTHER TRANSMISSIONS.
 8133                                  ;+
 8134                                  ; CHECK IF THE READ CHARACTER IS A BMP CODE.
 8135                                  ;  IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
 8136                                  ;  ABORT THE TEST.
 8137                                  ;-
 8138  032374  012737  012746  004054  8$:      MOV     #5606.,ERRNBR   ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
 8139  032402  004737  014360                JSR     PC,CHKBMP       ;CHECK IF CHARACTER IS A BMP CODE.
 8140  032406  103446                        BCS     16$             ;GO REPORT ERROR AND ABORT TEST IF BMP FOUND.
 8141                                  ;+
 8142                                  ; CHECK FOR XOFF CHARACTER.  IF ONE IS FOUND, COUNT IT.
 8143                                  ; TRANSMIT A NULL CHARACTER UNTIL THE FIRST 192 CHARS HAVE BEEN READ.
 8144                                  ;-
 8145  032410  122702  000023          10$:     CMPB    #23,R2          ;CHECK IF THE RECEIVED CHARACTER WAS AN XOFF.
 8146  032414  001001                        BNE     12$             ;BRANCH IF CHARACTER WAS NOT AN XOFF.
 8147  032416  005204                         INC     R4              ;INCREMENT THE XOFF CHAR FOUND COUNTER.
 8148
 8149  032420  005700                  12$:    TST     R0              ;CHECK READ COUNT, TO SEE IF A CHAR CAN BE TX.
 8150  032422  001350                        BNE     6$              ;BRANCH IF 3 CHARS NAVE NOT YET BEEN READ.
 8151  032424  005705                         TST     R5              ;CHECK THE TRANSMISSION ENABLED FLAG.
 8152  032426  100744                        BMI     4$              ;SKIP TRANSMITTING A CHARACTER IF TX DISABLED.
 8153  032430  012777  100000  147612         MOV     #100000,@TXCHA  ;TX A NULL CHARACTER.
 8154  032436  010446                         MOV     R4,-(SP)        ;SAVE THE XOFF COUNT ON THE STACK.
 8155                                  ;+
 8156                                  ; WAIT FOR THE CHARACTER TO BE RECEIVED BEFORE CONTINUING THE TEST.
 8157                                  ;-
 8158  032440  005237  004054             INC     ERRNBR          ;SET ERROR NUMBER TO 5607.
 8159  032444  012701  170012             MOV     #170012,R1      ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
 8160  032450  013702  002246             MOV     CSRA,R2         ;PASS THE ADDRESS OF THE CSR.
 8161  032454  004737  017276            JSR     PC,WAIBIS       ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
 8162  032460  103023                    BCC     50$             ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
 8163  032462  012704  000005             MOV     #5,R4           ;PASS DELAY OF 5 MILLI SECS.
 8164  032466  004737  014574            JSR     PC,DELAY        ;WAIT FOR XOFF TO GET INTO THE FIFO.
 8165  032472  012604                     MOV     (SP)+,R4        ;RESTORE THE XOFF COUNT.
 8166  032474  000721                    BR      4$              ;GO RESET THE READ COUNT AND GET NEXT CHAR.
 8167
 8168                                  ;+
 8169                                  ; CHECK IF THE CORRECT NUMBER OF XOFF'S WERE FOUND IN THE FIFO
 8170                                  ; REPORT ERROR IF COUNT IS INCORRECT.
 8171                                  ;-
 8172  032476  012737  012750  004054  14$:     MOV     #5608.,ERRNBR   ;SET ERROR NUMBER TO 5608.
 8173  032504  020427  000077             CMP     R4,#63.         ;COMPARE THE EXPECTED AND ACTUAL XOFF COUNTS.
 8174  032510  001411                    BEQ     60$             ;EXIT TEST IF SUCCESS.
 8175  032512  012737  012640  004060     MOV     #ER0503,ERRBLK  ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
 8176  032520  012701  010665             MOV     #EM5402,R1      ;PASS THE MESSAGE TO BE REPORTED.
 8177                                  ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
 8178  032524                          16$:    ERROR                   ;                >>>>> ERROR <<<<<.
 8179  032524  104460                                                                    TRAP    C$ERROR
 8180  032526  000402                    BR      60$             ;EXIT THIS TEST.
 8181
 8182  032530  004737  016456          50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED.  ERROR # INDICATES FAULT.
```

```
 8183  032534  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
 8184
 8185  032540                                  ENDTST
 8186  032540                                                                  L10044:
 8187  032540  104401                                                                  TRAP    C$ETST
```

```
8188                                    .SBTTL  HARDWARE TEST           - FIHAVL -
8189                                    ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8190                                    ;*                  - FIFO HALF LEVEL ACTIVE/INACTIVE TEST -
8191                                    ;*
8192                                    ;*      THIS TEST CHECKS THAT THE DUT'S FIFO HALF LEVEL ALARM SYSTEM
8193                                    ;*      BECOMES ACTIVE AND INACTIVE AT THE CORRECT LEVELS.
8194                                    ;*      ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
8195                                    ;*      HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
8196                                    ;*      REPORTED LATER.
8197                                    ;*      THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
8198                                    ;*      INTERNAL LOOPBACK MODE.
8199                                    ;*
8200                                    ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8201
8202  032542                                    BGNTST
8203  032542                                                                          T19::
8204  032542                                    SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
8205  032542  012700  000240                                                        MOV     #PRI05,R0
8206  032546  104441                                                               TRAP    C$SPRI
8207          000023                     TNUM == TNUM + 1            ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8208  032550  012737  000023  002272     MOV     #TNUM,TSTNUM        ;SET UP THE TEST NUMBER.            (57)
8209  032556  012737  177777  002270     MOV     #-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
8210  032564  012737  000001  004052     MOV     #1,ERRTYP           ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
8211  032572  012737  013105  004054     MOV     #5701.,ERRNBR       ;SET ERROR NUMBER TO 5701.
8212  032600  012737  011031  004056     MOV     #EM5701,ERRMSG      ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8213  032606  012737  012640  004060     MOV     #ER0503,ERRBLK      ;SELECT THE ERROR REPORTING ROUTINE.
8214                                    ;+
8215                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
8216                                    ;  CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
8217                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 5701 <<<<<.
8218                                    ;-
8219  032614  004737  014460                   JSR     PC,CLNRST           ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
8220  032620  103402                           BCS     2$                  ;SKIP EXITING TEST A SUCCESSFUL RESET.
8221  032622  000137  033206                   JMP     60$                 ;EXIT THIS TEST.
8222  032626                                 2$:
8223                                    ;+
8224                                    ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
8225                                    ;-
8226  032626  004737  015010                   JSR     PC,FINACT           ;FIND AN ACTIVE LINE.
8227  032632  103165                           BCC     60$                 ;EXIT IF NO ACTIVE LINES AVAILABLE.
8228                                    ;+
8229                                    ; INITIALIZE THE 256 BYTE DATA PATTERN.
8230                                    ;  ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
8231                                    ;  NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
8232                                    ;-
8233  032634  004737  015120                   JSR     PC,INDTPX           ;INITIALISE THE DATA PATTERN.
8234                                    ;+
8235                                    ; FILL THE FIFO BY TRANSMITTING 225 CHARS (IE 225 + 31 XOFF'S).
8236                                    ; TRANSMIT DATA PATTERN USING DMA, ON A SINGLE CHANNEL
8237                                    ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
8238                                    ;-
8239                                         ;+
8240                                         ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
8241                                         ; TRANSMIT THE 225 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
8242                                         ;-
8243  032640  005237  004054                      INC     ERRNBR              ;SET ERROR NUMBER TO 5702.
```

```
 8244  032644  004737  016332                JSR     PC,SETPAR       ;SET UP PARAMETERS FOR TRANSMISSION.
 8245  032650  012700  000341                 MOV     #225.,R0        ;PASS LENGTH OF DATA PATTERN.
 8246  032654  004737  016570                JSR     PC,TXDATP       ;TRANSMIT DATA PATTERN.
 8247  032660  103150                        BCC     50$             ;EXIT IF ERROR FOUND DURING TX.
 8248  032662  010105                         MOV     R1,R5           ;COPY THE LINE NUMBER.
 8249                                      ;+
 8250                                      ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
 8251                                      ; THE FIFO.
 8252                                      ;-
 8253  032664  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5703.
 8254  032670  004737  017352                JSR     PC,WAITTX       ;WAIT FOR TRANSMISSION TO COMPLETE.
 8255  032674  103142                        BCC     50$             ;GO REPORT ERROR IF TX FAILED TO COMPLETE.
 8256                                    ;+
 8257                                    ; READ THE FIRST 130 CHARACTERS FROM THE FIFO, IF ANY XON'S ARE FOUND
 8258                                    ; REPORT THE ERROR. IF ANY BMP CODES ARE FOUND THEN SAVE THEM ON THE QUEUE
 8259                                    ; AND ABORT THE TEST.
 8260                                    ;-
 8261  032676  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5704.
 8262  032702  012700  000202                 MOV     #130.,R0        ;PASS THE NUMBER OF CHARS TO READ.
 8263  032706  004737  015740                JSR     PC,READBX       ;READ THE FIRST 130 CHARS FROM THE FIFO.
 8264  032712  103133                        BCC     50$             ;GO REPORT ERROR IF BMP CODE FOUND.
 8265  032714  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5705.
 8266  032720  005701                         TST     R1              ;CHECK IF AN XON WAS FOUND.
 8267  032722  001125                        BNE     40$             ;GO REPORT ERROR IF AN XON WAS FOUND.
 8268
 8269                                      ;+
 8270                                      ; TRANSMIT A NULL CHARACTER.
 8271                                      ;-
 8272  032724  010577  147316                 MOV     R5,@CSRA        ;SELECT THE LINE READY FOR TRANSMISSION.
 8273  032730  012777  100000  147312         MOV     #100000,@TXCHA  ;TRANSMIT A NULL CHARACTER.
 8274  032736  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5706.
 8275  032742  004737  017352                JSR     PC,WAITTX       ;WAIT FOR TX TO COMPLETE.
 8276  032746  103115                        BCC     50$             ;GO REPORT ERROR IF TX DID NOT COMPLETE.
 8277                                    ;+
 8278                                    ; READ THREE CHARACTERS, TO CAUSE THE XON TO BE GENERATED.
 8279                                    ;-
 8280  032750  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5707.
 8281  032754  012700  000003                 MOV     #3,R0           ;SET THE READ COUNT TO 3.
 8282  032760  004737  015740                JSR     PC,READBX       ;READ 3 CHARACTERS FROM THE FIFO.
 8283  032764  103106                        BCC     50$             ;GO REPORT ERROR IF FIFO EMPTY.
 8284  032766  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5708.
 8285  032772  005701                         TST     R1              ;CHECK IF AN XON WAS FOUND.
 8286  032774  001100                        BNE     40$             ;GO REPORT ERROR IF AN XON WAS FOUND.
 8287                                    ;+
 8288                                    ; TRANSMIT 62 CHARACTERS TO BRACKET THE XON AND FILL THE FIFO WITH 191 CHARS.
 8289                                    ;-
 8290  032776  012700  000076                 MOV     #62.,R0         ;PASS LENGTH OF DATA PATTERN.
 8291  033002  010501                         MOV     R5,R1           ;PASS THE LINE NUMBER.
 8292  033004  005237  004054                 INC     ERRNBR          ;SET ERROR NUMBER TO 5709.
 8293  033010  004737  016570                JSR     PC,TXDATP       ;TRANSMIT DATA PATTERN.
 8294  033014  103072                        BCC     50$             ;EXIT IF ERROR FOUND DURING TX.
 8295
 8296                                      ;+
 8297                                      ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
 8298                                      ; THE FIFO.
 8299                                      ;-
```

```
8300  033016  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5710.
8301  033022  004737  017352                  JSR     PC,WAITTX       ;WAIT FOR TX TO COMPLETE.
8302  033026  103065                          BCC     50$             ;GO REPORT ERROR IF TX FAILED TO COMPLETE.
8303                                        ;+
8304                                        ; READ THE FIRST 126 CHARACTERS.
8305                                        ; READ THE NEXT 4 CHARACTERS AND CHECK IF THEY ARE IN THE FOLLOWING ORDER
8306                                        ; NULL, XOFF, XON, NULL.
8307                                        ;-
8308  033030  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5711.
8309  033034  012700  000176                  MOV     #126.,R0        ;SET UP READ COUNTER.
8310  033040  004737  015740                  JSR     PC,READBX       ;READ THE FIRST 126 CHARS.
8311  033044  103056                          BCC     50$             ;GO REPORT THE ERROR IF FIFO EMPTY.
8312  033046  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5712.
8313  033052  005701                          TST     R1              ;CHECK IF AN XON WAS FOUND.
8314  033054  001050                          BNE     40$             ;GO REPORT ERROR IF AN XON WAS FOUND.
8315  033056  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5713.
8316  033062  012701  010665                  MOV     #EM5402,R1      ;PASS THE MESSAGE TO BE REPORTED.
8317  033066  013703  002250                  MOV     RBUFA,R3        ;GET THE RECEIVER BUFFER ADDRESS.
8318  033072  011302                          MOV     (R3),R2         ;READ THE NULL CHARACTER FROM THE FIFO.
8319  033074  120227  000000                  CMPB    R2,#000         ;CHECK IF IT IS A NULL CHARACTER.
8320  033100  001036                          BNE     40$             ;GO REPORT THE ERROR IF NOT THE SAME.
8321  033102  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5714.
8322  033106  011302                          MOV     (R3),R2         ;READ THE XOFF FROM THE FIFO.
8323  033110  120227  000023                  CMPB    R2,#23          ;CHECK IF THE READ CHAR IS AN XOFF.
8324  033114  001030                          BNE     40$             ;GO REPORT THE ERROR IF NOT THE SAME.
8325  033116  011302                          MOV     (R3),R2         ;READ THE XON FROM THE FIFO.
8326  033120  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5715.
8327  033124  120227  000021                  CMPB    R2,#21          ;CHECK IF THE READ CHARACTER IS AN XON.
8328  033130  001022                          BNE     40$             ;GO REPORT THE ERROR IF NOT THE SAME.
8329  033132  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5716.
8330  033136  011302                          MOV     (R3),R2         ;READ THE NULL CHARACTER FROM THE FIFO.
8331  033140  120227  000000                  CMPB    R2,#000         ;CHECK IF IT IS A NULL CHARACTER.
8332  033144  001014                          BNE     40$             ;GO REPORT THE ERROR IF NOT THE SAME.
8333
8334                                        ;+
8335                                        ; READ THE REMAINING CHARACTERS FROM THE FIFO.
8336                                        ;-
8337  033146  012700  000075  6$:             MOV     #61.,R0         ;SET UP READ COUNTER.
8338  033152  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5717.
8339  033156  004737  015740                  JSR     PC,READBX       ;READ THE FIRST 125 CHARS.
8340  033162  103007                          BCC     50$             ;GO REPORT THE ERROR IF FIFO EMPTY.
8341  033164  005237  004054                  INC     ERRNBR          ;SET ERROR NUMBER TO 5718.
8342  033170  005701                          TST     R1              ;CHECK IF AN XON WAS FOUND.
8343  033172  001001                          BNE     40$             ;GO REPORT ERROR IF AN XON WAS FOUND.
8344  033174  000404                          BR      60$             ;EXIT THE TEST.
8345  033176                  40$:    ERROR                           ;                >>>>> ERROR <<<<<
8346  033176  104460                                                                          TRAP    C$ERROR
8347  033200  000402                          BR      60$             ;EXIT THE TEST.
8348
8349  033202  004737  016456  50$:    JSR     PC,TSABRT       ;REPORT TEST ABORTED.  ERROR # INDICATES FAULT.
8350  033206  005037  002270  60$:     CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8351
8352  033212                          ENDTST
8353  033212                                                                  L10045:
8354  033212  104401                                                                  TRAP    C$ETST
```

```
8355                                    .SBTTL  HARDWARE TEST          - DTRMCS -
8356                                    ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8357                                    ;*              - DATA TERMINAL READY MODEM CONTROL SIGNAL TEST -
8358                                    ;*
8359                                    ;*      THIS TEST VERIFIES THAT THE DTR MODEM CONTROL SIGNAL IS WORKING
8360                                    ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
8361                                    ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK SIGNALS RI
8362                                    ;*      AND DSR TO TEST THE DTR SIGNAL.  THIS TEST IS PERFORMED ON ALL
8363                                    ;*      ACTIVE LINES.
8364                                    ;*
8365                                    ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8366
8367  033214                                    BGNTST
8368  033214                                                                  T20::
8369                                    ;+
8370                                    ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
8371                                    ;-
8372  033214  032737  000002  002242            BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
8373  033222  001002                            BNE     2$
8374  033224  000137  033714                    JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
8375  033230                            2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
8376  033230  012700  000240                                                          MOV     #PRI05,R0
8377  033234  104441                                                                  TRAP    C$SPRI
8378          000024                            TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8379  033236  012737  000024  002272            MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (78)
8380  033244  012737  177777  002270            MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
8381  033252  012737  000001  004052            MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
8382  033260  012737  017171  004054            MOV     #7801.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
8383  033266  012737  011077  004056            MOV     #EM7801,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8384                                    ;+
8385                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
8386                                    ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
8387                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 7801 <<<<<.
8388                                    ;-
8389  033274  004737  014460                    JSR     PC,CLNRST       ;RESET THE DUT.
8390  033300  103402                            BCS     4$
8391  033302  000137  033714                    JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
8392                                    ;+
8393                                    ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
8394                                    ;-
8395  033306  004737  014024            4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
8396                                    ;+
8397                                    ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8398                                    ; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
8399                                    ;   A RESPONSE ON THE ASSOCIATED RI AND DSR SIGNALS.
8400                                    ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8401                                    ;-
8402  033312  005003                            CLR     R3              ;CLEAR THE LINE COUNTER.
8403  033314  010300                    6$:     MOV     R3,R0
8404  033316  006300                            ASL     R0
8405  033320  036037  002374  002240            BIT     BITTBL(R0),ACTLNS
8406  033326  001465                            BEQ     12$             ;DON'T TEST IF NOT ACTIVE LINE.
8407                                    ;+
8408                                    ; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
8409                                    ;-
8410  033330  005000                            CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
8411  033332  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8412  033336  004737  017412                  JSR     PC,WTWLNC       ;CLEAR ALL THE DUT DTR BITS.
8413  033342  012704  000074                  MOV     #60.,R4
8414  033346  004737  014574                  JSR     PC,DELAY        ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
8415                                          ;+
8416                                          ; CHECK THAT AT LEAST ONE OF ASSOCIATED DSR OR RI IS CLEAR AND RECORD STATES.
8417                                          ;-
8418  033352  116304  004012                  MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8419  033356  010477  146664                  MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8420  033362  017705  146666                  MOV     @STATA,R5       ;GET THE STATE OF THE ASSOCIATED DSR, RI BITS.
8421  033366  012700  120000                  MOV     #BIT15!BIT13,R0
8422  033372  040500                          BIC     R5,R0           ;CHECK FOR BOTH DSR AND RI SET.
8423  033374  001431                          BEQ     10$             ;GO REPORT DTR IS BAD IF BOTH ARE SET.
8424                                          ;+
8425                                          ; SET THE DTR FOR THE SELECTED LINE AND WAIT FOR EITHER DSR OR RI TO SET.
8426                                          ;-
8427  033376  010377  146644                  MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8428  033402  052777  001000  146646          BIS     #BIT9,@LNCTRA   ;SET THE SELECTED LINE DTR.
8429  033410  012701  150074                  MOV     #150074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR RI TO SET.
8430  033414  032705  100000                  BIT     #BIT15,R5       ;CHECK PREVIOUS STATE OF DSR BIT.
8431  033420  001002                          BNE     8$              ;GO USE RI IF DSR BIT WAS NOT CLEAR.
8432  033422  012701  170074                  MOV     #170074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR DSR SET.
8433  033426  013702  002254          8$:     MOV     STATA,R2        ;SPECIFY TO LOOK IN STAT REG FOR BIT TO SET.
8434  033432  010477  146610                  MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8435  033436  004737  017276                  JSR     PC,WAIBIS       ;WAIT UP TO 60 MS FOR SIGNAL TO GO SET.
8436  033442  103417                          BCS     12$             ;SELECT NEXT LINE AND LOOP IF SIGNAL IS SET.
8437  033444  017700  146604                  MOV     @STATA,R0       ;GET THE STATUS REGISTER CONTENTS.
8438  033450  042700  057777                  BIC     #57777,R0       ;REMOVE ALL BUT THE DSR AND RI BITS.
8439  033454  040500                          BIC     R5,R0           ;TEST FOR SIGNAL ONCE CLEAR, BUT NOW SET.
8440  033456  001011                          BNE     12$             ;GO LOOP IF SIGNAL HAS GONE FROM CLR TO SET.
8441  033460                          10$:    ;REPORT DTR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8442  033460  012737  017172  004054          MOV     #7802.,ERRNBR   ;SELECT THE ERROR NUMBER.
8443  033466  012737  013102  004060          MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8444  033474  012701  011132                  MOV     #EM7802,R1      ;SELECT THE ERROR MESSAGE.
8445  033500                                  ERROR
8446  033500  104460                                                                  TRAP    C$ERROR
8447  033502  005203                  12$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
8448  033504  020327  000010                  CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8449  033510  002701                          BLT     6$              ;LOOP IF NOT ALL LINES DONE.
8450                                          ;+
8451                                          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8452                                          ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
8453                                          ;   A RESPONSE ON THE ASSOCIATED RI AND DSR SIGNALS.
8454                                          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8455                                          ;-
8456  033512  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
8457  033514  010300                  14$:    MOV     R3,R0
8458  033516  006300                          ASL     R0
8459  033520  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
8460  033526  001466                          BEQ     20$             ;DON'T TEST IF NOT ACTIVE LINE.
8461                                          ;+
8462                                          ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
8463                                          ;-
8464  033530  012700  001000                  MOV     #BIT9,R0        ;SPECIFY THAT DTR BITS ARE TO BE SET.
8465  033534  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8466  033540  004737  017412                  JSR     PC,WTWLNC       ;SET ALL THE DUT DTR BITS.
```

