

DHU11

DHU-11 FUNC TST PART 2
CZDHVAO

COPYRIGHT (c) 1983-84
AH-T797A-MC
FICHE 1 OF 1

APR 1984
digital
Made In USA

This microfiche card contains a grid of 100 frames of technical data, arranged in 10 rows and 10 columns. Each frame contains a small table or diagram, likely representing functional test data for the DHU-11 system. The data is organized into columns, with some frames showing vertical bar patterns or specific numerical values. The overall layout is a dense grid of small, structured data points.

.REM 6

IDENTIFICATION

PRODUCT CODE: AC-T796A-MC
PRODUCT NAME: CZDHVAO DMU-11 FUNC TST PART2
PRODUCT DATE: 15 DECEMBER 1983
MAINTAINER: ENE - DIAGNOSTICS GROUP
AUTHOR: ANTHONY HART
MODIFIED BY:

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

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

COPYRIGHT (C) 1983,1984 BY DIGITAL EQUIPMENT CORPORATION
THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL
DEC

PDP
DECUS

UNIBUS
DECTAPE

MASSBUS

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

ORIGINAL RELEASE:

15-DEC-83

ANTHONY HART

TABLE OF CONTENTS

- 1.0 GENERAL PROGRAM CONSIDERATIONS
- 1.1 PROGRAM ABSTRACT
- 1.2 SYSTEM REQUIREMENTS
- 1.3 RELATED DOCUMENTS AND STANDARDS
- 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
- 1.5 ASSUMPTIONS
- 2.0 OPERATING INSTRUCTIONS
- 2.1 COMMANDS
- 2.2 SWITCHES
- 2.3 FLAGS
- 2.4 EXTENDED COMMAND SYNTAX
- 2.4.1 START COMMAND
- 2.4.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)
- 2.4.1.2 PASS SWITCH (/PASS:<PASS-CNT>)
- 2.4.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
- 2.4.1.4 END OF PASS SWITCH (/EOP:<INCR>)
- 2.4.1.5 EFFECT OF START COMMAND
- 2.4.2 RESTART COMMAND
- 2.4.2.1 TESTS, PASS, AND FLAGS SWITCHES
- 2.4.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)
- 2.4.2.3 EFFECT OF RESTART COMMAND
- 2.4.3 CONTINUE COMMAND
- 2.4.3.1 FLAG SWITCH (/FLAGS:<FLAG-LIST>)
- 2.4.3.2 EFFECT OF CONTINUE COMMAND
- 2.4.4 PROCEED COMMAND
- 2.4.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)
- 2.4.4.2 EFFECT OF PROCEED COMMAND
- 2.4.5 ADD COMMAND
- 2.4.6 EFFECT OF ADD COMMAND
- 2.4.7 DROP COMMAND
- 2.4.8 EFFECT OF DROP COMMAND
- 2.4.9 PRINT COMMAND
- 2.4.9.1 EFFECT OF PRINT COMMAND
- 2.4.10 DISPLAY COMMAND
- 2.4.10.1 EFFECT OF DISPLAY COMMAND
- 2.4.11 FLAGS COMMAND
- 2.4.11.1 EFFECT OF FLAGS COMMAND
- 2.4.12 ZFLAGS COMMAND
- 2.4.13 ZFLAGS COMMAND
- 2.4.14 CONTROL CHARACTERS
- 2.5 HARDWARE QUESTIONS
- 2.6 SOFTWARE QUESTIONS
- 2.7 EXTENDED P-TABLE DIALOGUE
- 2.8 QUICK START-UP PROCEDURE (XXDP*)
- 3.0 ERROR INFORMATION
- 3.1 TYPES OF ERROR MESSAGES
- 3.2 SPECIFIC ERROR MESSAGES
- 4.0 PERFORMANCE AND PROGRESS REPORTS
- 5.0 TEST SUMMARIES
- 6.0 EXAMPLE ERROR FREE PASS

1.0 GENERAL PROGRAM CONSIDERATIONS

1.1 PROGRAM ABSTRACT

CZDHVA0 IS PART OF THE DMU-11 FUNCTIONAL VERIFICATION TEST. THIS PART OF THE TEST VERIFIES THAT THE MAJOR COMMUNICATIONS 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 DMU FVT:

- 0 UNIBUS PROCESSOR WITH AT LEAST 32K BYTES OF MEMORY.
- 0 DMU BOARDS INSTALLED ON THE UNIBUS.
- 0 APPROPRIATE PROGRAM LOAD DEVICE SUPPORTING XXDP+ MEDIA OR A DOWN LINE LOADING SYSTEM.