```
8467  033544  012704  000074            MOV     #60.,R4
8468  033550  004737  014574            JSR     PC,DELAY        ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
8469                                    ;+
8470                                    ; CHECK THAT AT LEAST ONE OF ASSOCIATED DSR OR RI IS SET AND RECORD STATES.
8471                                    ;-
8472  033554  116304  004012            MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8473  033560  010477  146462            MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8474  033564  017705  146464            MOV     @STATA,R5       ;GET THE STATE OF THE ASSOCIATED DSR, RI BITS.
8475  033570  010500                    MOV     R5,R0
8476  033572  042700  057777            BIC     #57777,R0       ;CHECK FOR BOTH DSR AND RI CLEAR.
8477  033576  001431                    BEQ     18$             ;GO REPORT DTR IS BAD IF BOTH ARE CLEAR.
8478                                    ;+
8479                                    ; CLEAR THE DTR FOR THE SELECTED LINE AND WAIT FOR EITHER DSR OR RI TO CLEAR.
8480                                    ;-
8481  033600  010377  146442            MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8482  033604  042777  001000  146444    BIC     #BIT9,@LNCTRA   ;CLEAR THE SELECTED LINE DTR.
8483  033612  012701  150074            MOV     #150074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR RI TO CLEAR.
8484  033616  032705  100000            BIT     #BIT15,R5       ;CHECK PREVIOUS STATE OF DSR BIT.
8485  033622  001402                    BEQ     16$             ;GO USE RI IF DSR BIT WAS NOT SET.
8486  033624  012701  170074            MOV     #170074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR DSR CLEAR.
8487  033630  013702  002254    16$:    MOV     STATA,R2        ;SPECIFY TO LOOK IN STAT REG FOR BIT TO CLR.
8488  033634  010477  146406            MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8489  033640  004737  017222            JSR     PC,WAIBIC       ;WAIT UP TO 60 MS FOR SIGNAL TO GO CLEAR.
8490  033644  103417                    BCS     20$             ;SELECT NEXT LINE AND LOOP IF SIGNAL IS CLEAR.
8491  033646  017700  146402            MOV     @STATA,R0       ;GET THE STATUS REGISTER CONTENTS.
8492  033652  042705  057777            BIC     #57777,R5
8493  033656  040005                    BIC     R0,R5           ;TEST FOR SIGNAL ONCE SET, BUT NOW CLEAR.
8494  033660  001011                    BNE     20$             ;GO LOOP IF SIGNAL HAS GONE FROM SET TO CLR.
8495  033662                    18$:    ;REPORT DTR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8496  033662  012737  017173  004054    MOV     #7803.,ERRNBR   ;SELECT THE ERROR NUMBER.
8497  033670  012737  013102  004060    MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8498  033676  012701  011132            MOV     #EM7802,R1      ;SELECT THE ERROR MESSAGE.
8499  033702                            ERROR
8500  033702  104460                                                    TRAP    C$ERROR
8501  033704  005203            20$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
8502  033706  020327  000010            CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8503  033712  002700                    BLT     14$             ;LOOP IF NOT ALL LINES DONE.
8504
8505  033714  005037  002270    60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8506  033720                            SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
8507  033720  012700  000340                                                    MOV     #PRI07,R0
8508  033724  104441                                                           TRAP    C$SPRI
8509
8510  033726                            ENDTST
8511  033726                                                    L10046:
8512  033726  104401                                                           TRAP    C$ETST
```

```
8513                                    .SBTTL  HARDWARE TEST           - RTSMCS -
8514                                    ;++*************************************************************************
8515                                    ;*              - REQUEST TO SEND MODEM CONTROL SIGNAL TEST -
8516                                    ;*
8517                                    ;*      THIS TEST VERIFIES THAT THE RTS MODEM CONTROL SIGNAL IS WORKING
8518                                    ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
8519                                    ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK SIGNALS CTS
8520                                    ;*      AND DCD TO TEST THE RTS SIGNAL.  THIS TEST IS PERFORMED ON ALL
8521                                    ;*      ACTIVE LINES.
8522                                    ;*
8523                                    ;--*************************************************************************
8524
8525  033730                                    BGNTST
8526  033730                                                                          T21::
8527                                    ;+
8528                                    ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
8529                                    ;-
8530  033730  032737  000002  002242            BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
8531  033736  001002                            BNE     1$
8532  033740  000137  034430                    JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
8533  033744                            1$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
8534  033744  012700  000240                                                           MOV     #PRI05,R0
8535  033750  104441                                                                   TRAP    C$SPRI
8536           000025                           TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8537  033752  012737  000025  002272            MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (79)
8538  033760  012737  177777  002270            MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
8539  033766  012737  000001  004052            MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
8540  033774  012737  017335  004054            MOV     #7901.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
8541  034002  012737  011163  004056            MOV     #EM7901,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8542                                    ;+
8543                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
8544                                    ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
8545                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 7901 <<<<<.
8546                                    ;-
8547  034010  004737  014460                    JSR     PC,CLNRST       ;RESET THE DUT.
8548  034014  103402                            BCS     3$
8549  034016  000137  034430                    JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
8550                                    ;+
8551                                    ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
8552                                    ;-
8553  034022  004737  014024            3$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
8554                                    ;+
8555                                    ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8556                                    ; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
8557                                    ;   A RESPONSE ON THE ASSOCIATED CTS AND DCD SIGNALS.
8558                                    ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8559                                    ;-
8560  034026  005003                            CLR     R3              ;CLEAR THE LINE COUNTER.
8561  034030  010300                    2$:     MOV     R3,R0
8562  034032  006300                            ASL     R0
8563  034034  036037  002374  002240            BIT     BITTBL(R0),ACTLNS
8564  034042  001465                            BEQ     8$              ;DON'T TEST IF NOT ACTIVE LINE.
8565                                    ;+
8566                                    ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
8567                                    ;-
8568  034044  005000                            CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
8569  034046  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8570  034052  004737  017412                 JSR     PC,WTWLNC       ;CLEAR ALL THE DUT RTS BITS.
8571  034056  012704  000074                  MOV     #60.,R4
8572  034062  004737  014574                 JSR     PC,DELAY        ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
8573                                        ;+
8574                                        ; CHECK THAT AT LEAST ONE OF ASSOCIATED DCD OR CTS IS CLEAR AND RECORD STATES.
8575                                        ;-
8576  034066  116304  004012                  MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8577  034072  010477  146150                  MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8578  034076  017705  146152                  MOV     @STATA,R5       ;GET THE STATE OF THE ASSOCIATED DCD, CTS BITS.
8579  034102  012700  014000                  MOV     #BIT12!BIT11,R0
8580  034106  040500                          BIC     R5,R0           ;CHECK FOR BOTH DCD AND CTS SET.
8581  034110  001431                         BEQ     6$              ;GO REPORT RTS IS BAD IF BOTH ARE SET.
8582                                        ;+
8583                                        ; SET THE RTS FOR THE SELECTED LINE AND WAIT FOR EITHER DCD OR CTS TO SET.
8584                                        ;-
8585  034112  010377  146130                  MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8586  034116  052777  010000  146132          BIS     #BIT12,@LNCTRA  ;SET THE SELECTED LINE RTS.
8587  034124  012701  130074                  MOV     #130074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR CTS TO SET.
8588  034130  032705  010000                  BIT     #BIT12,R5       ;CHECK PREVIOUS STATE OF DCD BIT.
8589  034134  001002                         BNE     4$              ;GO USE CTS IF DCD BIT WAS NOT CLEAR.
8590  034136  012701  140074                  MOV     #140074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR DCD SET.
8591  034142  013702  002254          4$:     MOV     STATA,R2        ;SPECIFY TO LOOK IN STAT REG FOR BIT TO SET.
8592  034146  010477  146074                  MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
8593  034152  004737  017276                 JSR     PC,WAIBIS       ;WAIT UP TO 60 MS FOR SIGNAL TO GO SET.
8594  034156  103417                         BCS     8$              ;SELECT NEXT LINE AND LOOP IF SIGNAL IS SET.
8595  034160  017700  146070                  MOV     @STATA,R0       ;GET THE STATUS REGISTER CONTENTS.
8596  034164  042700  163777                  BIC     #163777,R0      ;REMOVE ALL BUT THE DCD AND CTS BITS.
8597  034170  040500                          BIC     R5,R0           ;TEST FOR SIGNAL ONCE CLEAR, BUT NOW SET.
8598  034172  001011                         BNE     8$              ;GO LOOP IF SIGNAL HAS GONE FROM CLR TO SET.
8599  034174                          6$:     ;REPORT RTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8600  034174  012737  017336  004054          MOV     #7902.,ERRNBR   ;SELECT THE ERROR NUMBER.
8601  034202  012737  013102  004060          MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8602  034210  012701  011216                  MOV     #EM79C2,R1      ;SELECT THE ERROR MESSAGE.
8603  034214                                  ERROR                   ;                >>>>> ERROR <<<<<.
8604  034214  104460                                                                    TRAP    C$ERROR
8605  034216  005203                  8$:     INC     R3              ;SELECT THE NEXT LINE NUMBER.
8606  034220  020327  000010                  CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8607  034224  002701                         BLT     2$              ;LOOP IF NOT ALL LINES DONE.
8608                                        ;+
8609                                        ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8610                                        ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
8611                                        ;   A RESPONSE ON THE ASSOCIATED CTS AND DCD SIGNALS.
8612                                        ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8613                                        ;-
8614  034226  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
8615  034230  010300                  10$:    MOV     R3,R0
8616  034232  006300                          ASL     R0
8617  034234  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
8618  034242  001466                         BEQ     16$             ;DON'T TEST IF NOT ACTIVE LINE.
8619                                        ;+
8620                                        ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
8621                                        ;-
8622  034244  012700  010000                  MOV     #BIT12,R0       ;SPECIFY THAT RTS BITS ARE TO BE SET.
8623  034250  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8624  034254  004737  017412                 JSR     PC,WTWLNC       ;SET ALL THE DUT RTS BITS.
```

```
 8625  034260  012704  000074                    MOV     #60.,R4
 8626  034264  004737  014574                    JSR     PC,DELAY        ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
 8627                                          ;+
 8628                                          ; CHECK THAT AT LEAST ONE OF ASSOCIATED DCD OR CTS IS SET AND RECORD STATES.
 8629                                          ;-
 8630  034270  116304  004012                    MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
 8631  034274  010477  145746                    MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
 8632  034300  017705  145750                    MOV     @STATA,R5       ;GET THE STATE OF THE ASSOCIATED DCD, CTS BITS.
 8633  034304  010500                            MOV     R5,R0
 8634  034306  042700  163777                    BIC     #163777,R0      ;CHECK FOR BOTH DCD AND CTS CLEAR.
 8635  034312  001431                            BEQ     14$             ;GO REPORT RTS IS BAD IF BOTH ARE CLEAR.
 8636                                          ;+
 8637                                          ; CLEAR THE RTS FOR THE SELECTED LINE AND WAIT FOR EITHER DCD OR CTS TO CLEAR.
 8638                                          ;-
 8639  034314  010377  145726                    MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
 8640  034320  042777  010000  145730            BIC     #BIT12,@LNCTRA  ;CLEAR THE SELECTED LINE RTS.
 8641  034326  012701  130074                    MOV     #130074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR CTS TO CLEAR.
 8642  034332  032705  010000                    BIT     #BIT12,R5       ;CHECK PREVIOUS STATE OF DCD BIT.
 8643  034336  001402                            BEQ     12$             ;GO USE CTS IF DCD BIT WAS NOT SET.
 8644  034340  012701  140074                    MOV     #140074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR DCD CLEAR.
 8645  034344  013702  002254            12$:    MOV     STATA,R2        ;SPECIFY TO LOOK IN STAT REG FOR BIT TO CLR.
 8646  034350  010477  145672                    MOV     R4,@CSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
 8647  034354  004737  017222                    JSR     PC,WAIBIC       ;WAIT UP TO 60 MS FOR SIGNAL TO GO CLEAR.
 8648  034360  103417                            BCS     16$             ;SELECT NEXT LINE AND LOOP IF SIGNAL IS CLEAR.
 8649  034362  017700  145666                    MOV     @STATA,R0       ;GET THE STATUS REGISTER CONTENTS.
 8650  034366  042705  163777                    BIC     #163777,R5
 8651  034372  040005                            BIC     R0,R5           ;TEST FOR SIGNAL ONCE SET, BUT NOW CLEAR.
 8652  034374  001011                            BNE     16$             ;GO LOOP IF SIGNAL HAS GONE FROM SET TO CLR.
 8653  034376                          14$:    ;REPORT RTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
 8654  034376  012737  017337  004054            MOV     #7903.,ERRNBR   ;SELECT THE ERROR NUMBER.
 8655  034404  012737  013102  004060            MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 8656  034412  012701  011216                    MOV     #EM7902,R1      ;SELECT THE ERROR MESSAGE.
 8657  034416                                    ERROR                   ;                  >>>>> ERROR <<<<<.
 8658  034416  104460                                                                       TRAP    C$ERROR
 8659  034420  005203                  16$:      INC     R3              ;SELECT THE NEXT LINE NUMBER.
 8660  034422  020327  000010                    CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 8661  034426  002700                            BLT     10$             ;LOOP IF NOT ALL LINES DONE.
 8662
 8663  034430  005037  002270            60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
 8664  034434                                    SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
 8665  034434  012700  000340                                                                 MOV     #PRI07,R0
 8666  034440  104441                                                                       TRAP    C$SPRI
 8667
 8668  034442                                    ENDTST
 8669  034442                                                                      L10047:
 8670  034442  104401                                                                       TRAP    C$ETST
```

```
8671                                    .SBTTL  HARDWARE TEST           - DSRMS -
8672                                    ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8673                                    ;*              - DATA SET READY MODEM SIGNAL TEST -
8674                                    ;*
8675                                    ;*      THIS TEST VERIFIES THAT THE DSR MODEM STATUS SIGNAL IS WORKING
8676                                    ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
8677                                    ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK DTR SIGNALS
8678                                    ;*      TO TEST THE DSR SIGNAL.  THIS TEST IS PERFORMED ON ALL THE ACTIVE
8679                                    ;*      LINES.
8680                                    ;*
8681                                    ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8682
8683  034444                                    BGNTST
8684  034444                                                                          T22::
8685                                    ;+
8686                                    ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
8687                                    ;-
8688  034444  032737  000002  002242            BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
8689  034452  001002                            BNE     2$
8690  034454  000137  035060                    JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
8691  034460                            2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
8692  034460  012700  000240                                                          MOV     #PRI05,R0
8693  034464  104441                                                                  TRAP    C$SPRI
8694           000026                           TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8695  034466  012737  000026  002272            MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (80)
8696  034474  012737  177777  002270            MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
8697  034502  012737  000001  004052            MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
8698  034510  012737  017501  004054            MOV     #8001.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
8699  034516  012737  011247  004056            MOV     #EM8001,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8700                                    ;+
8701                                    ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
8702                                    ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
8703                                    ; THIS SUBROUTINE REPORTS ERROR >>>>> 8001 <<<<<.
8704                                    ;-
8705  034524  004737  014460                    JSR     PC,CLNRST       ;RESET THE DUT.
8706  034530  103402                            BCS     4$
8707  034532  000137  035060                    JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
8708                                    ;+
8709                                    ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
8710                                    ;-
8711  034536  004737  014024            4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
8712                                    ;+
8713                                    ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8714                                    ; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
8715                                    ;   A RESPONSE ON THE ASSOCIATED DSR SIGNAL.
8716                                    ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8717                                    ;-
8718  034542  005003                            CLR     R3              ;CLEAR THE LINE COUNTER.
8719  034544  010300                    6$:     MOV     R3,R0
8720  034546  006300                            ASL     R0
8721  034550  036037  002374  002240            BIT     BITTBL(R0),ACTLNS
8722  034556  001450                            BEQ     10$             ;DON'T TEST IF NOT ACTIVE LINE.
8723                                    ;+
8724                                    ; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
8725                                    ;-
8726  034560  005000                            CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
 8727  034562  012705  000377                    MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 8728  034566  004737  017412                   JSR     PC,WTWLNC       ;CLEAR ALL THE DUT DTR BITS.
 8729  034572  012704  000050                    MOV     #40.,R4
 8730  034576  004737  014574                   JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 8731                                          ;+
 8732                                          ; CHECK THAT THE SPECIFIED DSR IS CLEAR.
 8733                                          ;-
 8734  034602  010377  145440                    MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 8735  034606  032777  100000  145440            BIT     #BIT15,@STATA
 8736  034614  001020                           BNE     8$              ;GO REPORT DSR IS BAD IF BIT IS NOT CLEAR.
 8737                                          ;+
 8738                                          ; SET THE DTR FOR THE ASSOCIATED LINE.
 8739                                          ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
 8740                                          ;       IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
 8741                                          ;-
 8742  034616  116304  004012                    MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
 8743  034622  010477  145420                    MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
 8744  034626  052777  001000  145422            BIS     #BIT9,@LNCTRA   ;SET THE ASSOCIATED LINE DTR.
 8745                                          ;+
 8746                                          ; CHECK THAT THE SELECTED LINE DSR IS ACTIVE.
 8747                                          ;-
 8748  034634  010377  145406                    MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 8749  034640  012701  170050                    MOV     #170050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
 8750  034644  013702  002254                    MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
 8751  034650  004737  017276                   JSR     PC,WAIBIS       ;WAIT FOR DSR TO BECOME SET OR TIMEOUT.
 8752  034654  103411                           BCS     10$             ;SKIP ERROR REPORT IF SELECTED DSR IS SET.
 8753
 8754
 8755  034656                          8$:    ;REPORT DSR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
 8756  034656  012737  017502  004054              MOV     #8002.,ERRNBR   ;SELECT THE ERROR NUMBER.
 8757  034664  012737  013102  004060              MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 8758  034672  012701  011305                      MOV     #EM8002,R1      ;SELECT THE ERROR MESSAGE.
 8759  034676                                     ERROR                                                   TRAP    C$ERROR
 8760  034676  104460
 8761  034700  005203                  10$:       INC     R3              ;SELECT THE NEXT LINE NUMBER.
 8762  034702  020327  000010                     CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 8763  034706  002716                            BLT     6$              ;LOOP IF NOT ALL LINES DONE.
 8764                                          ;+
 8765                                          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
 8766                                          ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
 8767                                          ;   A RESPONSE ON THE SELECTED DSR SIGNAL.
 8768                                          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
 8769                                          ;-
 8770  034710  005003                           CLR     R3              ;CLEAR THE LINE COUNTER.
 8771  034712  010300                  12$:     MOV     R3,R0
 8772  034714  006300                           ASL     R0
 8773  034716  036037  002374  002240           BIT     BITTBL(R0),ACTLNS
 8774  034724  001451                           BEQ     16$             ;DON'T TEST IF NOT ACTIVE LINE.
 8775                                          ;+
 8776                                          ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
 8777                                          ;-
 8778  034726  012700  001000                    MOV     #BIT9,R0        ;SPECIFY THAT DTR BITS ARE TO BE SET.
 8779  034732  012705  000377                    MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 8780  034736  004737  017412                   JSR     PC,WTWLNC       ;SET ALL THE DUT DTR BITS.
 8781  034742  012704  000050                    MOV     #40.,R4
 8782  034746  004737  014574                   JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
```

```
8783                                    ;+
8784                                    ; CHECK THAT THE SPECIFIED DSR IS SET.
8785                                    ;-
8786  034752  010377  145270                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8787  034756  032777  100000  145270        BIT     #BIT15,@STATA
8788  034764  001420                      BEQ     14$             ;GO REPORT DSR IS BAD IF BIT IS NOT SET.
8789                                    ;+
8790                                    ; CLEAR THE DTR FOR THE ASSOCIATED LINE.
8791                                    ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
8792                                    ;       IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
8793                                    ;-
8794  034766  116304  004012                MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8795  034772  010477  145250                MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
8796  034776  042777  001000  145252        BIC     #BIT9,@LNCTRA   ;CLEAR THE ASSOCIATED LINE DTR.
8797                                    ;+
8798                                    ; CHECK THAT THE SELECTED LINE DSR IS CLEAR.
8799                                    ;-
8800  035004  010377  145236                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8801  035010  012701  170050                MOV     #170050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
8802  035014  013702  002254                MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
8803  035020  004737  017222              JSR     PC,WAIBIC       ;WAIT FOR DSR TO BECOME CLEAR OR TIMEOUT.
8804  035024  103411                      BCS     16$             ;SKIP ERROR REPORT IF SELECTED DSR IS CLEAR.
8805
8806  035026                          14$:    ;REPORT DSR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8807  035026  012737  017503  004054          MOV     #8003.,ERRNBR   ;SELECT THE ERROR NUMBER.
8808  035034  012737  013102  004060          MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8809  035042  012701  011305                MOV     #EM8002,R1      ;SELECT THE ERROR MESSAGE.
8810  035046                              ERROR
8811  035046  104460                                                      TRAP    C$ERROR
8812  035050  005203                  16$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
8813  035052  020327  000010                CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8814  035056  002715                      BLT     12$             ;LOOP IF NOT ALL LINES DONE.
8815
8816  035060  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8817  035064                              SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
8218  035064  012700  000340                                                  MOV     #PRI07,R0
8819  035070  104441                                                          TRAP    C$SPRI
8820
8821  035072                              ENDTST
8822  035072                                                           L10050:
8823  035072  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - RINGI -
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*              - RING INDICATOR MODEM SIGNAL TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE RI MODEM STATUS SIGNAL IS WORKING
                                ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
                                ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK DTR SIGNALS
                                ;*      TO TEST THE RI SIGNAL.  THIS TEST IS PERFORMED ON ALL THE ACTIVE
                                ;*      LINES.
                                ;*
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

035074                                  BGNTST
035074                                                                  T23::
                                ;+
                                ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
                                ;-
035074  032737  000002  002242          BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
035102  001002                          BNE     2$
035104  000137  035510                  JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
035110                          2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
035110  012700  000240                                                  MOV     #PRI05,R0
035114  104441                                                          TRAP    C$SPRI
                                        TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
000027
035116  012737  000027  002272          MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (81)
035124  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
035132  012737  000001  004052          MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
035140  012737  017645  004054          MOV     #8101.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
035146  012737  011351  004056          MOV     #EM8101,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 8101 <<<<<.
                                ;-
035154  004737  014460                  JSR     PC,CLNRST       ;RESET THE DUT.
035160  103402                          BCS     4$
035162  000137  035510                  JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
                                ;+
                                ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
                                ;-
035166  004737  014024          4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
                                ;+
                                ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
                                ; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
                                ;   A RESPONSE ON THE ASSOCIATED RI SIGNAL.
                                ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
                                ;-
035172  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
035174  010300                  6$:     MOV     R3,R0
035176  006300                          ASL     R0
035200  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
035206  001450                          BEQ     10$             ;DON'T TEST IF NOT ACTIVE LINE.
                                ;+
                                ; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
                                ;-
035210  005000                          CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
8880  035212  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8881  035216  004737  017412                  JSR     PC,WTWLNC       ;CLEAR ALL THE DUT DTR BITS.
8882  035222  012704  000050                  MOV     #40.,R4
8883  035226  004737  014574                  JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
8884                                          ;+
8885                                          ; CHECK THAT THE SPECIFIED RI IS CLEAR.
8886                                          ;-
8887  035232  010377  145010                  MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8888  035236  032777  020000  145010          BIT     #BIT13,@STATA
8889  035244  001020                          BNE     8$              ;GO REPORT RI IS BAD IF BIT IS NOT CLEAR.
8890                                          ;+
8891                                          ; SET THE DTR FOR THE ASSOCIATED LINE.
8892                                          ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
8893                                          ;       IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
8894                                          ;-
8895  035246  116304  004012                  MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8896  035252  010477  144770                  MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
8897  035256  052777  001000  144772          BIS     #BIT9,@LNCTRA   ;SET THE ASSOCIATED LINE DTR.
8898                                          ;+
8899                                          ; CHECK THAT THE SELECTED LINE RI IS ACTIVE.
8900                                          ;-
8901  035264  010377  144756                  MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8902  035270  012701  150050                  MOV     #150050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
8903  035274  013702  002254                  MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
8904  035300  004737  017276                  JSR     PC,WAIBIS       ;WAIT FOR RI TO BECOME SET OR TIMEOUT.
8905  035304  103411                          BCS     10$             ;SKIP ERROR REPORT IF SELECTED RI IS SET.
8906
8907
8908  035306                          8$:     ;REPORT RI MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8909  035306  012737  017646  004054            MOV     #8102.,ERRNBR   ;SELECT THE ERROR NUMBER.
8910  035314  012737  013102  004060            MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8911  035322  012701  011406                    MOV     #EM8102,R1      ;SELECT THE ERROR MESSAGE.
8912  035326                                    ERROR                                                   TRAP    C$ERROR
8913  035326  104460
8914  035330  005203                  10$:      INC     R3              ;SELECT THE NEXT LINE NUMBER.
8915  035332  020327  000010                    CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8916  035336  002716                          BLT     6$              ;LOOP IF NOT ALL LINES DONE.
8917                                          ;+
8918                                          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8919                                          ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
8920                                          ;   A RESPONSE ON THE SELECTED RI SIGNAL.
8921                                          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8922                                          ;-
8923  035340  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
8924  035342  010300                  12$:    MOV     R3,R0
8925  035344  006300                          ASL     R0
8926  035346  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
8927  035354  001451                          BEQ     16$             ;DON'T TEST IF NOT ACTIVE LINE.
8928                                          ;+
8929                                          ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
8930                                          ;-
8931  035356  012700  001000                  MOV     #BIT9,R0        ;SPECIFY THAT DTR BITS ARE TO BE SET.
8932  035362  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8933  035366  004737  017412                  JSR     PC,WTWLNC       ;SET ALL THE DUT DTR BITS.
8934  035372  012704  000050                  MOV     #40.,R4
8935  035376  004737  014574                  JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
```

```
8936                                    ;+
8937                                    ; CHECK THAT THE SPECIFIED RI IS SET.
8938                                    ;-
8939  035402  010377  144640                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8940  035406  032777  020000  144640        BIT     #BIT13,@STATA
8941  035414  001420                      BEQ     14$             ;GO REPORT RI IS BAD IF BIT IS NOT SET.
8942                                    ;+
8943                                    ; CLEAR THE DTR FOR THE ASSOCIATED LINE.
8944                                    ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
8945                                    ;       IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
8946                                    ;-
8947  035416  116304  004012                MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
8948  035422  010477  144620                MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
8949  035426  042777  001000  144622        BIC     #BIT9,@LNCTRA   ;CLEAR THE ASSOCIATED LINE DTR.
8950                                    ;+
8951                                    ; CHECK THAT THE SELECTED LINE RI IS CLEAR.
8952                                    ;-
8953  035434  010377  144606                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
8954  035440  012701  150050                MOV     #150050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
8955  035444  013702  002254                MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
8956  035450  004737  017222              JSR     PC,WAIBIC       ;WAIT FOR RI TO BECOME CLEAR OR TIMEOUT.
8957  035454  103411                      BCS     16$             ;SKIP ERROR REPORT IF SELECTED RI IS CLEAR.
8958
8959  035456                          14$:    ;REPORT RI MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
8960  035456  012737  017647  004054        MOV     #8103.,ERRNBR   ;SELECT THE ERROR NUMBER.
8961  035464  012737  013102  004060        MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8962  035472  012701  011406                MOV     #EM8102,R1      ;SELECT THE ERROR MESSAGE.
8963  035476                              ERROR
8964  035476  104460                                                      TRAP    C$ERROR
8965  035500  005203                  16$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
8966  035502  020327  000010              CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8967  035506  002715                      BLT     12$             ;LOOP IF NOT ALL LINES DONE.
8968
8969  035510  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8970  035514                              SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
8971  035514  012700  000340                                                  MOV     #PRI07,R0
8972  035520  104441                                                          TRAP    C$SPRI
8973
8974  035522                              ENDTST
8975  035522                                                          L10051:
8976  035522  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - CTSMS -
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*              - CLEAR TO SEND MODEM SIGNAL TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE CTS MODEM STATUS SIGNAL IS WORKING
                                ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
                                ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK RTS SIGNALS
                                ;*      TO TEST THE CTS SIGNAL.  THIS TEST IS PERFORMED ON ALL THE ACTIVE
                                ;*      LINES.
                                ;*
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                        BGNTST
035524                                                                  T24::
035524
                                ;+
                                ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
                                ;-
035524  032737  000002  002242          BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
035532  001002                        BNE     2$
035534  000137  036140                JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
035540                          2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
035540  012700  000240                                                  MOV     #PRI05,R0
035544  104441                                                          TRAP    C$SPRI
        000030                          TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
035546  012737  000030  002272          MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (82)
035554  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
035562  012737  000001  004052          MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
035570  012737  020011  004054          MOV     #8201.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
035576  012737  011451  004056          MOV     #EM8201,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 8201 <<<<<.
                                ;-
035604  004737  014460                  JSR     PC,CLNRST       ;RESET THE DUT.
035610  103402                        BCS     4$
035612  000137  036140                JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
                                ;+
                                ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
                                ;-
035616  004737  014024          4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
                                ;+
                                ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
                                ; THIS LOOP CLEARS ALL THE RTS'S AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
                                ;   A RESPONSE ON THE ASSOCIATED CTS SIGNAL.
                                ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
                                ;-
035622  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
035624  010300                  6$:     MOV     R3,R0
035626  006300                          ASL     R0
035630  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
035636  001450                        BEQ     10$             ;DON'T TEST IF NOT ACTIVE LINE.
                                ;+
                                ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
                                ;-
035640  005000                          CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
 9033  035642  012705  000377                MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9034  035646  004737  017412                JSR     PC,WTWLNC       ;CLEAR ALL THE DUT RTS BITS.
 9035  035652  012704  000050                MOV     #40.,R4
 9036  035656  004737  014574                JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9037                                  ;+
 9038                                  ; CHECK THAT THE SPECIFIED CTS IS CLEAR.
 9039                                  ;-
 9040  035662  010377  144360                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 9041  035666  032777  004000  144360        BIT     #BIT11,@STATA
 9042  035674  001020                        BNE     8$              ;GO REPORT CTS IS BAD IF BIT IS NOT CLEAR.
 9043                                  ;+
 9044                                  ; SET THE RTS FOR THE ASSOCIATED LINE.
 9045                                  ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
 9046                                  ;       IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
 9047                                  ;-
 9048  035676  116304  004012                MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
 9049  035702  010477  144340                MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
 9050  035706  052777  010000  144342        BIS     #BIT12,@LNCTRA  ;SET THE ASSOCIATED LINE RTS.
 9051                                  ;+
 9052                                  ; CHECK THAT THE SELECTED LINE CTS IS ACTIVE.
 9053                                  ;-
 9054  035714  010377  144326                MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 9055  035720  012701  130050                MOV     #130050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
 9056  035724  013702  002254                MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
 9057  035730  004737  017276                JSR     PC,WAIBIS       ;WAIT FOR CTS TO BECOME SET OR TIMEOUT.
 9058  035734  103411                        BCS     10$             ;SKIP ERROR REPORT IF SELECTED CTS IS SET.
 9059
 9060
 9061  035736                        8$:     ;REPORT CTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
 9062  035736  012737  020012  004054          MOV     #8202.,ERRNBR   ;SELECT THE ERROR NUMBER.
 9063  035744  012737  013102  004060          MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 9064  035752  012701  011507                MOV     #EM8202,R1      ;SELECT THE ERROR MESSAGE.
 9065  035756                                ERROR                                                   TRAP    C$ERROR
 9066  035756  104460
 9067  035760  005203                10$:     INC     R3              ;SELECT THE NEXT LINE NUMBER.
 9068  035762  020327  000010                CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 9069  035766  002716                        BLT     6$              ;LOOP IF NOT ALL LINES DONE.
 9070                                  ;+
 9071                                  ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
 9072                                  ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
 9073                                  ;   A RESPONSE ON THE SELECTED CTS SIGNAL.
 9074                                  ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
 9075                                  ;-
 9076  035770  005003                        CLR     R3              ;CLEAR THE LINE COUNTER.
 9077  035772  010300                12$:     MOV     R3,R0
 9078  035774  006300                        ASL     R0
 9079  035776  036037  002374  002240        BIT     BITTBL(R0),ACTLNS
 9080  036004  001451                        BEQ     16$             ;DON'T TEST IF NOT ACTIVE LINE.
 9081                                  ;+
 9082                                  ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
 9083                                  ;-
 9084  036006  012700  010000                MOV     #BIT12,R0       ;SPECIFY THAT RTS BITS ARE TO BE SET.
 9085  036012  012705  000377                MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9086  036016  004737  017412                JSR     PC,WTWLNC       ;SET ALL THE DUT RTS BITS.
 9087  036022  012704  000050                MOV     #40.,R4
 9088  036026  004737  014574                JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
```

```
                                        ;+
                                        ; CHECK THAT THE SPECIFIED CTS IS SET.
                                        ;-
036032  010377  144210                          MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
036036  032777  004000  144210                  BIT     #BIT11,@STATA
036044  001420                                BEQ     14$             ;GO REPORT CTS IS BAD IF BIT IS NOT SET.
                                        ;+
                                        ; CLEAR THE RTS FOR THE ASSOCIATED LINE.
                                        ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
                                        ;       IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
                                        ;-
036046  116304  004012                          MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
036052  010477  144170                          MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
036056  042777  010000  144172                  BIC     #BIT12,@LNCTRA  ;CLEAR THE ASSOCIATED LINE RTS.
                                        ;+
                                        ; CHECK THAT THE SELECTED LINE CTS IS CLEAR.
                                        ;-
036064  010377  144156                          MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
036070  012701  130050                          MOV     #130050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
036074  013702  002254                          MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
036100  004737  017222                        JSR     PC,WAIBIC       ;WAIT FOR CTS TO BECOME CLEAR OR TIMEOUT.
036104  103411                                BCS     16$             ;SKIP ERROR REPORT IF SELECTED CTS IS CLEAR.

036106                                  14$:    ;REPORT CTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
036106  012737  020013  004054                  MOV     #8203.,ERRNBR   ;SELECT THE ERROR NUMBER.
036114  012737  013102  004060                  MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
036122  012701  011507                          MOV     #EM8202,R1      ;SELECT THE ERROR MESSAGE.
036126                                          ERROR
036126  104460                                                          TRAP    C$ERROR
036130  005203                          16$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
036132  020327  000010                          CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
036136  002715                                BLT     12$             ;LOOP IF NOT ALL LINES DONE.

036140  005037  002270                  60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
036144                                          SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
036144  012700  000340                                                  MOV     #PRI07,R0
036150  104441                                                          TRAP    C$SPRI

036152                                  ENDTST
036152                                                          L10052:
036152  104401                                                          TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST           - DCDMS -
                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                        ;*              - DATA CARRIER DETECTED MODEM SIGNAL TEST -
                        ;*
                        ;*      THIS TEST VERIFIES THAT THE DCD MODEM STATUS SIGNAL IS WORKING
                        ;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
                        ;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK RTS SIGNALS
                        ;*      TO TEST THE DCD SIGNAL.  THIS TEST IS PERFORMED ON ALL THE ACTIVE
                        ;*      LINES.
                        ;*
                        ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

036154                          BGNTST
036154                                                                  T25::
                        ;+
                        ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
                        ;-
036154  032737  000002  002242          BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
036162  001002                        BNE     2$
036164  000137  036570                JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
036170                  2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
036170  012700  000240                                                  MOV     #PRI05,R0
036174  104441                                                          TRAP    C$SPRI
                                TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
000031
036176  012737  000031  002272        MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (83)
036204  012737  177777  002270        MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
036212  012737  000001  004052        MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
036220  012737  020155  004054        MOV     #8301.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
036226  012737  011553  004056        MOV     #EM8301,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                        ;+
                        ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
                        ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
                        ; THIS SUBROUTINE REPORTS ERROR >>>>> 8301 <<<<<.
                        ;-
036234  004737  014460          JSR     PC,CLNRST       ;RESET THE DUT.
036240  103402                BCS     4$
036242  000137  036570        JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
                        ;+
                        ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
                        ;-
036246  004737  014024  4$:      JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
                        ;+
                        ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
                        ; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
                        ;   A RESPONSE ON THE ASSOCIATED DCD SIGNAL.
                        ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
                        ;-
036252  005003                  CLR     R3              ;CLEAR THE LINE COUNTER.
036254  010300          6$:     MOV     R3,R0
036256  006300                  ASL     R0
036260  036037  002374  002240        BIT     BITTBL(R0),ACTLNS
036266  001450                BEQ     10$             ;DON'T TEST IF NOT ACTIVE LINE.
                        ;+
                        ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
                        ;-
036270  005000                  CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
```

```
 9186  036272  012705  000377                 MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9187  036276  004737  017412                JSR     PC,WTWLNC       ;CLEAR ALL THE DUT RTS BITS.
 9188  036302  012704  000050                 MOV     #40.,R4
 9189  036306  004737  014574                JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9190                                      ;+
 9191                                      ; CHECK THAT THE SPECIFIED DCD IS CLEAR.
 9192                                      ;-
 9193  036312  010377  143730                 MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 9194  036316  032777  010000  143730         BIT     #BIT12,@STATA
 9195  036324  001020                        BNE     8$              ;GO REPORT DCD IS BAD IF BIT IS NOT CLEAR.
 9196                                      ;+
 9197                                      ; SET THE RTS FOR THE ASSOCIATED LINE.
 9198                                      ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
 9199                                      ;       IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
 9200                                      ;-
 9201  036326  116304  004012                 MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
 9202  036332  010477  143710                 MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
 9203  036336  052777  010000  143712         BIS     #BIT12,@LNCTRA  ;SET THE ASSOCIATED LINE RTS.
 9204                                      ;+
 9205                                      ; CHECK THAT THE SELECTED LINE DCD IS ACTIVE.
 9206                                      ;-
 9207  036344  010377  143676                 MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
 9208  036350  012701  140050                 MOV     #140050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
 9209  036354  013702  002254                 MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
 9210  036360  004737  017276                JSR     PC,WAIBIS       ;WAIT FOR DCD TO BECOME SET OR TIMEOUT.
 9211  036364  103411                        BCS     10$             ;SKIP ERROR REPORT IF SELECTED DCD IS SET.
 9212
 9213
 9214  036366                          8$:   ;REPORT DCD MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
 9215  036366  012737  020156  004054           MOV     #8302.,ERRNBR   ;SELECT THE ERROR NUMBER.
 9216  036374  012737  013102  004060           MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 9217  036402  012701  011611                   MOV     #EM8302,R1      ;SELECT THE ERROR MESSAGE.
 9218  036406                                  ERROR
 9219  036406  104460                                                                  TRAP    C$ERROR
 9220  036410  005203                  10$:      INC     R3              ;SELECT THE NEXT LINE NUMBER.
 9221  036412  020327  000010                   CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 9222  036416  002716                          BLT     6$              ;LOOP IF NOT ALL LINES DONE.
 9223                                      ;+
 9224                                      ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
 9225                                      ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
 9226                                      ;   A RESPONSE ON THE SELECTED DCD SIGNAL.
 9227                                      ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
 9228                                      ;-
 9229  036420  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
 9230  036422  010300                  12$:    MOV     R3,R0
 9231  036424  006300                          ASL     R0
 9232  036426  036037  002374  002240         BIT     BITTBL(R0),ACTLNS
 9233  036434  001451                        BEQ     16$             ;DON'T TEST IF NOT ACTIVE LINE.
 9234                                      ;+
 9235                                      ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
 9236                                      ;-
 9237  036436  012700  010000                 MOV     #BIT12,R0       ;SPECIFY THAT RTS BITS ARE TO BE SET.
 9238  036442  012705  000377                 MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9239  036446  004737  017412                JSR     PC,WTWLNC       ;SET ALL THE DUT RTS BITS.
 9240  036452  012704  000050                 MOV     #40.,R4
 9241  036456  004737  014574                JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
```

```
                                        ;+
                                        ; CHECK THAT THE SPECIFIED DCD IS SET.
                                        ;-
036462  010377  143560                          MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
036466  032777  010000  143560                  BIT     #BIT12,@STATA
036474  001420                                BEQ     14$             ;GO REPORT DCD IS BAD IF BIT IS NOT SET.
                                        ;+
                                        ; CLEAR THE RTS FOR THE ASSOCIATED LINE.
                                        ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
                                        ;       IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
                                        ;-
036476  116304  004012                          MOVB    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
036502  010477  143540                          MOV     R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
036506  042777  010000  143542                  BIC     #BIT12,@LNCTRA  ;CLEAR THE ASSOCIATED LINE RTS.
                                        ;+
                                        ; CHECK THAT THE SELECTED LINE DCD IS CLEAR.
                                        ;-
036514  010377  143526                          MOV     R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
036520  012701  140050                          MOV     #140050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
036524  013702  002254                          MOV     STATA,R2        ;PASS THE ADDRESS OF THE REGISTER TO TEST.
036530  004737  017222                        JSR     PC,WAIBIC       ;WAIT FOR DCD TO BECOME CLEAR OR TIMEOUT.
036534  103411                                BCS     16$             ;SKIP ERROR REPORT IF SELECTED DCD IS CLEAR.

036536                                  14$:    ;REPORT DCD MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
036536  012737  020157  004054                    MOV     #8303.,ERRNBR   ;SELECT THE ERROR NUMBER.
036544  012737  013102  004060                    MOV     #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
036552  012701  011611                          MOV     #EM8302,R1      ;SELECT THE ERROR MESSAGE.
036556                                          ERROR
036556  104460                                                          TRAP    C$ERROR
036560  005203                          16$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
036562  020327  000010                          CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
036566  002715                                BLT     12$             ;LOOP IF NOT ALL LINES DONE.

036570  005037  002270                  60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
036574                                          SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
036574  012700  000340                                                  MOV     #PRI07,R0
036600  104441                                                          TRAP    C$SPRI

036602                                  ENDTST
036602                                                          L10053:
036602  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - DTRINT -
                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*              - DATA TERMINAL READY SIGNAL INTERACTIONS TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE DTR SIGNAL (AND THE LOOPED BACK DSR AND
                                ;*      RI STATUS SIGNALS) DO NOT INTERACT WITH ANY OTHER MODEM STATUS SIGNALS.
                                ;*      IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED LOOPBACK IS
                                ;*      SPECIFIED.  THIS TEST IS PERFORMED ON ALL ACTIVE LINES.
                                ;*
                                ;---------------------------------------------------------------------------------------

                                        BGNTST
036604                                                                  T26::
                                ;+
                                ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
                                ;-
036604  032737  000002  002242          BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
036612  001002                        BNE     2$
036614  000137  037200                JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
036620                          2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
036620  012700  000240                                                  MOV     #PRI05,R0
036624  104441                                                          TRAP    C$SPRI
        000032                          TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
036626  012737  000032  002272          MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (84)
036634  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
036642  012737  000001  004052          MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
036650  012737  020321  004054          MOV     #8401.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
036656  012737  011655  004056          MOV     #EM8401,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 8401 <<<<<.
                                ;-
036664  004737  014460                JSR     PC,CLNRST       ;RESET THE DUT.
036670  103402                        BCS     4$
036672  000137  037200                JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
                                ;+
                                ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
                                ;-
036676  004737  014024          4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
                                ;+
                                ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
                                ; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND CHECKS
                                ;   FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED RI AND DSR SIGNALS.
                                ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
                                ;-
036702  005003                        CLR     R3              ;CLEAR THE LINE COUNTER.
036704  010300                  6$:   MOV     R3,R0
036706  006300                        ASL     R0
036710  036037  002374  002240        BIT     BITTBL(R0),ACTLNS
036716  001444                        BEQ     8$              ;DON'T TEST IF NOT ACTIVE LINE.
                                ;+
                                ; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
                                ;-
036720  005000                        CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
036722  012705  000377                MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
```