1.3 RELATED DOCUMENTS AND STANDARDS

- 0 XXDP+ USER'S MANUAL - DESCRIBES THE RUNNING OF DIAGNOSTICS UNDER THE XXDP+ MONITOR.

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE PROCESSOR, THE UNIBUS, 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 PERFORMANCE 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 "DDDD".

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

/PASS:DDDDD BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
 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
-----	-----
HJE	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>, SEPERATED 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 "0
UNITS (0) ?" 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 UNITS SWITCH SHOULD
NOT BE USED WITH THIS PROGRAM. 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 -
THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST
START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT
STATISTICAL REPORTING) FLAG IS CLEARED.

2.4.10 DISPLAY COMMAND -

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

2.4.10.1 EFFECT OF DISPLAY COMMAND -
THE HARDWARE P-TABLE FOR THE TEST STATION IS PRINTED IN THE
FORMAT IN WHICH IT WAS 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 DHU-11. THE DEFAULT ANSWER FOR THIS QUESTION IS ADDRESS 160460 (OCTAL).
2. INTERRUPT VECTOR ADDRESS - THIS QUESTION REQUESTS THE INTERRUPT VECTOR ADDRESS OF THE SPECIFIED DHU-11. THE DEFAULT ANSWER IS 310 (OCTAL).
3. ACTIVE LINES BIT MAP - THIS QUESTION REQUESTS AN OCTAL BIT MAP OF THE SERIAL COMMUNICATION LINES ON THE DHU11 WHICH ARE BEING SELECTED FOR TESTING. IF THE BIT IN THE BIT MAP IS SET WHICH CORRESPONDS TO A PARTICULAR LINE (I.E. BIT 5 FOR LINE 5) THAT LINE WILL BE TESTED BY THE FVT.
4. TYPE OF LOOPBACK (1=INTERNAL, 2=STAGGERED, 3=H325) - THIS QUESTION REQUESTS THE TYPE OF LOOPBACK TO BE USED WHEN TESTING THE DHU-11. THE FOLLOWING TYPES ARE SUPPORTED:
 - 0 INTERNAL - ONLY INTERNAL UART LOOPBACK IS TO BE USED IN TESTING THE DHU-11.
 - 0 STAGGERED - STAGGERED BERG CONNECTOR(S) ARE INSTALLED ON THE BERG CONNECTOR SOCKETS OF THE DHU-11.
 - 0 H325 - SINGLE LINE, 25 PIN LOOPBACK CONNECTORS (TYPE H325) ARE INSTALLED ON THE LINES TO BE TESTED.
5. BR LEVEL - THIS QUESTION REQUESTS THE INTERRUPT BR LEVEL OF THE SPECIFIED DHU-11. THE DEFAULT ANSWER IS BR 5.

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 SOFTWARE QUESTIONS AND THE DEFAULT VALUES ARE DESCRIBED IN THE NEXT PARAGRAPH(S).

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 THAT UNIT.
2. EXTENDED ERROR REPORTING - THIS QUESTION ASKS WHETHER EXTENDED ERROR INFORMATION IS REQUIRED OTHER THAN THE "TEST FAILED" MESSAGE, ON EACH ERROR REPORTED. THE DEFAULT IS "NO" I.E. ONLY A MESSAGE REPORTING THE FACT THAT THE TEST FAILED WILL BE PRINTED.
3. NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE - THIS QUESTION IS ASKED ONLY IF THE PREVIOUS QUESTION WAS ANSWERED "YES". THE 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 # (O) ? 3<CR>
Q-FACTOR (O) 0 ? <CR>