```
 9339  036726  004737  017412                 JSR     PC,WTWLNC       ;CLEAR ALL THE DUT DTR BITS.
 9340  036732  012704  000050                  MOV     #40.,R4
 9341  036736  004737  014574                 JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9342                                        ;+
 9343                                        ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
 9344                                        ;-
 9345  036742  004737  016266                 JSR     PC,SAVMST       ;SAVE THE PRESENT MODEM STATUS STATES.
 9346                                        ;+
 9347                                        ; SET THE DTR FOR THE SELECTED LINE.
 9348                                        ;-
 9349  036746  010377  143274                  MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
 9350  036752  052777  001000  143276          BIS     #BIT9,@LNCTRA   ;SET THE SELECTED LINE DTR.
 9351  036760  012704  000050                  MOV     #40.,R4
 9352  036764  004737  014574                 JSR     PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
 9353                                        ;+
 9354                                        ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
 9355                                        ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
 9356                                        ;-
 9357  036770  116301  004012                  MOVB    TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
 9358  036774  012702  120000                  MOV     #BIT15!BIT13,R2 ;IGNORE DSR AND RI ON ASSOCIATED LINE.
 9359  037000  004737  014502                 JSR     PC,CMPMST       ;COMPARE OLD AND NEW STAT CONTENTS.
 9360  037004  103411                         BCS     8$              ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
 9361                                        ;REPORT INTERACTIONS FOUND BETWEEN DTR FOR LINE NN AND THE FOLLOWING SIGNALS:
 9362  037006  012737  020322  004054          MOV     #8402.,ERRNBR   ;SELECT THE ERROR NUMBER.
 9363  037014  012737  013130  004060          MOV     #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 9364  037022  012701  011731                  MOV     #EM8402,R1      ;SELECT THE DTR ERROR MESSAGES.
 9365  037026                                 ERROR                   ;ER8401 USES R1, R2, AND R3 VALUES.
 9366  037026  104460                                                                 TRAP    C$ERROR
 9367                                        ;+
 9368                                        ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
 9369                                        ;-
 9370  037030  005203                        8$:     INC     R3              ;SELECT THE NEXT LINE NUMBER.
 9371  037032  020327  000010                 CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 9372  037036  002722                        BLT     6$              ;LOOP IF NOT ALL LINES DONE.
 9373                                        ;+
 9374                                        ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
 9375                                        ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND CHECKS
 9376                                        ;   FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED RI AND DSR SIGNALS.
 9377                                        ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
 9378                                        ;-
 9379  037040  005003                         CLR     R3              ;CLEAR THE LINE COUNTER.
 9380  037042  010300                        10$:    MOV     R3,R0
 9381  037044  006300                         ASL     R0
 9382  037046  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
 9383  037054  001445                        BEQ     12$             ;DON'T TEST IF NOT ACTIVE LINE.
 9384                                        ;+
 9385                                        ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
 9386                                        ;-
 9387  037056  012700  001000                  MOV     #BIT9,R0        ;SPECIFY THAT DTR BITS ARE TO BE SET.
 9388  037062  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9389  037066  004737  017412                 JSR     PC,WTWLNC       ;SET ALL THE DUT DTR BITS.
 9390  037072  012704  000050                  MOV     #40.,R4
 9391  037076  004737  014574                 JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9392                                        ;+
 9393                                        ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
 9394                                        ;-
```

```
9395  037102  004737  016266                  JSR     PC,SAVMST       ;SAVE THE PRESENT MODEM STATUS STATES.
9396                                          ;+
9397                                          ; CLEAR THE DTR FOR THE SELECTED LINE.
9398                                          ;-
9399  037106  010377  143134                  MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
9400  037112  042777  001000  143136          BIC     #BIT9,@LNCTRA   ;CLEAR THE SELECTED LINE DTR.
9401  037120  012704  000050                  MOV     #40.,R4
9402  037124  004737  014574                  JSR     PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
9403                                          ;+
9404                                          ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
9405                                          ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
9406                                          ;-
9407  037130  116301  004012                  MOVB    TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
9408  037134  012702  120000                  MOV     #BIT15!BIT13,R2 ;IGNORE DSR AND RI ON ASSOCIATED LINE.
9409  037140  004737  014502                  JSR     PC,CMPMST       ;COMPARE OLD AND NEW STAT CONTENTS.
9410  037144  103411                          BCS     12$             ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
9411                                          ;REPORT INTERACTIONS FOUND BETWEEN DTR FOR LINE NN AND THE FOLLOWING SIGNALS:
9412  037146  012737  020323  004054          MOV     #8403.,ERRNBR   ;SELECT THE ERROR NUMBER.
9413  037154  012737  013130  004060          MOV     #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
9414  037162  012701  011731                  MOV     #EM8402,R1      ;SELECT THE DTR ERROR MESSAGES.
9415  037166                                  ERROR                   ;ER8401 USES R1, R2, AND R3 VALUES.
9416  037166  104460                                                          TRAP    C$ERROR
9417                                          ;+
9418                                          ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
9419                                          ;-
9420  037170  005203                  12$:    INC     R3              ;SELECT THE NEXT LINE NUMBER.
9421  037172  020327  000010                  CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
9422  037176  002721                          BLT     10$             ;LOOP IF NOT ALL LINES DONE.
9423
9424  037200  005037  002270          60$:    CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
9425  037204                                  SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
9426  037204  012700  000340                                                  MOV     #PRI07,R0
9427  037210  104441                                                          TRAP    C$SPRI
9428
9429  037212                                  ENDTST
9430  037212                                                          L10054:
9431  037212  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  HARDWARE TEST           - RTSINT -
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;*              - REQUEST TO SEND SIGNAL INTERACTIONS TEST -
                                ;*
                                ;*      THIS TEST VERIFIES THAT THE RTS SIGNAL (AND THE LOOPED BACK DCD AND CTS
                                ;*      STATUS SIGNALS) DO NOT INTERACT WITH ANY OTHER MODEM STATUS SIGNALS.
                                ;*      IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED LOOPBACK IS
                                ;*      SPECIFIED.  THIS TEST IS PERFORMED ON ALL ACTIVE LINES.
                                ;*
                                ;--++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                                        BGNTST
037214                                                                  T27::
037214
                                ;+
                                ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
                                ;-
037214  032737  000002  002242          BIT     #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
037222  001002                        BNE     2$
037224  000137  037610                JMP     60$             ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
037230                          2$:     SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.
037230  012700  000240                                                  MOV     #PRI05,R0
037234  104441                                                          TRAP    C$SPRI
        000033                          TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
037236  012737  000033  002272          MOV     #TNUM,TSTNUM    ;SET UP THE TEST NUMBER.        (85)
037244  012737  177777  002270          MOV     #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
037252  012737  000001  004052          MOV     #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
037260  012737  020465  004054          MOV     #8501.,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
037266  012737  011754  004056          MOV     #EM8501,ERRMSG  ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
                                ;+
                                ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
                                ;  CLEAR TX AND  RX INTERRUPT ENABLE BITS.
                                ; THIS SUBROUTINE REPORTS ERROR >>>>> 8501 <<<<<.
                                ;-
037274  004737  014460                  JSR     PC,CLNRST       ;RESET THE DUT.
037300  103402                        BCS     4$
037302  000137  037610                JMP     60$             ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
                                ;+
                                ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
                                ;-
037306  004737  014024          4$:     JSR     PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
                                ;+
                                ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
                                ; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND CHECKS
                                ;   FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED DCD AND CTS SIGNALS.
                                ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
                                ;-
037312  005003                          CLR     R3              ;CLEAR THE LINE COUNTER.
037314  010300                  6$:     MOV     R3,R0
037316  006300                          ASL     R0
037320  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
037326  001444                        BEQ     8$              ;DON'T TEST IF NOT ACTIVE LINE.
                                ;+
                                ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
                                ;-
037330  005000                          CLR     R0              ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
037332  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
```

```
 9488  037336  004737  017412                 JSR     PC,WTWLNC       ;CLEAR ALL THE DUT RTS BITS.
 9489  037342  012704  000050                  MOV     #40.,R4
 9490  037346  004737  014574                 JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9491                                        ;+
 9492                                        ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
 9493                                        ;-
 9494  037352  004737  016266                 JSR     PC,SAVMST       ;SAVE THE PRESENT MODEM STATUS STATES.
 9495                                        ;+
 9496                                        ; SET THE RTS FOR THE SELECTED LINE.
 9497                                        ;-
 9498  037356  010377  142664                  MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
 9499  037362  052777  010000  142666          BIS     #BIT12,@LNCTRA  ;SET THE SELECTED LINE RTS.
 9500  037370  012704  000050                  MOV     #40.,R4
 9501  037374  004737  014574                 JSR     PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
 9502                                        ;+
 9503                                        ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
 9504                                        ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
 9505                                        ;-
 9506  037400  116301  004012                  MOVB    TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
 9507  037404  012702  014000                  MOV     #BIT12!BIT11,R2 ;IGNORE DCD AND CTS ON ASSOCIATED LINE.
 9508  037410  004737  014502                 JSR     PC,CMPMST       ;COMPARE OLD AND NEW STAT CONTENTS.
 9509  037414  103411                         BCS     8$              ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
 9510                                      ;REPORT INTERACTIONS FOUND BETWEEN RTS FOR LINE NN AND THE FOLLOWING SIGNALS:
 9511  037416  012737  020466  004054          MOV     #8502.,ERRNBR   ;SELECT THE ERROR NUMBER.
 9512  037424  012737  013130  004060          MOV     #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
 9513  037432  012701  012030                  MOV     #EM8502,R1      ;SELECT THE RTS ERROR MESSAGES.
 9514  037436                                 ERROR                   ;ER1901 USES R1, R2, AND R3 VALUES.
 9515  037436  104460                                                         TRAP    C$ERROR
 9516                                        ;+
 9517                                        ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
 9518                                        ;-
 9519  037440  005203                       8$:       INC     R3              ;SELECT THE NEXT LINE NUMBER.
 9520  037442  020327  000010                         CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
 9521  037446  002722                               BLT     6$              ;LOOP IF NOT ALL LINES DONE.
 9522                                        ;+
 9523                                        ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
 9524                                        ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND CHECKS
 9525                                        ;   FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED DCD AND CTS SIGNALS.
 9526                                        ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
 9527                                        ;-
 9528  037450  005003                         CLR     R3              ;CLEAR THE LINE COUNTER.
 9529  037452  010300                       10$:      MOV     R3,R0
 9530  037454  006300                         ASL     R0
 9531  037456  036037  002374  002240          BIT     BITTBL(R0),ACTLNS
 9532  037464  001445                         BEQ     12$             ;DON'T TEST IF NOT ACTIVE LINE.
 9533                                        ;+
 9534                                        ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
 9535                                        ;-
 9536  037466  012700  010000                  MOV     #BIT12,R0       ;SPECIFY THAT RTS BITS ARE TO BE SET.
 9537  037472  012705  000377                  MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
 9538  037476  004737  017412                 JSR     PC,WTWLNC       ;SET ALL THE DUT RTS BITS.
 9539  037502  012704  000050                  MOV     #40.,R4
 9540  037506  004737  014574                 JSR     PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
 9541                                        ;+
 9542                                        ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
 9543                                        ;-
```

```
9544  037512  004737  016266                  JSR     PC,SAVMST       ;SAVE THE PRESENT MODEM STATUS STATES.
9545                                          ;+
9546                                          ; CLEAR THE RTS FOR THE SELECTED LINE.
9547                                          ;-
9548  037516  010377  142524                   MOV     R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
9549  037522  042777  010000  142526           BIC     #BIT12,@LNCTRA  ;CLEAR THE SELECTED LINE RTS.
9550  037530  012704  000050                   MOV     #40.,R4
9551  037534  004737  014574                  JSR     PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
9552                                          ;+
9553                                          ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
9554                                          ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
9555                                          ;-
9556  037540  116301  004012                   MOVB    TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
9557  037544  012702  014000                   MOV     #BIT12!BIT11,R2 ;IGNORE DCD AND CTS ON ASSOCIATED LINE.
9558  037550  004737  014502                  JSR     PC,CMPMST       ;COMPARE OLD AND NEW STAT CONTENTS.
9559  037554  103411                          BCS     12$             ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
9560                                          ;REPORT INTERACTIONS FOUND BETWEEN RTS FOR LINE NN AND THE FOLLOWING SIGNALS:
9561  037556  012737  020467  004054           MOV     #8503.,ERRNBR   ;SELECT THE ERROR NUMBER.
9562  037564  012737  013130  004060           MOV     #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
9563  037572  012701  012030                   MOV     #EM8502,R1      ;SELECT THE RTS ERROR MESSAGES.
9564  037576                                  ERROR                   ;ER1901 USES R1, R2, AND R3 VALUES.
9565  037576  104460                                                          TRAP    C$ERROR
9566                                          ;+
9567                                          ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
9568                                          ;-
9569  037600  005203                  12$:     INC     R3              ;SELECT THE NEXT LINE NUMBER.
9570  037602  020327  000010                   CMP     R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
9571  037606  002721                          BLT     10$             ;LOOP IF NOT ALL LINES DONE.
9572
9573  037610  005037  002270          60$:     CLR     CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
9574  037614                                  SETPRI  #PRI07          ;DISABLE ALL INTERRUPTS.
9575  037614  012700  000340                                                  MOV     #PRI07,R0
9576  037620  104441                                                          TRAP    C$SPRI
9577
9578  037622                                   ENDTST
9579  037622                                                          L10055:
9580  037622  104401                                                          TRAP    C$ETST
```

```
                        .SBTTL  HARDWARE TEST           - REPBMP -
                        ;++ ******************************************************************************
                        ;*                      - REPORT ANY BMP CODES IN THE QUEUE -
                        ;*      THIS IS A PSEUDO-TEST USED TO REPORT ANY BMP CODES THAT WERE FOUND
                        ;*      IN THE DUT'S FIFO DURING PREVIOUS TEST, AND LOGGED IN THE BMP CODE
                        ;*      QUEUE.
                        ;*      IT IS UNLIKELY THAT RUNNING THIS PSEUDO-TEST ALONE WILL PRODUCE ANY
                        ;*      ERROR REPORTS.
                        ;*
                        ;-- ******************************************************************************
                              BGNTST
                                                                                T28::
                               TNUM == TNUM + 1         ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
037624  000034         002272  MOV    #TNUM,TSTNUM      ;SET UP THE TEST NUMBER.        (93)
037632  012737  177777 002270  MOV    #-1,CTRLCF        ;INDICATE THAT WE ARE IN A TEST.
037640  013702  002450         MOV    BMPCQP,R2         ;GET THE CONTENTS OF THE POINTER.
037644  012703  002452         MOV    #BMPCQB,R3        ;GET THE START ADDRESS OF THE QUEUE.
037650  020203                 CMP    R2,R3             ;SEE IF THE POINTER HAS MOVED FROM THE BASE.
037652  001411                 BEQ    60$               ;EXIT NO CODES IN THE QUEUE.
                        ;+
                        ; THERE IS AT LEAST ONE BMP CODE IN THE QUEUE. REPORT THE ERROR.
                        ;-
                              ;REPORT ERROR BMP CODE FOUND IN TEST NN, BMP CODE:NNNNNN''

037654  012701  012245         MOV    #EM9304,R1        ;PASS THE FIRST MESSAGE TO BE REORTED.
037660                         ERRDF  9301,EM9301,ER9301 ;              >>>>> ERROR #9301 <<<<<.
037660  104455                                                                  TRAP    C$ERDF
037662  022125                                                                  .WORD   9301
037664  012130                                                                  .WORD   EM9301
037666  013572                                                                  .WORD   ER9301

037670  012737  002452 002450         MOV    #BMPCQB,BMPCQP    ;SET POINTER BACK TO THE BEGINING OF THE QUE.

037676  005037  002270         60$:    CLR    CTRLCF            ;INDICATE THAT WE ARE NOT WITHIN A TEST.
037702                                 ENDTST
037702                                                                L10056:
037702  104401                                                                TRAP    C$ETST
```

```
                                .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.
                                ;--

037704                                  BGNHRD
037704  000027                                                  .WORD L10057-L$HARD/2
037706                                                  L$HARD::

                                ;DEVICE CSR ADDRESS QUESTION:
037706                                  GPRMA   HWPTQ1,0,0,160000,177776,YES
037706  000031                                                  .WORD   T$CODE
037710  037764                                                  .WORD   HWPTQ1
037712  160000                                                  .WORD   T$LOLIM
037714  177776                                                  .WORD   T$HILIM

                                ;DEVICE INTERRUPT VECTOR QUESTION:
037716                                  GPRMA   HWPTQ2,2,0,40,776,YES
037716  001031                                                  .WORD   T$CODE
037720  040002                                                  .WORD   HWPTQ2
037722  000040                                                  .WORD   T$LOLIM
037724  000776                                                  .WORD   T$HILIM

                                ;ACTIVE LINES BIT MAP QUESTION:
037726                                  GPRMD   HWPTQ3,4,0,MAPLNS,0,177777,YES
037726  002032                                                  .WORD   T$CODE
037730  040035                                                  .WORD   HWPTQ3
037732  000377                                                  .WORD   MAPLNS
037734  000000                                                  .WORD   T$LOLIM
037736  177777                                                  .WORD   T$HILIM

                                ;TYPE OF LOOPBACK QUESTION:
037740                                  GPRMD   HWPTQ4,6,0,377,1,3,YES
037740  003032                                                  .WORD   T$CODE
037742  040063                                                  .WORD   HWPTQ4
037744  000377                                                  .WORD   377
037746  000001                                                  .WORD   T$LOLIM
037750  000003                                                  .WORD   T$HILIM

                                ;INTERRUPT BR LEVEL QUESTION:
037752                                  GPRMD   HWPTQ5,6,0,177400,0,6,YES
037752  003032                                                  .WORD   T$CODE
037754  040215                                                  .WORD   HWPTQ5
037756  177400                                                  .WORD   177400
037760  000000                                                  .WORD   T$LOLIM
037762  000006                                                  .WORD   T$HILIM


037764                                  ENDHRD
                                                        .EVEN
```

```
 9674  037764                                                          L10057:
 9675
 9676
 9677  037764  051503  020122  042101        HWPTQ1: .ASCIZ  /CSR ADDRESS: /
 9678  037772  051104  051505  035123
 9679  040000  000040
 9680  040002  047111  042524  051122        HWPTQ2: .ASCIZ  /INTERRUPT VECTOR ADDRESS: /
 9681  040010  050125  020124  042526
 9682  040016  052103  051117  040440
 9683  040024  042104  042522  051523
 9684  040032  020072     000
 9685  040035     101  052103  053111        HWPTQ3: .ASCIZ  /ACTIVE LINE BIT MAP: /
 9686  040042  020105  044514  042516
 9687  040050  041040  052111  046440
 9688  040056  050101  020072     000
 9689  040063     124  050131  020105        HWPTQ4: .ASCII  /TYPE OF LOOPBACK (1=INTERNAL OR NONE,2=STAGGERD,/<15><12>
 9690  040070  043117  046040  047517
 9691  040076  041120  041501  020113
 9692  040104  030450  044475  052116
 9693  040112  051105  040516  020114
 9694  040120  051117  047040  047117
 9695  040126  026105  036462  052123
 9696  040134  043501  042507  042122
 9697  040142  006454     012
 9698  040145     040  020040  020040              .ASCIZ  /                3=25 PIN CONNECTOR): /
 9699  040152  020040  020040  020040
 9700  040160  020040  020040  020040
 9701  040166  031440  031075  020065
 9702  040174  044520  020116  047503
 9703  040202  047116  041505  047524
 9704  040210  024522  020072     000
 9705  040215     111  052116  051105        HWPTQ5: .ASCIZ  /INTERRUPT BR LEVEL: /
 9706  040222  052522  052120  041040
 9707  040230  020122  042514  042526
 9708  040236  035114  000040
 9709
 9710                                        .EVEN
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION

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

040242                                  BGNSFT
040242  000010                                                    .WORD L10060-L$SOFT/2
040244                                                          L$SOFT::

                                ;UNIT NUMBER PRINTOUT QUESTION:
040244                                  GPRML   SWPTQ1,0,20,YES
040244  000130                                                  .WORD   T$CODE
040246  040264                                                  .WORD   SWPTQ1
040250  000020                                                  .WORD   20
                                ;NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE QUESTION:
040252                                  GPRMD   SWPTQ2,2,D,177777,0,177777,YES
040252  001052                                                  .WORD   T$CODE
040254  040340                                                  .WORD   SWPTQ2
040256  177777                                                  .WORD   177777
040260  000000                                                  .WORD   T$LOLIM
040262  177777                                                  .WORD   T$HILIM

                                .EVEN

                                ENDSFT
040264                                                          .EVEN
040264                                                        L10060:

040264  042522  047520  052122    SWPTQ1: .ASCIZ  /REPORT UNIT NUMBER AS EACH UNIT IS TESTED: /
040272  052440  044516  020124
040300  052516  041115  051105
040306  040440  020123  040505
040314  044103  052440  044516
040322  020124  051511  052040
040330  051505  042524  035104
040336  000040
040340  052516  041115  051105    SWPTQ2: .ASCIZ  /NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE: /
040346  047440  020106  047111
040354  044504  044526  052504
040362  046101  042040  052101
040370  020101  051105  047522
040376  051522  052040  020117
040404  042522  047520  052122
040412  047440  020116  020101
040420  044514  042516  020072
040426     000
040430                            .EVEN
```

```
 9767
 9768
 9769  040430                          $PATCH::
 9770  040430  000024                          .BLKW   24
 9771
 9772
 9773
 9774
 9775
 9776  040500                          LASTAD
 9777                                                            .EVEN
 9778  040500  000000                                            .WORD   0
 9779  040502  000000                                            .WORD   0
 9780  040504                          L$LAST::
 9781  040504                          ENDMOD
 9782
 9783
 9784
 9785
 9786
 9787
 9788
 9789          000001                  .END
```

```
ACTLNS  002240 G  1379# 3524  5319* 5322* 5627  5640  5714  5727  5810  5826  5913  5929  6573
                  6679  6688  6818  6827  6970  6981  7087  7144  7155  7261  7321  7474  8405
                  8459  8563  8617  8721  8773  8874  8926  9027  9079  9180  9232  9332  9382
                  9481  9531
ADR   = 000020 G  1351#
ADRPTR  014442 G  3274#  5087
ALTFLD  013752 G  3004#  4778  4822  4862
ASLNTL  014024 G  3062#  6963  7137  8395  8553  8711  8864  9017  9170  9322  9471
ASSEMB= 000010    1073
BCOUNT  002340 G  1424#  4960  5239*
BITTBL  002374 G  1448#  3648  4652  4700  8405  8459  8563  8617  8721  8773  8874  8926  9027
                  9079  9180  9232  9332  9382  9481  9531
BIT0  = 000001 G  1324#  2437  4374  4430  5000  5001  5127  6857
BIT00 = 000001 G  1313#  1324
BIT01 = 000002 G  1312#  1323
BIT02 = 000004 G  1311#  1322
BIT03 = 000010 G  1310#  1321
BIT04 = 000020 G  1309#  1320
BIT05 = 000040 G  1308#  1319  4018  4263
BIT06 = 000100 G  1307#  1318  4123
BIT07 = 000200 G  1306#  1317
BIT08 = 000400 G  1305#  1316
BIT09 = 001000 G  1304#  1315
BIT1  = 000002 G  1323#  2445  6840  6880  6900  7352  7505  8372  8530  8688  8841  8994  9147
                  9299  9448
BIT10 = 002000 G  1303#
BIT11 = 004000 G  1302#  2764  8579  9041  9093  9507  9557
BIT12 = 010000 G  1301#  2760  8579  8586  8588  8622  8640  8642  9050  9084  9102  9194  9203
                  9237  9246  9255  9499  9507  9536  9549  9557
BIT13 = 020000 G  1300#  2756  8421  8888  8940  9358  9408
BIT14 = 040000 G  1299#  4512  5003  6519
BIT15 = 100000 G  1298#  2752  5050  5090  5126  6079  6152  6154  6395  6411  8132  8421  8430
                  8484  8735  8787  9358  9408
BIT2  = 000004 G  1322#
BIT3  = 000010 G  1321#
BIT4  = 000020 G  1320#  5353  6995  7169
BIT5  = 000040 G  1319#
BIT6  = 000100 G  1318#  5254
BIT7  = 000200 G  1317#  4393  4449  5048  6076  6484  6881
BIT8  = 000400 G  1316#
BIT9  = 001000 G  1315#  8428  8464  8482  8744  8778  8796  8897  8931  8949  9350  9387  9400
BMPCQB  002452 G  1482#  2924  5343  9597  9612
BMPCQE  002652 G  1483#  2937  4157  5344
BMPCQP  002450 G  1481#  4152  4160* 5345* 9596  9612*
BOE   = 000400 G  1355#
BRLEVL  002243 G  1381#  5324* 6465  6505
BUFBAS  002712 G  1508#  3566  3604  4343  6039  6705  6847  7363  7517  7647  7648  7765  7899
                  7920  7939  7950  8077  8098
BUFEND  003712 G  1512#  5057
BUFMID  003312 G  1510#  3570  3614  7648  7941
BUFPTR  002266 G  1400#  5055  5059* 6039*
BUF3QT  003512 G  1511#
CACHRX  017516 G  4888#  6132
CACHTX  017544 G  4919#  6049
CALMSL  014134 G  3127#  5265
CHKBMP  014360 G  3230#  3975  7680  7801  7981  8139
```