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

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

UNIT 7
CSR ADDRESS (O) ? 160000<CR>
SUB-DEVICE # (O) ? 6<CR>
Q-FACTOR (O) 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 (0) ? 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 (0) ? 8<CR>  
  
UNIT 1
```

CSR ADDRESS (0) ? 160000<CR>
SUB-DEVICE # (0) ? 0-7<CR>
Q-FACTOR (0) 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) 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 SPECIFIC ERROR MESSAGES

THIS PROGRAM IS INTENDED TO PROVIDE A GO/NOGO INDICATION
OF THE FUNCTIONALITY OF THE DHU-11 BOARDS. TO EXECUTE THE
PROGRAM IN THIS MODE THE OPERATOR NEED ONLY ANSWER THE
"EXTENDED ERROR REPORTING" SOFTWARE QUESTION WITH "NO". THE
PROGRAM WILL THEN ONLY PRINT THE NAME OF THE FAILING TEST
THE TEST AND ERROR NUMBERS. FOR A LIST OF THE TEST NAMES
IN THIS PROGRAM SEE THE TEST SUMMARIES SECTION OF THIS
DOCUMENT. AN EXAMPLE OF SUCH A AN ERROR MESSAGE IS THE
FOLLOWING:

CZDHV DVC FTL ERR 04106 ON UNIT 00 TST 003 SUB 000 PC: XXXXXX
DMA_ABORT BIT TEST FAILED

THIS ERROR INDICATES THAT A FATAL ERROR WAS ENCOUNTERED
DURING THE TEST WHICH TESTS THE DMA_ABORT BIT.

IF THE OPERATOR HAD REQUESTED EXTENDED ERROR REPORTING THE
SAME ERROR WOULD BE REPORTED AS FOLLOWS:

CZDHV DVC FTL ERR 04106 ON UNIT 00 TST 003 SUB 000 PC: XXXXXX
DMA_ABORT BIT TEST FAILED
DMA_START BIT FOUND SET AFTER DMA ABORTED ON LINE: 8

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 FURTHER INFORMATION SEE THE SWITCHES SECTION OF THIS DOCUMENT.

5.0 TEST SUMMARIES

THE FOLLOWING ARE INCLUDED WITHIN CZDHVA:

1. DEVICE REGISTER ACCESS TEST - VERIFIES THAT THE UUT REGISTERS WILL RESPOND WITH THE CORRECT UNIBUS HANDSHAKING SIGNALS. VERIFIES THAT THE UUT IS AT THE CORRECT ADDRESS.
2. DMA.START TEST - VERIFIES THAT EACH DMA START BIT WILL INITIATE A DMA TRANSMISSION ON A LINE
3. DMA.ABORT TEST - VERIFIES THAT EACH DMA ABORT BIT WILL STOP A DMA TRANSMISSION, RETURN A TX.ACTION AND SUCCESSFULLY RESTART THE DMA.
4. O.AUTO INACTIVE TEST - VERIFIES THAT THE DUT WILL NOT RESPOND TO INCOMING XON AND XOFF CHARACTERS WHEN O.AUTO IS CLEAR.
5. O.AUTO ACTIVE TEST - VERIFIES THAT THE DUT RESPONDS CORRECTLY TO INCOMING FLOW CONTROL CHARACTERS WHEN ACTIVE
6. I.AUTO INACTIVE TEST - VERIFIES THAT THE DUT WILL NOT GENERATE XON AND XOFF CHARACTERS IN RESPONSE TO THE APPROPRIATE FIFO CONDITIONS WHEN I.AUTO IS INACTIVE.
7. I.AUTO ACTIVE TEST - VERIFIES THAT THE DUT WILL GENERATE XON AND XOFF CHARACTERS IN RESPONSE TO THE APPROPRIATE FIFO CONDITIONS WHEN I.AUTO IS ACTIVE.
8. FIFO DATA TEST - VERIFIES THAT THE FIFO WILL HOLD 256 CHARACTERS WITHOUT CORRUPTING DATA.
9. FIFO 3/4 LEVEL INACTIVE TEST - VERIFIES THAT THE 3/4 LEVEL ALARM DOES NOT BECOME ACTIVE BELOW THE 3/4 LEVEL.
10. FIFO 3/4 LEVEL ACTIVE TEST - VERIFIES THAT THE 3/4 LEVEL ALARM BECOMES ACTIVE WHEN THE FIFO IS 3/4 FULL.
11. FIFO 3/4 LEVEL ACTIVE/INACTIVE TEST - VERIFIES THAT THE 3/4 LEVEL ALARM, ONCE ACTIVATED, REMAINS ACTIVE UNTIL THE FIFO IS REDUCED BELOW THE 1/2 LEVEL.
12. FIFO 1/2 LEVEL TEST - VERIFIES THAT THE FIFO 1/2 LEVEL ALARM SYSTEM BECOMES ACTIVE AND INACTIVE AT THE CORRECT LEVELS.

- 13. DTR TEST - VERIFIES THAT CHANGING THE STATE OF THE DTR BIT AFFECTS THE STATE OF THE DTR CONTROL LINE.
- 14. RTS TEST - VERIFIES THAT CHANGING THE STATE OF THE RTS BIT AFFECTS THE STATE OF THE RTS CONTROL LINE.
- 15. DSR TEST - VERIFIES THAT THE DSR STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK DTR CONTROL LINE.
- 16. RI TEST - VERIFIES THAT THE RI STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK DTR CONTROL LINE.
- 17. CTS TEST - VERIFIES THAT THE CTS STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK RTS CONTROL LINE.
- 18. DCD TEST - VERIFIES THAT THE DCD STATUS SIGNAL CORRECTLY REPORTS THE STATE OF THE LOOPED BACK RTS CONTROL LINE.
- 19. 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.
- 20. 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.
- 21. REPORT BHP CODES TEST - THIS PSEUDO TEST REPORTS THE FIRST 32 CHARACTERS WHICH WERE DISCOVERED IN THE FIFO DURING THE EXECUTION OF THE OTHER TESTS. THIS AVOIDS INTERRUPTION OF THE OTHER TESTS BY THESE CODES IF THEY ARE NOT CRITICAL TO THE PERFORMANCE OF THE TESTS.

6.4 EXAMPLE ERROR FREE PASS

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

```
.R CZDHVAO
CZDHVAO.BIN
DRS
CZDHV-A-0
DHU FUNC TST PART2
UNIT IS DHU-11
RESTRT ADDR: 147670
DR>STA/PAS:1

CHANGE HW (L) ? Y

# UNITS (D) ? 2

UNIT 0
CSR ADDRESS: (0) 160460 ? +Z
```

UNIT 1
CSR ADDRESS: (0) 160460 ? 160500
INTERRUPT VECTOR ADDRESS: (0) 310 ? 320
ACTIVE LINE BIT MAP: (0) 177777 ? <CR>
TYPE OF LOOPBACK (1=INTERNAL, 2=H3277, 3=H325): (0) 2 ? 1
INTERRUPT BR LEVEL: (0) 5 ? <CR>

CHANGE SW (L) ? Y

REPORT UNIT NUMBER AS EACH UNIT IS TESTED: (L) Y ? <CR>
EXTENDED ERROR REPORTING: (L) N ? <CR>

TESTING UNIT : 0

TESTING UNIT : 1

CZDHV EOP 1
0 TOTAL ERRS

DR>

&

1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052 000000
1053
1054
1055
1056
1057
1058 000001
1059 000001
1060 000001
1061 000001
1062 000001
1063
1064
1065
1066
1067

.LIST SEQ,LOC,BIN,MEB
.NLIST CND

: FVTA.PHD
: *****

.SBTTL PROGRAM HEADER

.MCALL SVC ; INITIALIZE SUPERVISOR MACROS
SVC

: IF STRUCTURED MACROS ARE TO BE USED, ADD ".MCALL STRUCT" AND "STRUCT"
: TO INITIALIZE THE STRUCTURED MACROS.

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

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

PROGRAM HEADER