```
CKTRAP  014430 G  3270# 5257  5278  5524  5531  5542  5547
CLKBRL  002324 G  1418# 5222*
CLKCSR  002322 G  1417# 5221* 5256
CLKHRZ  002330 G  1420# 3148* 5224* 5225  5237  5260*
CLKINT  017572 G  4952# 5232
CLKVEC  002326 G  1419# 5223* 5233
CLNRST  014460 G  3309# 5618  5706  5802  5905  6566  6669  6808  6957  7131  7308  7461  7624
                  7737  7868  8045  8219  8389  8547  8705  8858  9011  9164  9316  9465
CMPMST  014502 G  3351# 9359  9409  9508  9558
CSRA    002246 G  1388# 2741* 3025* 3355* 3483* 3875  4024  4037* 4098* 4125* 4194* 4268  4383*
                  4439* 4487* 4514* 4732  5048  5124  5314* 5329  5522  5530  5539  5650* 5657
                  5738* 5745  5835* 5846  5854  5939* 5950  5958  6076  6153  6223  6444  6484
                  6575  6600  6714* 6723  6734* 6839* 6854* 6864  6874* 6888  6899* 6994* 7005*
                  7012  7023* 7049* 7055  7066* 7168* 7179* 7186  7197* 7223* 7229  7240* 7351*
                  7374  7504* 7528  7659  7776  7910  7931  7959  8087  8107  8113* 8160  8272*
                  8419* 8427* 8434* 8473* 8481* 8488* 8577* 8585* 8592* 8631* 8639* 8646* 8734*
                  8743* 8748* 8786* 8795* 8800* 8887* 8896* 8901* 8939* 8948* 8953* 9040* 9049*
                  9054* 9092* 9101* 9106* 9193* 9202* 9207* 9245* 9254* 9259* 9349* 9399* 9498*
                  9548*
CSRO  = 000000 G  1278#
CTRLCF  002270 G  1401# 5366* 5407  5504* 5579* 5588* 5608* 5674* 5696* 5772* 5792* 5875* 5895*
                  5989* 6012* 6268* 6295* 6533* 6556* 6639* 6659* 6776* 6798* 6921* 6947* 7094*
                  7121* 7268* 7298* 7423* 7451* 7587* 7615* 7699* 7727* 7830* 7859* 8010* 8036*
                  8183* 8209* 8350* 8380* 8505* 8538* 8663* 8696* 8816* 8849* 8969* 9002* 9122*
                  9155* 9275* 9307* 9424* 9456* 9573* 9595* 9614*
C$AU  = 000052    1073# 5484
C$AUTO= 000061    1073# 5394
C$BRK = 000022    1073# 3823  4963
C$BSEG= 000004    1073#
C$BSUB= 000002    1073#
C$CEFG= 000045    1073#
C$CLCK= 000062    1073# 5219
C$CLEA= 000012    1073# 5423
C$CLOS= 000035    1073#
C$CLP1= 000006    1073#
C$CVEC= 000036    1073# 6117  6120  6263  6266  6428  6431
C$DCLN= 000044    1073# 5581
C$DODU= 000051    1073# 5578
C$DRPT= 000024    1073#
C$DU  = 000053    1073# 5462
C$EDIT= 000003    1073# 1138
C$ERDF= 000055    1073# 5571  6093  6105  6167  6186  6236  6248  6449  6471  6489  6511  6524
                  9607
C$ERHR= 000056    1073#
C$ERRO= 000060    1073# 4057  4305  5665  5763  5865  5977  6632  6768  6911  7000  7029  7072
                  7174  7203  7246  7356  7410  7510  7574  7683  7693  7825  8006  8179  8346
                  8446  8500  8604  8658  8760  8811  8913  8964  9066  9117  9219  9270  9366
                  9416  9515  9565
C$ERSF= 000054    1073# 3811
C$ERSO= 000057    1073#
C$ESCA= 000010    1073#
C$ESEG= 000005    1073#
C$ESUB= 000003    1073#
C$ETST= 000001    1073# 5591  5677  5775  5878  5992  6274  6539  6642  6780  6925  7098  7272
                  7427  7591  7703  7834  8014  8187  8354  8512  8670  8823  8976  9129  9282
                  9431  9580  9617
```

```
C$EXIT= 000032          1073#   5415
C$GETB= 000026          1073#
C$GETW= 000027          1073#
C$GMAN= 000043          1073#
C$GPHR= 000042          1073#   5306
C$GPLO= 000030          1073#
C$GPRI= 000040          1073#   3477    4092    4481
C$INIT= 000011          1073#   5375
C$INLP= 000020          1073#
C$MANI= 000050          1073#
C$MEM = 000031          1073#
C$MSG = 000023          1073#   2463    2522    2563    2611    2659    2695    2792    2857    2891    2968
C$OPEN= 000034          1073#
C$PNTB= 000014          1073#   2443    2451    2517    2551    2595    2604    2640    2690    2735    2826    2886    2922
C$PNTF= 000017          1073#   3820    5362    5440
C$PNTS= 000016          1073#
C$PNTX= 000015          1073#   2457    2558    2647    2654    2784    2834    2844    2852    2948    2960    3888
C$QIO = 000377          1073#
C$RDBU= 000007          1073#
C$REFG= 000047          1073#   5189    5195    5201    5207
C$RESE= 000033          1073#   5213    5286    5410
C$REVI= 000003          1073#   1137
C$RFLA= 000021          1073#
C$RPT = 000025          1073#   5152
C$SEFG= 000046          1073#
C$SPRI= 000041          1073#   3481    3492    4096    4101    4485    4490    5242    5372    5605    5693    5789    5892
                        6009    6056    6114    6146    6260    6271    6292    6320    6380    6425    6536    6553    6656
                        6795    6944    7118    7295    7448    7612    7724    7856    8033    8206    8377    8508    8535
                        8666    8693    8819    8846    8972    8999    9125    9152    9278    9304    9427    9453    9576
C$SVEC= 000037          1073#   5235    6045    6052    6135    6142    6326    6357
C$TPRI= 000013          1073#
DELAY   014574 G        3396#   4231    4259    4736    5633    5720    5816    5919    6064    6177    6213    6348    6382
                        6727    6856    6892    7016    7059    7190    7233    7327    7378    7480    7532    7646    7663
                        7764    7780    7897    7914    7935    7963    8075    8091    8111    8164    8414    8468    8572
                        8626    8730    8782    8883    8935    9036    9088    9189    9241    9341    9352    9391    9402
                        9490    9501    9540    9551
DFPTBL  002216 G        1232#
DIAGMC= 000000          1073
DODMA   014634 G        3441#   4344    6707    6849    7365    7519    7650    7767    7901    7922    7952    8079    8100
DRADRT  002246 G        1387#
DROP    021032          5437    5444#
EDROP   021110          5442    5453#
EF.CON= 000036 G        1331#   5206
EF.NEW= 000035 G        1332#   5200
EF.PWR= 000034 G        1333#
EF.RES= 000037 G        1330#   5194
EF.STA= 000040 G        1329#   5188
EF0503  004213 G        1821#   2514    2548    2592    2637    2919
EF0505  004220 G        1822#   2555
EF1601  004273 G        1830#   2601
EF3001  004317 G        1834#   2644
EF3002  004366 G        1841#   2651
EF7801  004435 G        1848#   2687
EF8401  004473 G        1854#   2732
EF8402  004565 G        1864#   2781
EF9004  004702 G        1878#   2831    2841
```

EF9005	004732	G	1882#	2849				
EF9006	004763	G	1887#	2823	2883			
EF9019	005002	G	1890#	3885				
EF9301	005021	G	1893#	2957				
EF9302	005077	G	1901#	2945				
EM0101	015412	G	3813	3829#				
EM0102	015476	G	3817	3838#				
EM0103	005177	G	1913#	5573				
EM0525	005235	G	1919#	6245				
EM0526	005325	G	1929#	6103				
EM1601	005415	G	1939#	4052				
EM2101	005500	G	1948#	5611				
EM2102	005543	G	1954#	5662				
EM2201	005637	G	1965#	5699				
EM2202	005674	G	1970#	5748				
EM2203	005766	G	1980#	5759				
EM2301	006073	G	1992#	5795				
EM2302	006131	G	1998#	5862	5974			
EM2401	006167	G	2004#	5898				
EM2601	006223	G	2009#	6015	6095	6107		
EM2602	006253	G	2014#	6081	6483			
EM2603	006344	G	2024#	6078				
EM2604	006432	G	2033#	6075				
EM2605	006526	G	2043#	6088				
EM2606	006611	G	2052#	6169	6188	6238	6250	
EM2607	006650	G	2058#	6163	6180			
EM2608	006744	G	2068#	6165				
EM2609	007015	G	2075#	6183	6231			
EM2610	007074	G	2083#	6222	6443			
EM2611	007166	G	2093#	6225	6446			
EM3001	007251	G	2102#	6298	6451	6473	6491	6513	6526
EM3002	007302	G	2107#	6486				
EM3003	007376	G	2117#	6469				
EM3004	007452	G	2125#	6509				
EM3005	007526	G	2133#	6522				
EM3101	007620	G	2143#	6559				
EM3102	007654	G	2148#	6629				
EM4001	007715	G	2154#	6662				
EM4002	007740	G	2158#	6765				
EM4101	007774	G	2163#	6801				
EM4102	010017	G	2167#	6908				
EM4103	010053	G	2172#	6872				
EM4901	010137	G	2181#	6950				
EM4902	010171	G	2186#	6997	7026	7069	7171	7200	7243
EM5001	010223	G	2191#	7124				
EM5101	010253	G	2196#	7301				
EM5102	010301	G	2200#	7354				
EM5103	010337	G	2206#	7408	7571			
EM5201	010367	G	2211#	7454				
EM5202	010413	G	2215#	7507				
EM5301	010451	G	2221#	7618				
EM5302	010476	G	2225#	7690				
EM5303	010555	G	2233#	3236				
EM5401	010626	G	2240#	7730				
EM5402	010665	G	2246#	3979	7812	7821	8003	8176	8316
EM5501	010726	G	2252#	7862				

```
EM5601  010763 G          2257#   8039
EM5701  011031 G          2264#   8212
EM7801  011077 G          2271#   8383
EM7802  011132 G          2276#   8444    8498
EM7901  011163 G          2281#   8541
EM7902  011216 G          2286#   8602    8656
EM8001  011247 G          2291#   8699
EM8002  011305 G          2297#   8758    8809
EM8101  011351 G          2304#   8852
EM8102  011406 G          2309#   8911    8962
EM8201  011451 G          2315#   9005
EM8202  011507 G          2321#   9064    9115
EM8301  011553 G          2328#   9158
EM8302  011611 G          2334#   9217    9268
EM8401  011655 G          2341#   9310
EM8402  011731 G          2349#   9364    9414
EM8403  011735 G          2350#   2754
EM8404  011741 G          2351#   2758
EM8405  011744 G          2352#   2762
EM8406  011750 G          2353#   2766
EM8501  011754 G          2354#   9459
EM8502  012030 G          2362#   9513    9563
EM9009  012034 G          2363#   2840    2848
EM9010  012060 G          2367#   2830
EM9026  012104 G          2371#   3884
EM9301  012130 G          2375#   9609
EM9302  012150 G          2378#   2925
EM9303  012200 G          2382#   2943
EM9304  012245 G          2389#   9605
ENDETB  003712 G          1513#
ENDIT   020752            5210    5366#
ERLTBL  002712 G          1509#
ERRBLK  004060 G          1593#   3237*   4053*   4303*   5612*   5700*   5796*   5899*   6560*   6663*   6802*   6951*   7125*
                          7302*   7455*   7689*   7731*   8002*   8175*   8213*   8443*   8497*   8601*   8655*   8757*   8808*
                          8910*   8961*   9063*   9114*   9216*   9267*   9363*   9413*   9512*   9562*
ERRMSG  004056 G          1592#   2598    5611*   5699*   5795*   5898*   6015*   6298*   6559*   6662*   6801*   6950*   7124*
                          7301*   7454*   7618*   7730*   7862*   8039*   8212*   8383*   8541*   8699*   8852*   9005*   9158*
                          9310*   9459*
ERRNBR  004054 G          1591#   5610*   5620*   5698*   5729*   5754*   5794*   5825*   5843*   5852*   5857*   5897*   5928*
                          5947*   5956*   5961*   5964*   6014*   6024*   6297*   6308*   6558*   6583*   6598*   6610*   6613*
                          6617*   6661*   6690*   6704*   6713*   6720*   6733*   6744*   6750*   6800*   6829*   6838*   6846*
                          6861*   6871*   6885*   6898*   6949*   6993*   7010*   7021*   7041*   7064*   7123*   7167*   7184*
                          7195*   7215*   7238*   7300*   7339*   7350*   7362*   7372*   7384*   7406*   7453*   7492*   7503*
                          7516*   7526*   7539*   7569*   7617*   7649*   7656*   7671*   7679*   7686*   7729*   7774*   7791*
                          7800*   7809*   7816*   7861*   7891*   7908*   7918*   7929*   7948*   7957*   7971*   7980*   7999*
                          8038*   8069*   8085*   8096*   8105*   8138*   8158*   8172*   8211*   8243*   8253*   8261*   8265*
                          8274*   8280*   8284*   8292*   8300*   8308*   8312*   8315*   8321*   8326*   8329*   8338*   8341*
                          8382*   8442*   8496*   8540*   8600*   8654*   8698*   8756*   8807*   8851*   8909*   8960*   9004*
                          9062*   9113*   9157*   9215*   9266*   9309*   9362*   9412*   9458*   9511*   9561*
ERRTYP  004052 G          1590#   5609*   5697*   5793*   5896*   6013*   6296*   6557*   6660*   6799*   6948*   7122*   7299*
                          7452*   7616*   7728*   7860*   8037*   8210*   8381*   8539*   8697*   8850*   9003*   9156*   9308*
                          9457*
ERSMRF  002446 G          1477#   6299*
ER0101  012322 G          2433#   5574
ER0503  012640 G          2510#   6096    6170    6189    6239    6452    6492    6527    7731    8002    8175    8213
ER0504  012664 G          2544#   6108    6251
```

```
ER1603  012732 G        2586#  3237  4053  4303
ER3001  013012 G        2633#  6474  6514
ER7801  013102 G        2682#  8443  8497  8601  8655  8757  8808  8910  8961  9063  9114  9216  9267
ER8401  013130 G        2725#  9363  9413  9512  9562
ER9002  013376 G        2816#  7689
ER9101  013544 G        2878#  5612  5700  5796  5899  6560  6663  6802  6951  7125  7302  7455
ER9301  013572 G        2913#  9610
EVL   = 000004 G        1349#
E$END = 002100          1073#
E$LOAD= 000035          1073#  1161
FINACT  015010 G        3517#  7630  7742  7874  8052  8226
F$AU  = 000015          1073#  5474  5483
F$AUTO= 000020          1073#  5389  5393
F$BGN = 000040          1073#  1083  2433  2510  2544  2586  2633  2682  2725  2816  2878  2913  5142
                        5147   5162  5185  5389  5405  5415  5434  5458  5474  5478  5501  5590  5602
                        5676   5690  5774  5786  5877  5889  5991  6006  6273  6289  6538  6550  6641
                        6653   6779  6792  6924  6938  7097  7112  7271  7292  7426  7445  7590  7609
                        7702   7721  7833  7853  8013  8030  8186  8203  8353  8368  8511  8526  8669
                        8684   8822  8837  8975  8990  9128  9143  9281  9295  9430  9444  9579  9592
                        9616   9634  9725  9782
F$CLEA= 000007          1073#  5405  5422
F$DU  = 000016          1073#  5434  5461
F$END = 000041          1073#  1083  2464  2523  2564  2612  2660  2696  2793  2858  2892  2969  5145
                        5153   5376  5395  5415  5424  5456  5463  5476  5485  5501  5590  5592  5602
                        5676   5678  5690  5774  5776  5786  5877  5879  5889  5991  5993  6006  6273
                        6275   6289  6538  6540  6550  6641  6643  6653  6779  6781  6792  6924  6926
                        6938   7097  7099  7112  7271  7273  7292  7426  7428  7445  7590  7592  7609
                        7702   7704  7721  7833  7835  7853  8013  8015  8030  8186  8188  8203  8353
                        8355   8368  8511  8513  8526  8669  8671  8684  8822  8824  8837  8975  8977
                        8990   9128  9130  9143  9281  9283  9295  9430  9432  9444  9579  9581  9592
                        9616   9618  9675  9746  9782
F$HARD= 000004          1073#  9634  9673
F$HW  = 000013          1073#  1230  1241
F$INIT= 000006          1073#  5185  5374
F$JMP = 000050          1073#  5145  5415  5456  5476
F$MOD = 000000          1073#  1083  9782
F$MSG = 000011          1073#  2433  2462  2510  2521  2544  2562  2586  2610  2633  2658  2682  2694
                        2725   2791  2816  2856  2878  2890  2913  2967
F$PROT= 000021          1073#  5162  5169
F$PWR = 000017          1073#
F$RPT = 000012          1073#  5142  5151
F$SEG = 000003          1073#
F$SOFT= 000005          1073#  9725  9744
F$SRV = 000010          1073#
F$SUB = 000002          1073#
F$SW  = 000014          1073#  1254  1262
F$TEST= 000001          1073#  5502  5590  5603  5676  5691  5774  5787  5877  5890  5991  6007  6273
                        6290   6538  6551  6641  6654  6779  6793  6924  6939  7097  7113  7271  7293
                        7426   7446  7590  7610  7702  7722  7833  7854  8013  8031  8186  8204  8353
                        8369   8511  8527  8669  8685  8822  8838  8975  8991  9128  9144  9281  9296
                        9430   9445  9579  9593  9616
GETPRM  020532          5209   5295  5299#  5310
GPRSOB  002434 G        1468#
G$CNTO= 000200          1073#
G$DELM= 000372          1073#
G$DISP= 000003          1073#
```

```
G$EXCP= 000400          1073#
G$HILI= 000002          1073#
G$LOLI= 000001          1073#
G$NO  = 000000          1073#
G$OFFS= 000400          1073#  9639  9645  9651  9658  9665  9730  9735
G$OFSI= 000376          1073#  9639  9645  9651  9658  9665  9730  9735
G$PRMA= 000001          1073#  9639  9645
G$PRMD= 000002          1073#  9651  9658  9665  9735
G$PRML= 000000          1073#  9730
G$RADA= 000140          1073#
G$RADB= 000000          1073#
G$RADD= 000040          1073#  9735
G$RADL= 000120          1073#  9730
G$RADO= 000020          1073#  9639  9645  9651  9658  9665
G$XFER= 000004          1073#
G$YES = 000010          1073#  9639  9645  9651  9658  9665  9730  9735
HELP  = 000000             1#  1066  1073  1078  1091  1179  1218  1242  1263  1266  1294  1364  1588
                        1774   1791  1803  1804  1810  1812  2400  2970  5132  5154  5170  5376  5391
                        5395   5413  5418  5424  5435  5463  5475  9618  9637  9676  9711  9728  9748
                        9767   9772
HOE   = 100000 G        1362#
HWPTQ1  037764          9640   9677#
HWPTQ2  040002          9646   9680#
HWPTQ3  040035          9652   9685#
HWPTQ4  040063          9659   9689#
HWPTQ5  040215          9666   9705#
IBE   = 010000 G        1359#
IDU   = 000040 G        1352#
IER   = 020000 G        1360#
IESTAT  002274 G        1403#  3015  3482  3878  4064* 4097* 4098  4123* 4124* 4125  4190  4378  4434
                        4486*  4487  4512* 4513* 4514
INDATP  015070 G        3563#  6672  6811  7632
INDTPX  015120 G        3597#  7316  7469  7750  7882  8060  8233
ISR   = 000100 G        1353#
IXE   = 004000 G        1358#
I$AU  = 000041          1073#  5474#  5485#
I$AUTO= 000041          1073#  5389#  5395#
I$CLN = 000041          1073#  5405#  5415  5424#
I$DU  = 000041          1073#  5434#  5463#
I$HRD = 000041          9634#  9675#
I$INIT= 000041          1073#  5185#  5376#
I$MOD = 000041          1073#  1083#  9782#
I$MSG = 000041          1073#  2433#  2464#  2510#  2523#  2544#  2564#  2586#  2612#  2633#  2660#  2682#  2696#
                        2725#  2793#  2816#  2858#  2878#  2892#  2913#  2969#
I$PROT= 000040          1073#  5162#
I$PTAB= 000041          1073#
I$PWR = 000041          1073#
I$RPT = 000041          1073#  5142#  5153#
I$SEG = 000041          1073#  5501   5602  5690  5786  5889  6006  6289  6550  6653  6792  6938  7112
                        7292   7445   7609  7721  7853  8030  8203  8368  8526  8684  8837  8990  9143
                        9295   9444   9592
I$SETU= 000041          1073#
I$SFT = 000041          9725#  9746#
I$SRV = 000041          1073#
I$SUB = 000041          1073#  5501   5602  5690  5786  5889  6006  6289  6550  6653  6792  6938  7112
                        7292   7445   7609  7721  7853  8030  8203  8368  8526  8684  8837  8990  9143
```