```

1068
1069
1070 000000
1071
1072          002000
1073
1074 002000
1075
1076
1077
1078
1079
1080
1081 002000
1082
1099
1100 002000
      002000
      002000      103
      002001      132
      002002      104
      002003      110
      002004      126
      002005      000
      002006      000
      002007      000
      002010
      002010      101
      002011
      002011      060
      002012
      002012 000000
      002014
      002014 000022
      002016
      002016 035552
      002020
      002020 036034
      002022
      002022 002206
      002024
      002024 002220
      002026
      002026 036336
      002030
      002030 000000
      002032
      002032 000000
      002034
      002034 000000
      002036
      002036 000000
      002040
      002040 002124
      002042
      002042 000340
      002044

```

```

.ENABL ABS
;.ENABL AMA
      "      2000

```

BGNMOD

```

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

```

POINTER BGNRPT,BGNSW,BGNSFT,BGNDU,ERRTBL

HEADER CZDHV,A,0,22,0,PRI07

```

L$NAME::
          .ASCII /C/
          .ASCII /Z/
          .ASCII /D/
          .ASCII /H/
          .ASCII /V/
          .BYTE 0
          .BYTE 0
          .BYTE 0
L$REV::
          .ASCII /A/
L$DEPO::
          .ASCII /O/
L$UNIT::
          .WORD 0
L$TIML::
          .WORD 22
L$HPCP::
          .WORD L$HARD
L$SPCP::
          .WORD L$SOFT
L$HPTP::
          .WORD L$HW
L$SPTP::
          .WORD L$SW
L$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::

```

PROGRAM HEADER

002044 000000
 002046
 002046 000000
 002050
 002050 003
 002051 003
 002052
 002052 000000
 002054 000000
 002056
 002056 000000
 002060
 002060 004130
 002062
 002062 017022
 002064
 002064 000000
 002066
 002066 000000
 002070
 002070 000000
 002072
 002072 017702
 002074
 002074 000000
 002076
 002076 004140
 002100
 002100 104035
 002102
 002102 004060
 002104
 002104 017036
 002106
 002106 017664
 002110
 002110 017662
 002112
 002112 017030
 002114
 002114 000000
 002116
 002116 000000
 002120
 002120 000000

1101

L\$EXP1:: .WORD 0
 L\$MREV:: .WORD 0
 .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
 L\$DLY:: .WORD 0
 L\$HIME:: .WORD 0

```

1113
1114
1115
1116
1117
1118
1119
1120 002122
      002122 000030
      002124
      002124 020020
      002126 020302
      002130 020706
      002132 021344
      002134 022144
      002136 022744
      002140 023366
      002142 024060
      002144 024366
      002146 024676
      002150 025376
      002152 026074
      002154 026552
      002156 027176
      002160 027714
      002162 030660
      002164 031414
      002166 032150
      002170 032620
      002172 033270
      002174 033740
      002176 034410
      002200 035040
      002202 035470
1121

```

.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 24

```

      .WORD 24
L$DISPATCH:
      .WORD T1
      .WORD T2
      .WORD T3
      .WORD T4
      .WORD T5
      .WORD T6
      .WORD T7
      .WORD T8
      .WORD T9
      .WORD T10
      .WORD T11
      .WORD T12
      .WORD T13
      .WORD T14
      .WORD T15
      .WORD T16
      .WORD T17
      .WORD T18
      .WORD T19
      .WORD T20
      .WORD T21
      .WORD T22
      .WORD T23
      .WORD T24

```

1129
1130
1131
1132
1133
1134

```
*****  
: FVTA.DHT  
:*****
```

1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146

.SBTTL DEFAULT HARDWARE P-TABLE

```
***  
: THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF  
: THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE  
: IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES.  
: AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.  
:--
```

1147 002204
002204 000004
002206
002206

BGNHW DFPTBL

.WORD L10000-L#HW/2
L#HW::
DFPTBL::

1148

1149 002206 160460
1150 002210 000310
1151 002212 177777
1152 002214 002
1153 002215 005

.WORD 160460 ;DEFAULT CSR ADDRESS
.WORD 310 ;DEFAULT VECTOR ADDRESS
.WORD 177777 ;DEFAULT ACTIVE LINES BIT MAP
.BYTE 2 ;DEFAULT LOOPBACK MODE
.BYTE 5 ;DEFAULT BR LEVEL

1154

1155 002216
002216

ENDHW

L10000:

1157
1158

1159
1160
1161
1162
1163
1164
1165

;
; FVTA.SWT
;

1166
1167
1168
1169
1170
1171
1172
1173
1174

.SBTTL SOFTWARE P-TABLE
; **
; THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
; PROGRAM AS OPERATIONAL PARAMETERS. THESE PARAMETERS ARE
; SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
; AT RUN TIME.
; --

1175
1176
1177
1178
1179
1180

002216
002216 000002
002220
002220
002220 000020
002222 000000
002224
002224

BGNSW SFPTBL
L1SW:: .WORD L10001-L1SW/2
SFPTBL::
OPTION:: .WORD 20 ;BIT MAP OF PROGRAM CONTROL FLAGS
NDERPT:: .WORD 0 ;DEFAULT NUMBER OF INDIVIDUAL DATA ERRORS TO RPT.
ENDSW
L10001:

```

1182
1183 ;*****
1184 ;
1185 ;           FVTA.EQU
1186 ;
1187 ;*****
1188
1189
1190 .SBTTL GLOBAL EQUATES SECTION
1191
1201
1202
1203
1204
1205 ;**
1206 ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
1207 ; ARE USED IN MORE THAN ONE TEST.
1208 ;--
1209           000020           NUMLNS==20           ;NUMBER OF LINES ON DHU11 IS 8.
1210           177777           MAPLNS==177777        ;BIT MAP OF LINES ON DHU11.
1211
1212 ;***** DEVICE REGISTER OFFSETS FROM THE CSR'S ADDRESS *****
1213           000000           CSRO==0             ;CSR REGISTER OFFSET FROM THE CSR ADDRESS
1214           000002           RBUFO==2            ;RECEIVE REGISTER OFFSET FROM THE CSR ADDRESS
1215           000002           RXTIMO==2           ;RECIEVE TIMER REGISTER OFFSET FROM THE CSR ADDRESS
1216           000004           LPRO==4             ;LINE PARAMETER REGISTER OFFSET FROM THE CSR ADDRESS
1217           000006           FLSO==6             ;FIFOSIZE/STATUS REGISTER OFFSET FROM THE CSR ADDRESS
1218           000006           FDATO==6            ;FIFODATA REGISTER OFFSET FROM THE CSR ADDRESS
1219           000010           LNCTRO==10          ;LINE CONTROL REGISTER OFFSET FROM THE CSR ADDRESS
1220           000012           TXAD10==12          ;TRANSMIT ADDRESS 1 REGISTER OFFSET FROM THE CSR ADDRESS
1221           000014           TXAD20==14          ;TRANSMIT ADDRESS 2 REGISTER OFFSET FROM THE CSR ADDRESS
1222           000016           TXBFCO==16          ;TRANSMIT COUNT REGISTER OFFSET FROM THE CSR ADDRESS
1223
1224 ;***** EQUATES USED WITH RESPECT TO THE RX BUFFER *****
1225           000020           RXBETX==16.          ;LEVEL OF RX BUFFER AT WHICH TO RE-ENABLE TRANSMISSION.
1226           000030           RXBDTX==24.          ;LEVEL OF RX BUFFER AT WHICH TO DISABLE TRANSMISSION.
1227           000100           RXBFUL==64.          ;TOTAL CHARACTER CAPACITY OF THE RX BUFFER.
1228
1229
1244 002224           EQUALS
;
; BIT DIFINITIONS
;
100000           BIT15== 100000
040000           BIT14== 40000
020000           BIT13== 20000
010000           BIT12== 10000
004000           BIT11== 4000
002000           BIT10== 2000
001000           BIT09== 1000
000400           BIT08== 400
000200           BIT07== 200
000100           BIT06== 100
000040           BIT05== 40
000020           BIT04== 20
000010           BIT03== 10
000004           BIT02== 4

```