```
                        9295    9444    9592
I$TST = 000041          1073#   5501#   5590#   5592#   5602#   5676#   5678#   5690#   5774#   5776#   5786#   5877#   5879#
                        5889#   5991#   5993#   6006#   6273#   6275#   6289#   6538#   6540#   6550#   6641#   6643#   6653#
                        6779#   6781#   6792#   6924#   6926#   6938#   7097#   7099#   7112#   7271#   7273#   7292#   7426#
                        7428#   7445#   7590#   7592#   7609#   7702#   7704#   7721#   7833#   7835#   7853#   8013#   8015#
                        8030#   8186#   8188#   8203#   8353#   8355#   8368#   8511#   8513#   8526#   8669#   8671#   8684#
                        8822#   8824#   8837#   8975#   8977#   8990#   9128#   9130#   9143#   9281#   9283#   9295#   9430#
                        9432#   9444#   9579#   9581#   9592#   9616#   9618#
J$JMP = 000167           1073#   5145    5456    5476
LGRP1M  002276 G         1404#   7088    7262
LGRP2M  002300 G         1405#   6982    7156
LINBIT  015170 G         3644#   4224
LNCTRA  002256 G         1392#   4771    6840    6857*   6880*   6900    6995    7169    7352    7505    8428*   8482*   8586*
                         8640*   8744*   8796*   8897*   8949*   9050*   9102*   9203*   9255*   9350*   9400*   9499*   9549*
LNCTRO= 000010 G         1283#
LOE   = 040000 G         1361#
LOPBCK  002242 G         1380#   3064    5323*   6939    7113    8372    8530    8688    8841    8994    9147    9299    9448
LOT   = 000010 G         1350#
LPCSLT= 000036 G         1640#
LPRA    002252 G         1390#   3876    4855    6585*   6591
LPRO  = 000004 G         1281#
L$ACP   002110 G         1168#
L$APT   002036 G         1126#
L$AU    021116 G         5474#
L$AUT   002070 G         1152#
L$AUTO  020766 G         1169    5389#
L$CCP   002106 G         1166#
L$CLEA  020770 G         1167    5405#
L$CO    002032 G         1122#
L$DEPO  002011 G         1104#
L$DESC  004132 G         1159    1795#
L$DESP  002076 G         1158#
L$DEVP  002060 G         1144#
L$DISP  002124 G         1129    1188#
L$DLY   002116 G         1174#
L$DTP   002040 G         1128#
L$DTYP  002034 G         1124#
L$DU    021006 G         1155    5434#
L$DUT   002072 G         1154#
L$DVTY  004122 G         1145    1786#
L$EF    002052 G         1139#
L$ENVI  002044 G         1132#
L$ERRT  004052 G         1163    1589#
L$ETP   002102 G         1162#
L$EXP1  002046 G         1134#
L$EXP4  002064 G         1148#
L$EXP5  002066 G         1150#
L$HARD  037706 G         1111    9634    9635#
L$HIME  002120 G         1176#
L$HPCP  002016 G         1110#
L$HPTP  002022 G         1114#
L$HW    002216 G         1115    1230    1231#
L$ICP   002104 G         1164#
L$INIT  020130 G         1165    5185#
L$LADP  002026 G         1118#
L$LAST  040504 G         1119    9780#
```

```
L$LOAD  002100 G          1160#
L$LUN   002074 G          1156#
L$MREV  002050 G          1136#
L$NAME  002000 G          1093#
L$PRIO  002042 G          1130#
L$PROT  020122 G          1171    5162#
L$PRT   002112 G          1170#
L$REPP  002062 G          1146#
L$REV   002010 G          1102#
L$RPT   020114 G          1147    5142#
L$SOFT  040244 G          1113    9725    9726#
L$SPC   002056 G          1142#
L$SPCP  002020 G          1112#
L$SPTP  002024 G          1116#
L$STA   002030 G          1120#
L$SW    002230 G          1117    1254    1255#
L$TEST  002114 G          1172#
L$TIML  002014 G          1108#
L$UNIT  002012 G          1106#   5301    5355
L10000  002226            1230    1241#
L10001  002234            1254    1262#
L10002  012424            2462#
L10003  012662            2521#
L10004  012730            2562#
L10005  013010            2610#
L10006  013100            2658#
L10007  013126            2694#
L10010  013374            2791#
L10011  013542            2856#
L10012  013570            2890#
L10013  013750            2967#
L10014  020120            5146    5151#
L10016  020764            5374#
L10017  020766            5393#
L10020  021004            5416    5422#
L10021  021114            5457    5461#
L10022  021122            5477    5483#
L10023  021412            5590#
L10024  021630            5676#
L10025  022072            5774#
L10026  022400            5877#
L10027  022744            5991#
L10030  023776            6273#
L10031  024666            6538#
L10032  025152            6641#
L10033  025546            6779#
L10034  026174            6924#
L10035  026742            7097#
L10036  027510            7271#
L10037  030112            7426#
L10040  030534            7590#
L10041  031032            7702#
L10042  031342            7833#
L10043  032042            8013#
L10044  032540            8186#
L10045  033212            8353#
```

```
L10046  033726            8511#
L10047  034442            8669#
L10050  035072            8822#
L10051  035522            8975#
L10052  036152            9128#
L10053  036602            9281#
L10054  037212            9430#
L10055  037622            9579#
L10056  037702            9616#
L10057  037764            9634    9674#
L10060  040264            9725    9745#
MAPLNS= 000377 G          1275#   4817    5320    5817    5920    6028    6198    6312    6333    6973    7085    7147    7259
                          8411    8465    8569    8623    8727    8779    8880    8932    9033    9085    9186    9238    9338
                          9388    9487    9537    9653
MFUNIT  004162 G          1816#   5359
MMENAB  002352 G          1433#   3444    5277*
MMPRES  002350 G          1432#   5276*   5281*
MMSR0   002346 G          1431#   5275
MSG1    012426 G          2440    2465#
MSG2    012504 G          2448    2473#
MSG3    012563 G          2454    2481#
MSLCNT  002344 G          1426#   3161    3180    3199*   3715
MSLGET  015216 G          3691#   3783    4029    4046    4654    4702
MSLOOP  015332 G          3402    3776#   6155    6158
MSTICK  002342 G          1425#   3193    5227*   5229*
NDERPT  002232 G          1259#
NEWPAS  020512            5203    5283    5288#   5302
NEWRES  020504            5197    5285#
NEWSTA  020174            5191    5211#
NUMLNS= 000010 G          1274#   2770    3073    3368    3523    4192    4263    4377    4433    5010    5559    5872    5986
                          8448    8502    8606    8660    8762    8813    8915    8966    9068    9119    9221    9272    9371
                          9421    9520    9570
OOPS    015346 G          3172    3198    3404    3807#
OPTION  002230 G          1258#   5353
O$APTS= 000000            1073#   1120
O$AU  = 000000            1073#   1152
O$BGNR= 000001            1073#   1090#   1146
O$BGNS= 000001            1073#   1090#   1112
O$DU  = 000001            1073#   1090#   1154
O$ERRT= 000001            1073#   1090#   1162
O$GNSW= 000001            1073#   1090#   1116
O$POIN= 000001            1073#   1090#   1178
O$SETU= 000000            1073#   1106    9778
PARATB  002354 G          1435#
PARATE  002374 G          1444#
PAR0A   002354 G          1436#   3456
PAR1A   002356 G          1437#
PAR2A   002360 G          1438#
PAR3A   002362 G          1439#
PAR4A   002364 G          1440#
PAR5A   002366 G          1441#
PAR6A   002370 G          1442#
PAR7A   002372 G          1443#
PASCNT  002302 G          1406#   5282*   5287*   5294*   5296*
PCSLOT= 000016 G          1641#
PNT   = 001000 G          1356#
```

CROSS REFERENCE TABLE -- USER SYMBOLS

```
PREGRT  004104 G  1764#
PREG05  004062    1745#  2435   2588   2727   2915   3005   3063   3128   3231   3271   3310   3352   3397
                  3442   3518   3564   3598   3645   3692   3777   3808   3874   3913   3965   4017   4151
                  4189   4223   4257   4301   4341   4372   4428   4540   4643   4691   4730   4767   4813
                  4851   4889   4920   4993   5047   5117
PRI    = 002000 G 1357#
PRI00  = 000000 G 1345#
PRI01  = 000040 G 1344#
PRI02  = 000100 G 1343#
PRI03  = 000140 G 1342#  6055   6145
PRI04  = 000200 G 1341#
PRI05  = 000240 G 1340#  5241   5604   5692   5788   5891   6008   6041   6048   6113   6131   6138   6259
                  6291   6552   6655   6794   6943   7117   7294   7447   7611   7723   7855   8032   8205
                  8376   8534   8692   8845   8998   9151   9303   9452
PRI06  = 000300 G 1339#  1418   5231
PRI07  = 000340 G 1131   1338#  3480   4095   4484   5371   6270   6319   6322   6353   6424   6535   8507
                  8665   8818   8971   9124   9277   9426   9575
PRTLPR  015574 G  2854   3873#
PUFIFO  015656 G  3320   3912#  6694   6833   7340   7493
RBUFA   002250 G  1389#  3915   3967   4994   5056   5333   5858   5962   6079   6611   6745   7006*  7050*
                  7180*  7224*  7386   7541   7670   7790   7972   8126   8317
RBUFO  = 000002 G 1279#
READBX  015740 G  3964#  8263   8282   8310   8339
RESETT  016022 G  3315   4016#  6021   6305
RXBDTX= 000030 G  1290#
RXBETX= 000020 G  1289#
RXBFUL= 000100 G  1291#
RXBRRT  017642 G  4992#  6354
RXIE0   016134 G  4090#  6065   6257   6532
RXIE1   016174 G  4123#  6062   6369
RXINPT  017732 G  5046#  6042
RXINTC  002304 G  1407#  4890   4894*  4995   4998*  5010   5051   5054*  6036*  6070   6127*  6243   6376*
                  6387
RXINTF  002306 G  1408#  4999   5012*  5050*  6037*  6086   6377*  6396
RXVECA  002234 G  1377#  5316*  6043   6116   6133   6262   6355   6427
R0SLOT= 000002 G  1647#  3650*  3733*
R1SLOT= 000004 G  1646#  3242*  3537*  3734*  3988*  4610*
R2SLOT= 000006 G  1645#  2750   4658*  4706*
R3SLOT= 000010 G  1644#
R4SLOT= 000012 G  1643#
R5SLOT= 000014 G  1642#  1752   3538*  4400*  4456*
SAVBMP  016220 G  3235   3932   4150#  6625   6751   7394   7549
SAVMST  016266 G  4188#  9345   9395   9494   9544
SETPAR  016332 G  4222#  8244
SFPTBL  002230 G  1256#
SKPSTS  016400 G  4038   4256#
STATA   002254 G  1391#  2742   3356   4195   8420   8433   8437   8474   8487   8491   8578   8591   8595
                  8632   8645   8649   8735   8750   8787   8802   8888   8903   8940   8955   9041   9056
                  9093   9108   9194   9209   9246   9261
STATO  = 000006 G 1282#
STGTRB  004032 G  1570#  3079
STSTB   002652 G  1487#  2740   3354   4191
STSTE   002712 G  1504#
SVCGBL= 000000    1073#  1076#  1093   1102   1104   1106   1108   1110   1112   1114   1116   1118   1120
                  1122   1124   1126   1128   1130   1132   1134   1136   1139   1142   1144   1146   1148
                  1150   1152   1154   1156   1158   1160   1162   1164   1166   1168   1170   1172   1174
```

```
                    1176  1188  1231  1232  1255  1256  1589  1786  1795  2433  2510  2544  2586
                    2633  2682  2725  2816  2878  2913  5142  5162  5185  5389  5405  5434  5474
                    9635  9726  9780#  9781
SVCINS= 000001      1073#  1094  1095  1096  1097  1098  1099  1100  1101  1103  1105  1107  1109
                    1111  1113  1115  1117  1119  1121  1123  1125  1127  1129  1131  1133  1135
                    1137  1138  1140  1141  1143  1145  1147  1149  1151  1153  1155  1157  1159
                    1161  1163  1165  1167  1169  1171  1173  1175  1177  1187  1189  1190  1191
                    1192  1193  1194  1195  1196  1197  1198  1199  1200  1201  1202  1203  1204
                    1205  1206  1207  1208  1209  1210  1211  1212  1213  1214  1215  1216  1230
                    1254  1787  1789  1796  1800  2440  2441  2442  2443  2444  2448  2449  2450
                    2451  2452  2454  2455  2456  2457  2458  2463  2513  2514  2515  2516  2517
                    2518  2522  2547  2548  2549  2550  2551  2552  2554  2555  2556  2557  2558
                    2559  2563  2591  2592  2593  2594  2595  2596  2600  2601  2602  2603  2604
                    2605  2611  2636  2637  2638  2639  2640  2641  2643  2644  2645  2646  2647
                    2648  2650  2651  2652  2653  2654  2655  2659  2685  2686  2687  2688  2689
                    2690  2691  2695  2730  2731  2732  2733  2734  2735  2736  2777  2778  2779
                    2780  2781  2782  2783  2784  2785  2792  2821  2822  2823  2824  2825  2826
                    2827  2829  2830  2831  2832  2833  2834  2835  2839  2840  2841  2842  2843
                    2844  2845  2848  2849  2850  2851  2852  2853  2857  2881  2882  2883  2884
                    2885  2886  2887  2891  2918  2919  2920  2921  2922  2923  2945  2946  2947
                    2948  2949  2954  2955  2956  2957  2958  2959  2960  2961  2968  3477  3478
                    3480  3481  3491  3492  3811  3812  3813  3814  3817  3818  3819  3820  3821
                    3823  3883  3884  3885  3886  3887  3888  3889  4057  4092  4093  4095  4096
                    4100  4101  4305  4481  4482  4484  4485  4489  4490  4963  5145  5146  5152
                    5188  5189  5191  5194  5195  5197  5200  5201  5203  5206  5207  5209  5213
                    5218  5219  5220  5231  5232  5233  5234  5235  5236  5241  5242  5286  5305
                    5306  5307  5309  5358  5359  5360  5361  5362  5363  5371  5372  5375  5394
                    5410  5415  5416  5423  5436  5437  5438  5439  5440  5441  5456  5457  5462
                    5476  5477  5484  5571  5572  5573  5574  5577  5578  5581  5591  5604  5605
                    5665  5677  5692  5693  5763  5775  5788  5789  5865  5878  5891  5892  5977
                    5992  6008  6009  6041  6042  6043  6044  6045  6046  6048  6049  6050  6051
                    6052  6053  6055  6056  6093  6094  6095  6096  6105  6106  6107  6108  6113
                    6114  6116  6117  6119  6120  6131  6132  6133  6134  6135  6136  6138  6139
                    6140  6141  6142  6143  6145  6146  6167  6168  6169  6170  6186  6187  6188
                    6189  6236  6237  6238  6239  6248  6249  6250  6251  6259  6260  6262  6263
                    6265  6266  6270  6271  6274  6291  6292  6319  6320  6322  6323  6324  6325
                    6326  6327  6353  6354  6355  6356  6357  6358  6379  6380  6424  6425  6427
                    6428  6430  6431  6449  6450  6451  6452  6471  6472  6473  6474  6489  6490
                    6491  6492  6511  6512  6513  6514  6524  6525  6526  6527  6535  6536  6539
                    6552  6553  6632  6642  6655  6656  6768  6780  6794  6795  6911  6925  6943
                    6944  7000  7029  7072  7098  7117  7118  7174  7203  7246  7272  7294  7295
                    7356  7410  7427  7447  7448  7510  7574  7591  7611  7612  7683  7693  7703
                    7723  7724  7825  7834  7855  7856  8006  8014  8032  8033  8179  8187  8205
                    8206  8346  8354  8376  8377  8446  8500  8507  8508  8512  8534  8535  8604
                    8658  8665  8666  8670  8692  8693  8760  8811  8818  8819  8823  8845  8846
                    8913  8964  8971  8972  8976  8998  8999  9066  9117  9124  9125  9129  9151
                    9152  9219  9270  9277  9278  9282  9303  9304  9366  9416  9426  9427  9431
                    9452  9453  9515  9565  9575  9576  9580  9607  9608  9609  9610  9617  9634
                    9639  9640  9641  9642  9645  9646  9647  9648  9651  9652  9653  9654  9655
                    9658  9659  9660  9661  9662  9665  9666  9667  9668  9669  9673  9725  9730
                    9731  9732  9735  9736  9737  9738  9739  9744  9777  9778  9779
SVCSUB= 000001       1073#  1075#
SVCTAG= 000001       1073#  1077#  1241  1262  2462  2521  2562  2610  2658  2694  2791  2856  2890
                    2967  5151  5374  5393  5422  5461  5483  5590  5676  5774  5877  5991  6273
                    6538  6641  6779  6924  7097  7271  7426  7590  7702  7833  8013  8186  8353
                    8511  8669  8822  8975  9128  9281  9430  9579  9616  9674  9745
```

```
SVCTST= 000001          1073#  1074#  5501   5602   5690   5786   5889   6006   6289   6550   6653   6792   6938
                        7112   7292   7445   7609   7721   7853   8030   8203   8368   8526   8684   8837   8990
                        9143   9295   9444   9592
SWPTQ1  040264          9731   9748#
SWPTQ2  040340          9736   9756#
S$LSYM= 010000          1073#  1242#  1263#  2463#  2522#  2563#  2611#  2659#  2695#  2792#  2857#  2891#  2968#
                        5152#  5375#  5394#  5423#  5462#  5484#  5591#  5677#  5775#  5878#  5992#  6274#  6539#
                        6642#  6780#  6925#  7098#  7272#  7427#  7591#  7703#  7834#  8014#  8187#  8354#  8512#
                        8670#  8823#  8976#  9129#  9282#  9431#  9580#  9617#  9675#  9746#
TIMER1  002332 G        1421#  3137*  3138   3155   4952   4954*
TIMER2  002334 G        1422#  4955   4957*
TIMER3  002336 G        1423#  4958*  4960*
TNUM   = 000034 G       5502#  5503   5606#  5607   5694#  5695   5790#  5791   5893#  5894   6010#  6011   6293#
                        6294   6554#  6555   6657#  6658   6796#  6797   6945#  6946   7119#  7120   7296#  7297
                        7449#  7450   7613#  7614   7725#  7726   7857#  7858   8034#  8035   8207#  8208   8378#
                        8379   8536#  8537   8694#  8695   8847#  8848   9000#  9001   9153#  9154   9305#  9306
                        9454#  9455   9593#  9594
TP4FLG  002316 G        1412#  3272*  3274   5090*  5253*  5272*  5521*
TP4RTN  020016 G        5087#  5249   5271   5509
TP4VEC  002314 G        1411#  5089   5248*  5258   5270*  5279   5508*  5566
TSABRT  016456 G        4300#  6775   6920   7093   7267   7420   7584   7698   7829   8009   8182   8349
TSTNUM  002272 G        1402#  4153   5503*  5607*  5695*  5791*  5894*  6011*  6294*  6555*  6658*  6797*  6946*
                        7120*  7297*  7450*  7614*  7726*  7858*  8035*  8208*  8379*  8537*  8695*  8848*  9001*
                        9154*  9306*  9455*  9594*
TXAD1A  002260 G        1393#  3488*
TXAD1O= 000012 G        1284#
TXAD2A  002262 G        1394#  3484   3489*  4375   4431   5836   5940   6715   6735   6875   6881*  7024   7067
                        7198   7241
TXAD2O= 000014 G        1285#
TXBFCA  002264 G        1395#  3487*  4271
TXBFCO= 000016 G        1286#
TXCHA   002250 G        1389#  5651*  5739*  5844*  5948*  6206   6341   8153*  8273*
TXCHRO= 000002 G        1280#
TXDATP  016570 G        4340#  8246   8293
TXDSBL  016612 G        4371#  5634   5721   5834   5921   5982   7038   7212
TXENBL  016706 G        4427#  5818   5937   6029   6313   6684   6823   6976   7086   7150   7260
TXIE0   017002 G        4479#  6256   6531
TXIE1   017042 G        4512#  6175   6370
TXINTC  002310 G        1409#  4921   4925*  5118   5122*  6038*  6100   6128*  6178   6193*  6217   6374*  6403
TXINTF  002312 G        1410#  5001   5123   5128*  6129*  6181   6194*  6229   6375*
TXINTR  020040 G        5116#  6139   6323
TXRLNB  004012 G        1544#  2748   3089   7005   7049   7179   7223   8418   8472   8576   8630   8742   8794
                        8895   8947   9048   9100   9201   9253   9357   9407   9506   9556
TXRLNE  004032 G        1561#
TXRXLB  003752 G        1520#  3070*  3080   3088
TXRXLE  004012 G        1537#  3083   3093
TXVECA  002236 G        1378#  5318*  6050   6119   6140   6265   6324   6430
T$ARGC= 000002          1094#  1095#  1096#  1097#  1098#  1099#  2440#  2444   2448#  2452   2454#  2458   2513#
                        2518   2547#  2552   2554#  2559   2591#  2596   2600#  2605   2636#  2641   2643#  2648
                        2650#  2655   2685#  2691   2730#  2736   2777#  2785   2821#  2827   2829#  2835   2839#
                        2845   2848#  2853   2881#  2887   2918#  2923   2945#  2949   2954#  2961   3817#  3821
                        3883#  3889   5358#  5363   5436#  5441
T$CODE= 001052          9639#  9645#  9651#  9658#  9665#  9730#  9735#
T$ERRN= 022125          1073#  3812#  5572#  6094#  6106#  6168#  6187#  6237#  6249#  6450#  6472#  6490#  6512#
                        6525#  9608#
T$EXCP= 000000          9639#  9643   9645#  9649   9651#  9656   9658#  9663   9665#  9670   9735#  9740
```