```

000002 BIT01== 2
000001 BIT00== 1
;
001000 BIT9== BIT09
000400 BIT8== BIT08
000200 BIT7== BIT07
000100 BIT6== BIT06
000040 BIT5== BIT05
000020 BIT4== BIT04
000010 BIT3== BIT03
000004 BIT2== BIT02
000002 BIT1== BIT01
000001 BIT0== BIT00
;
; EVENT FLAG DEFINITIONS
; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
;
000040 EF.START== 32. ; START COMMAND WAS ISSUED
000037 EF.RESTART== 31. ; RESTART COMMAND WAS ISSUED
000036 EF.CONTINUE== 30. ; CONTINUE COMMAND WAS ISSUED
000035 EF.NEW== 29. ; A NEW PASS HAS BEEN STARTED
000034 EF.PWR== 28. ; A POWER-FAIL/POWER-UP OCCURRED
;
; PRIORITY LEVEL DEFINITIONS
;
000340 PRI07== 340
000300 PRI06== 300
000240 PRI05== 240
000200 PRI04== 200
000140 PRI03== 140
000100 PRI02== 100
000040 PRI01== 40
000000 PRI00== 0
;
; OPERATOR FLAG BITS
;
000004 EVL== 4
000010 LOT== 10
000020 ADR== 20
000040 IDU== 40
000100 ISR== 100
000200 UAM== 200
000400 BOE== 400
001000 PNT== 1000
002000 PRI== 2000
004000 IXE== 4000
010000 IBE== 10000
020000 IER== 20000
040000 LOE== 40000
100000 HOE== 100000

```

1247
 1248
 1249
 1250
 1251
 1252
 1253
 1254
 1255
 1256
 1257
 1258
 1259
 1260
 1261
 1262
 1263
 1264
 1265
 1266
 1267 002224 000200
 1268 002226 000204
 1269 002230 177777
 1270 002232 000
 1271 002233 004
 1272 002234 000000
 1273
 1274
 1275
 1276
 1277 002236
 1278 002236 160020
 1279 002240 160022
 1280 002242 160024
 1281 002244 160026
 1282
 1283 002246 160030
 1284 002250 160032
 1285 002252 160034
 1286 002254 160036
 1287
 1288
 1289
 1290
 1291 002256 000000
 1292 002260 000000
 1293 002262 000001
 1294 002264 000000
 1295 002266 031463
 1296 002270 146314
 1297 002272 000000
 1298 002274 000000
 1299 002276 000000
 1300 002300 000000
 1301 002302 000000
 1302 002304 000000
 1303 002306 000000

```

;*****
;
;           FVTB.GDT
;
;*****

.SBTTL GLOBAL DATA SECTION

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

;*****
;           UNIT VARIABLE AREA
;*****

RXVECA:: .WORD 200 ;RX VECTOR ADDRESS.
TXVECA:: .WORD 204 ;TX VECTOR ADDRESS.
ACTLNS:: .WORD 177777 ;ACTIVE LINE BIT MAP.
LOPBCK:: .BYTE 0 ;LOOPBACK MODE
BRLEVL:: .BYTE 4 ;INTERRUPT BUS REQUEST LEVEL
UNITN:: .WORD 0 ;UNIT NUMBER.

;*****
;           DEVICE REGISTER ADDRESS TABLE
;*****
DRADRT::
RXTMA:: RBUFA:: .WORD 160022 ;DHU-11 RECIEVE BUFFER/TIMER ADDRESS.
        LPRA:: .WORD 160024 ;DHU-11 LINE PARAMETER REGISTER ADDRESS.
FDATA:: FLSA:: .WORD 160026 ;DHU-11 FIFO SIZE/LINE STATUS REGISTER ADDRESS.
        ;AND FIFO DATA REGISTER ADDRESS.
        LNCTRA:: .WORD 160030 ;DHU-11 LINE CONTROL REGISTER ADDRESS.
        TXAD1A:: .WORD 160032 ;DHU-11 TRANSMIT BUFFER 1 REGISTER ADDRESS
        TXAD2A:: .WORD 160034 ;DHU-11 TRANSMIT BUFFER 2 REGISTER ADDRESS
        TXBFCA:: .WORD 160036 ;DHU-11 TRANSMIT BUFFER COUNT REGISTER ADDRESS