```
T$FLAG= 000050   5145# 5147  5415# 5456# 5458  5476# 5478
T$GMAN= 000000   1073#
T$HILI= 177777   9639# 9642  9645# 9648  9651# 9655  9658# 9662  9665# 9669  9735# 9739
T$LAST= 000001   1073# 9778#
T$LOLI= 000000   9639# 9641  9645# 9647  9651# 9654  9658# 9661  9665# 9668  9735# 9738
T$LSYM= 010000   1073# 1242  1263  2463  2522  2563  2611  2659  2695  2792  2857  2891  2968
                 5152  5375  5394  5423  5462  5484  5591  5677  5775  5878  5992  6274  6539
                 6642  6780  6925  7098  7272  7427  7591  7703  7834  8014  8187  8354  8512
                 8670  8823  8976  9129  9282  9431  9580  9617  9675  9746
T$LTNO= 000034   9781#
T$NEST= 177777   1073# 1083# 1230# 1241# 1254# 1262# 2433# 2462# 2510# 2521# 2544# 2562# 2586#
                 2610# 2633# 2658# 2682# 2694# 2725# 2791# 2816# 2856# 2878# 2890# 2913# 2967#
                 5142# 5151# 5162# 5169# 5185# 5374# 5389# 5393# 5405# 5422# 5434# 5461# 5474#
                 5483# 5502# 5590# 5603# 5676# 5691# 5774# 5787# 5877# 5890# 5991# 6007# 6273#
                 6290# 6538# 6551# 6641# 6654# 6779# 6793# 6924# 6939# 7097# 7113# 7271# 7293#
                 7426# 7446# 7590# 7610# 7702# 7722# 7833# 7854# 8013# 8031# 8186# 8204# 8353#
                 8369# 8511# 8527# 8669# 8685# 8822# 8838# 8975# 8991# 9128# 9144# 9281# 9296#
                 9430# 9445# 9579# 9593# 9616# 9634# 9673# 9725# 9744# 9782#
T$NS0 = 000000   1083# 9782
T$NS1 = 000005   1230# 1241  1254# 1262  2433# 2462  2510# 2521  2544# 2562  2586# 2610  2633#
                 2658  2682# 2694  2725# 2791  2816# 2856  2878# 2890  2913# 2967  5142# 5151
                 5162# 5169  5185# 5374  5389# 5393  5405# 5422  5434# 5461  5474# 5483  5502#
                 5590  5603# 5676  5691# 5774  5787# 5877  5890# 5991  6007# 6273  6290# 6538
                 6551# 6641  6654# 6779  6793# 6924  6939# 7097  7113# 7271  7293# 7426  7446#
                 7590  7610# 7702  7722# 7833  7854# 8013  8031# 8186  8204# 8353  8369# 8511
                 8527# 8669  8685# 8822  8838# 8975  8991# 9128  9144# 9281  9296# 9430  9445#
                 9579  9593# 9616  9634# 9673  9725# 9744
T$PTNU= 000000   1073#
T$SAVL= 177777   1073#
T$SEGL= 177777   1073#
T$SUBN= 000000   1073# 5501# 5602# 5690# 5786# 5889# 6006# 6289# 6550# 6653# 6792# 6938# 7112#
                 7292# 7445# 7609# 7721# 7853# 8030# 8203# 8368# 8526# 8684# 8837# 8990# 9143#
                 9295# 9444# 9592#
T$TAGL= 177777   1073#
T$TAGN= 010061   1073# 1230# 1254# 2433# 2510# 2544# 2586# 2633# 2682# 2725# 2816# 2878# 2913#
                 5142# 5162# 5185# 5389# 5405# 5434# 5474# 5502# 5603# 5691# 5787# 5890# 6007#
                 6290# 6551# 6654# 6793# 6939# 7113# 7293# 7446# 7610# 7722# 7854# 8031# 8204#
                 8369# 8527# 8685# 8838# 8991# 9144# 9296# 9445# 9593# 9634# 9725#
T$TEMP= 000000   1189# 1190# 1191# 1192# 1193# 1194# 1195# 1196# 1197# 1198# 1199# 1200# 1201#
                 1202# 1203# 1204# 1205# 1206# 1207# 1208# 1209# 1210# 1211# 1212# 1213# 1214#
                 1215# 1216# 1217# 1241# 1262# 2462# 2521# 2562# 2610# 2658# 2694# 2791# 2856#
                 2890# 2967# 5145# 5146  5151# 5169# 5374# 5393# 5415# 5416  5422# 5456# 5457
                 5461# 5476# 5477  5483# 5590# 5676# 5774# 5877# 5991# 6273# 6538# 6641# 6779#
                 6924# 7097# 7271# 7426# 7590# 7702# 7833# 8013# 8186# 8353# 8511# 8669# 8822#
                 8975# 9128# 9281# 9430# 9579# 9616# 9639# 9645# 9651# 9658# 9665# 9673# 9730#
                 9735# 9744# 9782#
T$TEST= 000034   1073# 5501# 5602# 5690# 5786# 5889# 6006# 6289# 6550# 6653# 6792# 6938# 7112#
                 7292# 7445# 7609# 7721# 7853# 8030# 8203# 8368# 8526# 8684# 8837# 8990# 9143#
                 9295# 9444# 9592# 9781
T$TSTM= 177777   1073# 2443  2451  2457  2463  2517  2522  2551  2558  2563  2595  2604  2611
                 2640  2647  2654  2659  2690  2695  2735  2784  2792  2826  2834  2844  2852
                 2857  2886  2891  2922  2948  2960  2968  3477  3481  3492  3811  3820  3823
                 3888  4057  4092  4096  4101  4305  4481  4485  4490  4963  5152  5189  5195
                 5201  5207  5213  5219  5235  5242  5286  5306  5362  5372  5375  5394  5410
                 5415  5423  5440  5462  5484  5571  5578  5581  5591  5605  5665  5677  5693
                 5763  5775  5789  5865  5878  5892  5977  5992  6009  6045  6052  6056  6093
```

CVDHBAO DHV-11 FUNC TST PART2 MACY11 30A(1052) 12-JUL-83 10:59 PAGE 239
CVDHBA.P11 12-JUL-83 00:39 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    6105    6114    6117    6120    6135    6142    6146    6167    6186    6236    6248    6260    6263
                    6266    6271    6274    6292    6320    6326    6357    6380    6425    6428    6431    6449    6471
                    6489    6511    6524    6536    6539    6553    6632    6642    6656    6768    6780    6795    6911
                    6925    6944    7000    7029    7072    7098    7118    7174    7203    7246    7272    7295    7356
                    7410    7427    7448    7510    7574    7591    7612    7683    7693    7703    7724    7825    7834
                    7856    8006    8014    8033    8179    8187    8206    8346    8354    8377    8446    8500    8508
                    8512    8535    8604    8658    8666    8670    8693    8760    8811    8819    8823    8846    8913
                    8964    8972    8976    8999    9066    9117    9125    9129    9152    9219    9270    9278    9282
                    9304    9366    9416    9427    9431    9453    9515    9565    9576    9580    9607    9617
T$TSTS= 000001      1073#   5502#   5603#   5691#   5787#   5890#   6007#   6290#   6551#   6654#   6793#   6939#   7113#
                    7293#   7446#   7610#   7722#   7854#   8031#   8204#   8369#   8527#   8685#   8838#   8991#   9144#
                    9296#   9445#   9593#
T$$AU = 010022      5474#   5476    5483
T$$AUT= 010017      5389#   5393
T$$CLE= 010020      5405#   5415    5422
T$$DU = 010021      5434#   5456    5461
T$$HAR= 010057      9634#   9674
T$$HW = 010000      1230#   1241
T$$INI= 010016      5185#   5374
T$$MSG= 010013      2433#   2462    2510#   2521    2544#   2562    2586#   2610    2633#   2658    2682#   2694    2725#
                    2791    2816#   2856    2878#   2890    2913#   2967
T$$PRO= 010015      5162#
T$$RPT= 010014      5142#   5145    5151
T$$SOF= 010060      9725#   9745
T$$SW = 010001      1254#   1262
T$$TES= 010056      5502#   5590    5603#   5676    5691#   5774    5787#   5877    5890#   5991    6007#   6273    6290#
                    6538    6551#   6641    6654#   6779    6793#   6924    6939#   7097    7113#   7271    7293#   7426
                    7446#   7590    7610#   7702    7722#   7833    7854#   8013    8031#   8186    8204#   8353    8369#
                    8511    8527#   8669    8685#   8822    8838#   8975    8991#   9128    9144#   9281    9296#   9430
                    9445#   9579    9593#   9616
T1      021124 G    1189    5501#
T10     025550 G    1198    6792#
T11     026176 G    1199    6938#
T12     026744 G    1200    7112#
T13     027512 G    1201    7292#
T14     030114 G    1202    7445#
T15     030536 G    1203    7609#
T16     031034 G    1204    7721#
T17     031344 G    1205    7853#
T18     032044 G    1206    8030#
T19     032542 G    1207    8203#
T2      021414 G    1190    5602#
T20     033214 G    1208    8368#
T21     033730 G    1209    8526#
T22     034444 G    1210    8684#
T23     035074 G    1211    8837#
T24     035524 G    1212    8990#
T25     036154 G    1213    9143#
T26     036604 G    1214    9295#
T27     037214 G    1215    9444#
T28     037624 G    1216    9592#
T3      021632 G    1191    5690#
T4      022074 G    1192    5786#
T5      022402 G    1193    5889#
T6      022746 G    1194    6006#
T7      024000 G    1195    6289#
```

```
T8          024670 G          1196  6550#
T9          025154 G          1197  6653#
UAM     =   000200 G          1354#
UNITN       002244 G          1382#  5289*  5300*  5301   5305   5358   5577
UNSDIV      017066 G          3196   4539#
WAIBIC      017222 G          4642#  6592   8489   8647   8803   8956   9109   9262
WAIBIS      017276 G          4690#  4733   5658   5746   5847   5855   5951   5959   6601   6724   6865   6889   7013
                              7056   7187   7230   7375   7529   7660   7777   7911   7932   7960   7973   8088   8108
                              8127   8161   8435   8593   8751   8904   9057   9210
WAITTX      017352 G          4729#  8254   8275   8301
WORD1       002320 G          1413#  5254*  5255   5273*  5274
WTWLNC      017412 G          4227   4766#  5629   5716   5812   5915   6200   6335   6681   6820   6972   7146   7323
                              7476   7642   7760   7893   8071   8412   8466   8570   8624   8728   8780   8881   8933
                              9034   9086   9187   9239   9339   9389   9488   9538
WTWLNS      017442 G          4812#  6208   6343
WTWLPR      017466 G          4229   4850#  5631   5718   5814   5917   6202   6337   6683   6822   6975   7149   7325
                              7478   7644   7762   7895   8073
X$ALWA= 000000                1073#
X$FALS= 000040                1073#
X$OFFS= 000400                1073#
X$TRUE= 000020                1073#
$PATCH      040430 G          9769#
.       =   040504            1080#  1482#  1509#  1510#  1511#  1513#  1789#  2398#  5146   5416   5452#  5457   5477
                              6940   6958   7114   7132   7869   7875   8053   9766#  9770#
```

```
BCOMPL      1#  1073#  5190  5196  5202  5308
BERROR      1#  1073#
BGNAU       1#  1073#  5473
BGNAUT      1#  1073#  5388
BGNCLN      1#  1073#  5404
BGNDU       1#  1073#  5433
BGNHRD      1#  1073#  9633
BGNHW       1#  1073#  1229
BGNINI      1#  1073#  5184
BGNMOD      1#  1073#  1082
BGNMSG      1#  1073#  2432  2509  2543  2585  2632  2681  2724  2815  2877  2912
BGNPRO      1#  1073#  5161
BGNPTA      1#  1073#
BGNRPT      1#  1073#  5141
BGNSEG      1#  1073#
BGNSET      1#  1073#
BGNSFT      1#  1073#  9724
BGNSRV      1#  1073#
BGNSUB      1#  1073#
BGNSW       1#  1073#  1253
BGNTST      1#  1073#  5500  5601  5689  5785  5888  6005  6288  6549  6652  6791  6937  7111  7291
          7444  7608  7720  7852  8029  8202  8367  8525  8683  8836  8989  9142  9294  9443  9591
BNCOMP      1#  1073#  5208
BNERRO      1#  1073#
BREAK       1#  1073#  3822  4962
BRESET      1#  1073#  5212  5285  5409
CKLOOP      1#  1073#
CLOCK       1#  1073#  5217
CLOSE       1#  1073#
CLRVEC      1#  1073#  6115  6118  6261  6264  6326  6429
COMMEN      1#  1073#
DELAY       1#  1073#
DESCRI      1#  1073#  1794
DEVTYP      1#  1073#  1785
DISPAT      1#  1073#  1186
DISPLA      1#  1073#
DOCLN       1#  1073#  5580
DODU        1#  1073#  5576
DORPT       1#  1073#
ENDAU       1#  1073#  5482
ENDAUT      1#  1073#  5392
ENDCLN      1#  1073#  5421
ENDCOM      1#  1073#
ENDDU       1#  1073#  5460
ENDHRD      1#  1073#  9672
ENDHW       1#  1073#  1240
ENDINI      1#  1073#  5373
ENDMOD      1#  1073#  9781
ENDMSG      1#  1073#  2461  2520  2561  2609  2657  2693  2790  2855  2889  2966
ENDPRO      1#  1073#  5168
ENDPTA      1#  1073#
ENDRPT      1#  1073#  5150
ENDSEG      1#  1073#
ENDSET      1#  1073#
ENDSFT      1#  1073#  9743
ENDSRV      1#  1073#
```

```
ENDSUB      1#   1073#
ENDSW       1#   1073#  1261
ENDTST      1#   1073#  5589  5675  5773  5876  5990  6272  6537  6640  6778  6923  7096  7270  7425
         7589    7701   7832  8012  8185  8352  8510  8668  8821  8974  9127  9280  9429  9578  9615
EQUALS      1#   1073#  1294
ERRDF       1#   1073#  5570  6092  6104  6166  6185  6235  6247  6448  6470  6488  6510  6523  9606
ERRHRD      1#   1073#
ERROR       1#   1073#  4056  4304  5664  5762  5864  5976  6631  6767  6910  6999  7028  7071  7173
         7202    7245   7355  7409  7509  7573  7682  7692  7824  8005  8178  8345  8445  8499  8603
         8657    8759   8810  8912  8963  9065  9116  9218  9269  9365  9415  9514  9564
ERRSF       1#   1073#  3810
ERRSOF      1#   1073#
ERRTBL      1#   1073#  1588
ESCAPE      1#   1073#
EXIT        1#   1073#  5144  5414  5455  5475
FEQUAL      1#   1073#
GETBYT      1#   1073#
GETPRI      1#   1073#  3476  4091  4480
GETWOR      1#   1073#
GMANIA      1#   1073#
GMANID      1#   1073#
GMANIL      1#   1073#
GPHARD      1#   1073#  5304
GPRMA       1#   1073#  9638  9644
GPRMD       1#   1073#  9650  9657  9664  9734
GPRML       1#   1073#  9729
HEADER      1#   1073#  1092
INLOOP      1#   1073#
IOSETU      1#   1073#
IOSTAR      1#   1073#
KT11        1#   1073#
LASTAD      1#   1073#  9776
MANUAL      1#   1073#
MEMORY      1#   1073#
M$BYTE      1#   1073#  1093#  1099  1100  1101
M$CHEC      1#   1073#  5145#  5415#  5456#  5476#
M$CNTO      1#   1073#  9639#  9645#  9651#  9658#  9665#  9730#  9735#
M$COUN      1#   1073#  2440#  2448#  2454#  2513#  2547#  2554#  2591#  2600#  2636#  2643#  2650#  2685#  2730#
         2777#   2821#  2829#  2839#  2848#  2881#  2918#  2945#  2954#  3817#  3883#  5358#  5436#
M$DATA      1#   1073#  1093#  1102  1104  1106  1108  1110  1112  1114  1116  1118  1120  1122  1124
         1126    1128   1130  1132#  1134  1136  1139  1142  1144  1146  1148  1150  1152  1154  1156
         1158    1160   1162  1164  1166  1168  1170  1172  1174  1176  1786#  1795#
M$DECR      1#   1073#  1241#  1262#  2462#  2521#  2562#  2610#  2658#  2694#  2791#  2856#  2890#  2967#  5151#
         5169#   5374#  5393#  5422#  5461#  5483#  5590#  5676#  5774#  5877#  5991#  6273#  6538#  6641#  6779#
         6924#   7097#  7271#  7426#  7590#  7702#  7833#  8013#  8186#  8353#  8511#  8669#  8822#  8975#  9128#
         9281#   9430#  9579#  9616#  9673#  9744#  9782#
M$DEFA      1#   1073#  9639#  9645#  9651#  9658#  9665#  9730#  9735#
M$ENDE      1#   1073#  1241#  1262#  2462#  2521#  2562#  2610#  2658#  2694#  2791#  2856#  2890#  2967#  5151#
         5374#   5393#  5422#  5461#  5483#  5590#  5676#  5774#  5877#  5991#  6273#  6538#  6641#  6779#  6924#
         7097#   7271#  7426#  7590#  7702#  7833#  8013#  8186#  8353#  8511#  8669#  8822#  8975#  9128#  9281#
         9430#   9579#  9616#  9673#  9744#  9782#
M$ERRI      1#   1073#  3811#  5571#  6093#  6105#  6167#  6186#  6236#  6248#  6449#  6471#  6489#  6511#  6524#
         9607#
M$ESCA      1#   1073#
M$ESCS      1#   1073#
M$EXCP      1#   1073#  9639#  9645#  9651#  9658#  9665#  9735#
```

```
M$EXIT      1#  1073#  5145#  5415#  5416   5456#  5476#
M$EXSE      1#  1073#  5145#  5415#  5456#  5476#
M$EXTJ      1#  1073#  5145#  5146   5415#  5456#  5457   5476#  5477
M$GEN       1#  1073#  1093#  1102#  1104#  1106#  1108#  1110#  1112#  1114#  1116#  1118#  1120#  1122#  1124#
         1126#  1128#  1130#  1132#  1134#  1136#  1139#  1142#  1144#  1146#  1148#  1150#  1152#  1154#  1156#
         1158#  1160#  1162#  1164#  1166#  1168#  1170#  1172#  1174#  1176#  1188#  1231#  1232#  1241#  1255#
         1256#  1262#  1589#  1786#  1795#  2433#  2462#  2510#  2521#  2544#  2562#  2586#  2610#  2633#  2658#
         2682#  2694#  2725#  2791#  2816#  2856#  2878#  2890#  2913#  2967#  5142#  5151#  5162#  5185#  5374#
         5389#  5393#  5405#  5422#  5434#  5461#  5474#  5483#  5501#  5590#  5602#  5676#  5690#  5774#  5786#
         5877#  5889#  5991#  6006#  6273#  6289#  6538#  6550#  6641#  6653#  6779#  6792#  6924#  6938#  7097#
         7112#  7271#  7292#  7426#  7445#  7590#  7609#  7702#  7721#  7833#  7853#  8013#  8030#  8186#  8203#
         8353#  8368#  8511#  8526#  8669#  8684#  8822#  8837#  8975#  8990#  9128#  9143#  9281#  9295#  9430#
         9444#  9579#  9592#  9616#  9635#  9674#  9726#  9745#  9780#
M$GENB      1#  1073#
M$GETS      1#  1073#  1241#  1262#  2462#  2521#  2562#  2610#  2658#  2694#  2791#  2856#  2890#  2967#  5151#
         5169#  5374#  5393#  5422#  5461#  5483#  5590#  5676#  5774#  5877#  5991#  6273#  6538#  6641#  6779#
         6924#  7097#  7271#  7426#  7590#  7702#  7833#  8013#  8186#  8353#  8511#  8669#  8822#  8975#  9128#
         9281#  9430#  9579#  9616#  9673#  9744#  9782#
M$GETT      1#  1073#  5145#  5415#  5456#  5476#
M$GNGB      1#  1073#  1083#  1093#  1102#  1104#  1106#  1108#  1110#  1112#  1114#  1116#  1118#  1120#  1122#
         1124#  1126#  1128#  1130#  1132#  1134#  1136#  1139#  1142#  1144#  1146#  1148#  1150#  1152#  1154#
         1156#  1158#  1160#  1162#  1164#  1166#  1168#  1170#  1172#  1174#  1176#  1187#  1188   1230#  1231
         1232   1254#  1255   1256   1589#  1786#  1795#  2433#  2510#  2544#  2586#  2633#  2682#  2725#  2816#
         2878#  2913#  5142#  5162#  5185#  5389#  5405#  5434#  5474#  9634#  9635   9725#  9726   9777#  9780
M$GNIN      1#  1073#  1093#  1094   1095   1096   1097   1098   1099#  1100#  1101#  1102#  1103   1104#  1105
         1106#  1107   1108#  1109   1110#  1111   1112#  1113   1114#  1115   1116#  1117   1118#  1119   1120#
         1121   1122#  1123   1124#  1125   1126#  1127   1128#  1129   1130#  1131   1132#  1133   1134#  1135
         1136#  1137   1138   1139#  1140   1141#  1142#  1143   1144#  1145   1146#  1147   1148#  1149   1150#
         1151   1152#  1153   1154#  1155   1156#  1157   1158#  1159   1160#  1161   1162#  1163   1164#  1165
         1166#  1167   1168#  1169   1170#  1171   1172#  1173   1174#  1175   1176#  1177   1187#  1189#  1190#
         1191#  1192#  1193#  1194#  1195#  1196#  1197#  1198#  1199#  1200#  1201#  1202#  1203#  1204#  1205#
         1206#  1207#  1208#  1209#  1210#  1211#  1212#  1213#  1214#  1215#  1216#  1230#  1254#  1786#  1787
         1789   1795#  1796   1800   2440#  2441#  2442   2443#  2444   2448#  2449#  2450   2451#  2452   2454#
         2455#  2456   2457#  2458   2463#  2513#  2514#  2515#  2516   2517#  2518   2522#  2547#  2548#  2549#
         2550   2551#  2552   2554#  2555#  2556#  2557   2558#  2559   2563#  2591#  2592#  2593#  2594   2595#
         2596   2600#  2601#  2602#  2603   2604#  2605   2611#  2636#  2637#  2638#  2639   2640#  2641   2643#
         2644#  2645#  2646   2647#  2648   2650#  2651#  2652#  2653   2654#  2655   2659#  2685#  2686#  2687#
         2688#  2689   2690#  2691   2695#  2730#  2731#  2732#  2733#  2734   2735#  2736   2777#  2778#  2779#
         2780#  2781#  2782#  2783   2784#  2785   2792#  2821#  2822#  2823#  2824#  2825   2826#  2827   2829#
         2830#  2831#  2832#  2833   2834#  2835   2839#  2840#  2841#  2842#  2843   2844#  2845   2848#  2849#
         2850#  2851   2852#  2853   2857#  2881#  2882#  2883#  2884#  2885   2886#  2887   2891#  2918#  2919#
         2920#  2921   2922#  2923   2945#  2946#  2947   2948#  2949   2954#  2955#  2956#  2957#  2958#  2959
         2960#  2961   2968#  3477#  3478#  3480#  3481#  3491#  3492#  3811#  3812#  3813#  3814#  3817#  3818#
         3819   3820#  3821   3823#  3883#  3884#  3885#  3886#  3887   3888#  3889   4057#  4092#  4093#  4095#
         4096#  4100#  4101#  4305#  4481#  4482#  4484#  4485#  4489#  4490#  4963#  5145#  5146#  5152#  5188#
         5189#  5191#  5194#  5195#  5197#  5200#  5201#  5203#  5206#  5207#  5209#  5213#  5218#  5219#  5220#
         5231#  5232#  5233#  5234#  5235#  5236   5241#  5242#  5286#  5305#  5306#  5307#  5309#  5358#  5359#
         5360#  5361   5362#  5363   5371#  5372#  5375#  5394#  5410#  5415#  5416#  5423#  5436#  5437#  5438#
         5439   5440#  5441   5456#  5457#  5462#  5476#  5477#  5484#  5571#  5572#  5573#  5574#  5577#  5578#
         5581#  5591#  5604#  5605#  5665#  5677#  5692#  5693#  5763#  5775#  5788#  5789#  5865#  5878#  5891#
         5892#  5977#  5992#  6008#  6009#  6041#  6042#  6043#  6044#  6045#  6046   6048#  6049#  6050#  6051#
         6052#  6053   6055#  6056#  6093#  6094#  6095#  6096#  6105#  6106#  6107#  6108#  6113#  6114#  6116#
         6117#  6119#  6120#  6131#  6132#  6133#  6134#  6135#  6136   6138#  6139#  6140#  6141#  6142#  6143
         6145#  6146#  6167#  6168#  6169#  6170#  6186#  6187#  6188#  6189#  6236#  6237#  6238#  6239#  6248#
         6249#  6250#  6251#  6259#  6260#  6262#  6263#  6265#  6266#  6270#  6271#  6274#  6291#  6292#  6319#
         6320#  6322#  6323#  6324#  6325#  6326#  6327   6353#  6354#  6355#  6356#  6357#  6358   6379#  6380#
```

```
                6424#   6425#   6427#   6428#   6430#   6431#   6449#   6450#   6451#   6452#   6471#   6472#   6473#   6474#   6489#
                6490#   6491#   6492#   6511#   6512#   6513#   6514#   6524#   6525#   6526#   6527#   6535#   6536#   6539#   6552#
                6553#   6632#   6642#   6655#   6656#   6768#   6780#   6794#   6795#   6911#   6925#   6943#   6944#   7000#   7029#
                7072#   7098#   7117#   7118#   7174#   7203#   7246#   7272#   7294#   7295#   7356#   7410#   7427#   7447#   7448#
                7510#   7574#   7591#   7611#   7612#   7683#   7693#   7703#   7723#   7724#   7825#   7834#   7855#   7856#   8006#
                8014#   8032#   8033#   8179#   8187#   8205#   8206#   8346#   8354#   8376#   8377#   8446#   8500#   8507#   8508#
                8512#   8534#   8535#   8604#   8658#   8665#   8666#   8670#   8692#   8693#   8760#   8811#   8818#   8819#   8823#
                8845#   8846#   8913#   8964#   8971#   8972#   8976#   8998#   8999#   9066#   9117#   9124#   9125#   9129#   9151#
                9152#   9219#   9270#   9277#   9278#   9282#   9303#   9304#   9366#   9416#   9426#   9427#   9431#   9452#   9453#
                9515#   9565#   9575#   9576#   9580#   9607#   9608#   9609#   9610#   9617#   9634#   9639#   9640    9641    9642
                9645#   9646    9647    9648    9651#   9652    9653    9654    9655    9658#   9659    9660    9661    9662    9665#
                9666    9667    9668    9669    9673#   9725#   9730#   9731    9732    9735#   9736    9737    9738    9739    9744#
                9777#   9778#   9779#
M$GNLS             1#   1073#
M$GNSU             1#   1073#
M$GNTA             1#   1073#   1241#   1262#   2462#   2521#   2562#   2610#   2658#   2694#   2791#   2856#   2890#   2967#   5151#
                5374#   5393#   5422#   5461#   5483#   5590#   5676#   5774#   5877#   5991#   6273#   6538#   6641#   6779#   6924#
                7097#   7271#   7426#   7590#   7702#   7833#   8013#   8186#   8353#   8511#   8669#   8822#   8975#   9128#   9281#
                9430#   9579#   9616#   9673#   9674    9744#   9745
M$GNTE             1#   1073#   5501#   5602#   5690#   5786#   5889#   6006#   6289#   6550#   6653#   6792#   6938#   7112#   7292#
                7445#   7609#   7721#   7853#   8030#   8203#   8368#   8526#   8684#   8837#   8990#   9143#   9295#   9444#   9592#
M$HAPT             1#   1073#   1093#
M$HNAP             1#   1073#   1093#   1132
M$INCR             1#   1073#   1083#   1230#   1254#   2433#   2443#   2451#   2457#   2463#   2510#   2517#   2522#   2544#   2551#
                2558#   2563#   2586#   2595#   2604#   2611#   2633#   2640#   2647#   2654#   2659#   2682#   2690#   2695#   2725#
                2735#   2784#   2792#   2816#   2826#   2834#   2844#   2852#   2857#   2878#   2886#   2891#   2913#   2922#   2948#
                2960#   2968#   3477#   3481#   3492#   3811#   3820#   3823#   3888#   4057#   4092#   4096#   4101#   4305#   4481#
                4485#   4490#   4963#   5142#   5152#   5162#   5185#   5189#   5195#   5201#   5207#   5213#   5219#   5235#   5242#
                5286#   5306#   5362#   5372#   5375#   5389#   5394#   5405#   5410#   5415#   5423#   5434#   5440#   5462#   5474#
                5484#   5501#   5502#   5571#   5578#   5581#   5591#   5602#   5603#   5605#   5665#   5677#   5690#   5691#   5693#
                5763#   5775#   5786#   5787#   5789#   5865#   5878#   5889#   5890#   5892#   5977#   5992#   6006#   6007#   6009#
                6045#   6052#   6056#   6093#   6105#   6114#   6117#   6120#   6135#   6142#   6146#   6167#   6186#   6236#   6248#
                6260#   6263#   6266#   6271#   6274#   6289#   6290#   6292#   6320#   6326#   6357#   6380#   6425#   6428#   6431#
                6449#   6471#   6489#   6511#   6524#   6536#   6539#   6550#   6551#   6553#   6632#   6642#   6653#   6654#   6656#
                6768#   6780#   6792#   6793#   6795#   6911#   6925#   6938#   6939#   6944#   7000#   7029#   7072#   7098#   7112#
                7113#   7118#   7174#   7203#   7246#   7272#   7292#   7293#   7295#   7356#   7410#   7427#   7445#   7446#   7448#
                7510#   7574#   7591#   7609#   7610#   7612#   7683#   7693#   7703#   7721#   7722#   7724#   7825#   7834#   7853#
                7854#   7856#   8006#   8014#   8030#   8031#   8033#   8179#   8187#   8203#   8204#   8206#   8346#   8354#   8368#
                8369#   8377#   8446#   8500#   8508#   8512#   8526#   8527#   8535#   8604#   8658#   8666#   8670#   8684#   8685#
                8693#   8760#   8811#   8819#   8823#   8837#   8838#   8846#   8913#   8964#   8972#   8976#   8990#   8991#   8999#
                9066#   9117#   9125#   9129#   9143#   9144#   9152#   9219#   9270#   9278#   9282#   9295#   9296#   9304#   9366#
                9416#   9427#   9431#   9444#   9445#   9453#   9515#   9565#   9576#   9580#   9592#   9593#   9607#   9617#   9634#
                9725#
M$IOSE             1#   1073#
M$LDR0             1#   1073#   3480#   3491#   4095#   4100#   4484#   4489#   5188#   5194#   5200#   5206#   5218#   5241#   5305#
                5371#   5577#   5604#   5692#   5788#   5891#   6008#   6055#   6113#   6116#   6119#   6145#   6259#   6262#   6265#
                6270#   6291#   6319#   6379#   6424#   6427#   6430#   6535#   6552#   6655#   6794#   6943#   7117#   7294#   7447#
                7611#   7723#   7855#   8032#   8205#   8376#   8507#   8534#   8665#   8692#   8818#   8845#   8971#   8998#   9124#
                9151#   9277#   9303#   9426#   9452#   9575#
M$MASK             1#   1073#
M$MCHI             1#   1073#
M$MCLO             1#   1073#
M$MSK1             1#   1073#
M$POP              1#   1073#   1241#   1262#   2462#   2521#   2562#   2610#   2658#   2694#   2791#   2856#   2890#   2967#   5151#
                5169#   5374#   5393#   5422#   5461#   5483#   5590#   5676#   5774#   5877#   5991#   6273#   6538#   6641#   6779#
                6924#   7097#   7271#   7426#   7590#   7702#   7833#   8013#   8186#   8353#   8511#   8669#   8822#   8975#   9128#
```

```
                9281#  9430#  9579#  9616#  9673#  9744#  9782#
M$PRIN             1#  1073#  2440#  2448#  2454#  2513#  2547#  2554#  2591#  2600#  2636#  2643#  2650#  2685#  2730#
                2777#  2821#  2829#  2839#  2848#  2881#  2918#  2945#  2954#  3817#  3883#  5358#  5436#
M$PUSH             1#  1073#  1083#  1230#  1254#  2433#  2510#  2544#  2586#  2633#  2682#  2725#  2816#  2878#  2913#
                5142#  5162#  5185#  5389#  5405#  5434#  5474#  5501#  5502   5602#  5603   5690#  5691   5786#  5787
                5889#  5890   6006#  6007   6289#  6290   6550#  6551   6653#  6654   6792#  6793   6938#  6939   7112#
                7113   7292#  7293   7445#  7446   7609#  7610   7721#  7722   7853#  7854   8030#  8031   8203#  8204
                8368#  8369   8526#  8527   8684#  8685   8837#  8838   8990#  8991   9143#  9144   9295#  9296   9444#
                9445   9592#  9593   9634#  9725#
M$PUT              1#  1073#  2440#  2448#  2454#  2513#  2547#  2554#  2591#  2600#  2636#  2643#  2650#  2685#  2730#
                2777#  2821#  2829#  2839#  2848#  2881#  2918#  2945#  2954#  3817#  3883#  5231#  5358#  5436#  6041#
                6048#  6131#  6138#  6322#  6353#
M$PUT1             1#  1073#  2440#  2441   2448#  2449   2454#  2455   2513#  2514   2515   2547#  2548   2549   2554#
                2555   2556   2591#  2592   2593   2600#  2601   2602   2636#  2637   2638   2643#  2644   2645   2650#
                2651   2652   2685#  2686   2687   2688   2730#  2731   2732   2733   2777#  2778   2779   2780   2781
                2782   2821#  2822   2823   2824   2829#  2830   2831   2832   2839#  2840   2841   2842   2848#  2849
                2850   2881#  2882   2883   2884   2918#  2919   2920   2945#  2946   2954#  2955   2956   2957   2958
                3817#  3818   3883#  3884   3885   3886   5231#  5232   5233   5234   5358#  5359   5360   5436#  5437
                5438   6041#  6042   6043   6044   6048#  6049   6050   6051   6131#  6132   6133   6134   6138#  6139
                6140   6141   6322#  6323   6324   6325   6353#  6354   6355   6356
M$RADI             1#  1073#  9639#  9645#  9651#  9658#  9665#  9730#  9735#
M$RBR0             1#  1073#
M$RNR0             1#  1073#  3477#  3478   4092#  4093   4481#  4482   5218#  5220   5305#  5307
M$SETS             1#  1073#  1083#  1230#  1254#  2433#  2510#  2544#  2586#  2633#  2682#  2725#  2816#  2878#  2913#
                5142#  5162#  5185#  5389#  5405#  5434#  5474#  5502#  5603#  5691#  5787#  5890#  6007#  6290#  6551#
                6654#  6793#  6939#  7113#  7293#  7446#  7610#  7722#  7854#  8031#  8204#  8369#  8527#  8685#  8838#
                8991#  9144#  9296#  9445#  9593#  9634#  9725#
M$STAR             1#  1073#
M$SVC              1#  1073#  2440#  2443   2448#  2451   2454#  2457   2462#  2463   2513#  2517   2521#  2522   2547#
                2551   2554#  2558   2562#  2563   2591#  2595   2600#  2604   2610#  2611   2636#  2640   2643#  2647
                2650#  2654   2658#  2659   2685#  2690   2694#  2695   2730#  2735   2777#  2784   2791#  2792   2821#
                2826   2829#  2834   2839#  2844   2848#  2852   2856#  2857   2881#  2886   2890#  2891   2918#  2922
                2945#  2948   2954#  2960   2967#  2968   3477#  3480#  3481   3491#  3492   3811   3817#  3820   3823#
                3883#  3888   4057#  4092#  4095#  4096   4100#  4101   4305#  4481#  4484#  4485   4489#  4490   4963#
                5145#  5151#  5152   5188#  5189   5194#  5195   5200#  5201   5206#  5207   5213#  5218#  5219   5231#
                5235   5241#  5242   5286#  5305#  5306   5358#  5362   5371#  5372   5374#  5375   5393#  5394   5410#
                5415#  5422#  5423   5436#  5440   5456#  5461#  5462   5476#  5483#  5484   5571   5577#  5578   5581#
                5590#  5591   5604#  5605   5665#  5676#  5677   5692#  5693   5763#  5774#  5775   5788#  5789   5865#
                5877#  5878   5891#  5892   5977#  5991#  5992   6008#  6009   6041#  6045   6048#  6052   6055#  6056
                6093   6105   6113#  6114   6116#  6117   6119#  6120   6131#  6135   6138#  6142   6145#  6146   6167
                6186   6236   6248   6259#  6260   6262#  6263   6265#  6266   6270#  6271   6273#  6274   6291#  6292
                6319#  6320   6322#  6326   6353#  6357   6379#  6380   6424#  6425   6427#  6428   6430#  6431   6449
                6481   6489   6511   6524   6535#  6536   6538#  6539   6552#  6553   6632#  6641#  6642   6655#  6656
                6768#  6779#  6780   6794#  6795   6911#  6924#  6925   6943#  6944   7000#  7029#  7072#  7097#  7098
                7117#  7118   7174#  7203#  7246#  7271#  7272   7294#  7295   7356#  7410#  7426#  7427   7447#  7448
                7510#  7574#  7590#  7591   7611#  7612   7683#  7693#  7702#  7703   7723#  7724   7825#  7833#  7834
                7855#  7856   8006#  8013#  8014   8032#  8033   8179#  8186#  8187   8205#  8206   8346#  8353#  8354
                8376#  8377   8446#  8500#  8507#  8508   8511#  8512   8534#  8535   8604#  8658#  8665#  8666   8669#
                8670   8692#  8693   8760#  8811#  8818#  8819   8822#  8823   8845#  8846   8913#  8964#  8971#  8972
                8975#  8976   8998#  8999   9066#  9117#  9124#  9125   9128#  9129   9151#  9152   9219#  9270#  9277#
                9278   9281#  9282   9303#  9304   9366#  9416#  9426#  9427   9430#  9431   9452#  9453   9515#  9565#
                9575#  9576   9579#  9580   9607   9616#  9617
M$TLAB             1#  1073#  2443#  2451#  2457#  2463#  2517#  2522#  2551#  2558#  2563#  2595#  2604#  2611#  2640#
                2647#  2654#  2659#  2690#  2695#  2735#  2784#  2792#  2826#  2834#  2844#  2852#  2857#  2886#  2891#
                2922#  2948#  2960#  2968#  3477#  3481#  3492#  3811#  3820#  3823#  3888#  4057#  4092#  4096#  4101#
                4305#  4481#  4485#  4490#  4963#  5152#  5189#  5195#  5201#  5207#  5213#  5219#  5235#  5242#  5286#
```

```
                5306#  5362#  5372#  5375#  5394#  5410#  5415#  5423#  5440#  5462#  5484#  5571#  5578#  5581#  5591#
                5605#  5665#  5677#  5693#  5763#  5775#  5789#  5865#  5878#  5892#  5977#  5992#  6009#  6045#  6052#
                6056#  6093#  6105#  6114#  6117#  6120#  6135#  6142#  6146#  6167#  6186#  6236#  6248#  6260#  6263#
                6266#  6271#  6274#  6292#  6320#  6326#  6357#  6380#  6425#  6428#  6431#  6449#  6471#  6489#  6511#
                6524#  6536#  6539#  6553#  6632#  6642#  6656#  6768#  6780#  6795#  6911#  6925#  6944#  7000#  7029#
                7072#  7098#  7118#  7174#  7203#  7246#  7272#  7295#  7356#  7410#  7427#  7448#  7510#  7574#  7591#
                7612#  7683#  7693#  7703#  7724#  7825#  7834#  7856#  8006#  8014#  8033#  8179#  8187#  8206#  8346#
                8354#  8377#  8446#  8500#  8508#  8512#  8535#  8604#  8658#  8666#  8670#  8693#  8760#  8811#  8819#
                8823#  8846#  8913#  8964#  8972#  8976#  8999#  9066#  9117#  9125#  9129#  9152#  9219#  9270#  9278#
                9282#  9304#  9366#  9416#  9427#  9431#  9453#  9515#  9565#  9576#  9580#  9607#  9617#
M$TSTL             1#  1073#  2443#  2451#  2457#  2463#  2517#  2522#  2551#  2558#  2563#  2595#  2604#  2611#  2640#
                2647#  2654#  2659#  2690#  2695#  2735#  2784#  2792#  2826#  2834#  2844#  2852#  2857#  2886#  2891#
                2922#  2948#  2960#  2968#  3477#  3481#  3492#  3811#  3820#  3823#  3888#  4057#  4092#  4096#  4101#
                4305#  4481#  4485#  4490#  4963#  5152#  5189#  5195#  5201#  5207#  5213#  5219#  5235#  5242#  5286#
                5306#  5362#  5372#  5375#  5394#  5410#  5415#  5423#  5440#  5462#  5484#  5571#  5578#  5581#  5591#
                5605#  5665#  5677#  5693#  5763#  5775#  5789#  5865#  5878#  5892#  5977#  5992#  6009#  6045#  6052#
                6056#  6093#  6105#  6114#  6117#  6120#  6135#  6142#  6146#  6167#  6186#  6236#  6248#  6260#  6263#
                6266#  6271#  6274#  6292#  6320#  6326#  6357#  6380#  6425#  6428#  6431#  6449#  6471#  6489#  6511#
                6524#  6536#  6539#  6553#  6632#  6642#  6656#  6768#  6780#  6795#  6911#  6925#  6944#  7000#  7029#
                7072#  7098#  7118#  7174#  7203#  7246#  7272#  7295#  7356#  7410#  7427#  7448#  7510#  7574#  7591#
                7612#  7683#  7693#  7703#  7724#  7825#  7834#  7856#  8006#  8014#  8033#  8179#  8187#  8206#  8346#
                8354#  8377#  8446#  8500#  8508#  8512#  8535#  8604#  8658#  8666#  8670#  8693#  8760#  8811#  8819#
                8823#  8846#  8913#  8964#  8972#  8976#  8999#  9066#  9117#  9125#  9129#  9152#  9219#  9270#  9278#
                9282#  9304#  9366#  9416#  9427#  9431#  9453#  9515#  9565#  9576#  9580#  9607#  9617#
M$WORD             1#  1073#  1132#  1141   1187#  1189   1190   1191   1192   1193   1194   1195   1196   1197   1198
                1199   1200   1201   1202   1203   1204   1205   1206   1207   1208   1209   1210   1211   1212   1213
                1214   1215   1216   3811#  3812   3813   3814   5145#  5415#  5456#  5476#  5571#  5572   5573   5574
                6093#  6094   6095   6096   6105#  6106   6107   6108   6167#  6168   6169   6170   6186#  6187   6188
                6189   6236#  6237   6238   6239   6248#  6249   6250   6251   6449#  6450   6451   6452   6471#  6472
                6473   6474   6489#  6490   6491   6492   6511#  6512   6513   6514   6524#  6525   6526   6527   9607#
                9608   9609   9610   9639#  9645#  9651#  9658#  9665#  9730#  9735#  9778   9779
M$XFER             1#  1073#
OPEN               1#  1073#
PASS            1708#  2459   2607   2788   2963   3034   3096   3204   3241   3278   3323   3375   3405   3495   3536
                3573   3617   3649   3732   3785   3825   3890   3940   3987   4067   4162   4200   4233   4276   4317
                4345   4399   4455   4609   4657   4705   4739   4780   4824   4864   4895   4926   5013   5060   5129
POINTE             1#  1073#  1089
PRINTB             1#  1073#  2439   2447   2512   2546   2590   2599   2635   2684   2729   2820   2880   2917
PRINTF             1#  1073#  3816   5357   5435
PRINTS             1#  1073#
PRINTX             1#  1073#  2453   2553   2642   2649   2776   2828   2838   2847   2944   2953   3882
READBU             1#  1073#
READEF             1#  1073#  5187   5193   5199   5205
RFLAGS             1#  1073#
SAVE            1667#  2434   2587   2726   2914   3004   3062   3127   3230   3270   3309   3351   3396   3441   3517
                3563   3597   3644   3691   3776   3807   3873   3912   3964   4016   4150   4188   4222   4256   4300
                4340   4371   4427   4539   4642   4690   4729   4766   4812   4850   4888   4919   4992   5046   5116
SETPRI             1#  1073#  3479   3490   4094   4099   4483   4488   5240   5370   5603   5691   5787   5890   6007
                6054   6112   6144   6258   6269   6290   6318   6378   6423   6534   6551   6654   6793   6942   7116
                7293   7446   7610   7722   7854   8031   8204   8375   8506   8533   8664   8691   8817   8844   8970
                8997   9123   9150   9276   9302   9425   9451   9574
SETVEC             1#  1073#  5230   6040   6047   6130   6137   6321   6352
SLASH              1#  1073#
STARS              1#  1073#
SVC                1#  1071#  1072
XFER               1#  1073#  5145#  5415#  5456#  5476#
```

```
XFERF       1#   1073#
XFERT       1#   1073#


. ABS.  040504     000


 ERRORS DETECTED:  0

 CVDHBA.BIC,CVDHBA.LST/CRF/NL:TOC/SOL=SVC34R.MLB,CVDHBA.P11
 RUN-TIME: 25 35 3 SECONDS
 RUN-TIME RATIO: 172/64=2.6
 CORE USED:  16K  (31 PAGES)
```