;*****
;           ASSORTED GLOBAL VARIABLES:
;*****
BUFPTR:: .WORD 0 ;STORAGE FOR RECEIVE CHARACTER BUFFER POINTER.
CTRLCF:: .WORD 0 ;STORAGE FOR THE CONTROL-C FLAG.
TSTNUM:: .WORD 1 ;STORAGE FOR THE TEST NUMBER.
IESTAT:: .WORD 0 ;STORAGE FOR STATES OF THE DUT INT ENABLE BITS.
LGRP1M:: .WORD 31463 ;BIT MAP OF LINES IN LINE GROUP I.
LGRP2M:: .WORD 146314 ;BIT MAP OF LINES IN LINE GROUP II.
PASCNT:: .WORD 0 ;STO'G FOR PASS COUNT USED IN ROM VERSION# TST.
RXINTC:: .WORD 0 ;STORAGE FOR RECEIVER INTERRUPT FLAGS.
RXINTF:: .WORD 0 ;STORAGE FOR RECEIVER INTERRUPT FLAGS.
TXINTC:: .WORD 0 ;STORAGE FOR TRANSMIT INTERRUPT COUNT.
TXINTF:: .WORD 0 ;STORAGE FOR TRANSMIT INTERRUPT FLAGS.
TP4VEC:: .WORD 0 ;STORAGE FOR THE NORMAL 004 TRAP VECTOR.
TP4FLG:: .WORD 0 ;FLAGS SET WHEN AN EXPECTED 004 TRAP OCCURS.

```

1304 002310 000000
1305
1306
1307
1308 002312 177546
1309 002314 000300
1310 002316 000100
1311 002320 000074
1312 002322 000000
1313 002324 000000
1314 002326 000170
1315 002330 000170
1316 002332 000021
1317 002334 000062
1318
1319
1320
1321
1322 002336 177572
1323 002340 000000
1324 002342 000000
1325
1326
1327
1328
1329 002344 000001
1330 002346 000002
1331 002350 000004
1332 002352 000010
1333 002354 000020
1334 002356 000040
1335 002360 000100
1336 002362 000200
1337 002364 000400
1338 002366 001000
1339 002370 002000
1340 002372 004000
1341 002374 010000
1342 002376 020000
1343 002400 040000
1344 002402 100000
1345
1346
1347
1348
1349 002404
1350 002404 000000
1351 002406 000000
1352 002410 000000
1353 002412 000000
1354 002414 000000
1355
1356
1357
1358 002416 000000
1359 002420
1360 002620

```

WORD1:: .WORD 0 ;LOCATION FOR PASSING INDIRECT PARAMETERS.
;*****
; LINE TIME CLOCK VARIABLES AND STORAGE.
;*****
CLKCSR:: .WORD 177546 ;CSR ADDRESS OF THE LTC.
CLKBRL:: .WORD PRI06 ;INTERRUPT PRIORITY LEVEL OF THE LTC.
CLKVEC:: .WORD 100 ;INTERRUPT VECTOR ADDRESS OF THE LTC.
CLKHRZ:: .WORD 60. ;INTERRUPT FREQUENCY OF THE LTC.
TIMER1:: .WORD 0 ;HARDWARE CLOCK COUNTER #1.
TIMER2:: .WORD 0 ;HARDWARE CLOCK COUNTER #2.
TIMER3:: .WORD 120. ;HARDWARE BREAK COUNTER LOCATION.
BCOUNT:: .WORD 120. ;BREAK COUNT VALUE IN CLOCK TICKS.
MSTICK:: .WORD 17. ;NUMBER OF MILLI-SECONDS PER LTC TICK.
MSLCNT:: .WORD 62 ;LOOP COUNT (USED BY MSLOOP) TO DELAY 1 MS.
;*****
; MEMORY MANAGEMENT VARIABLES AND FLAGS.
;*****
MMSRO:: .WORD 177572 ;ADDRESS OF MEM MGT STATUS REGISTER #0.
MMPRES:: .WORD 0 ;MEM MGT PRESENT FLAG (0 IF MM NOT PRESENT).
MMENAB:: .WORD 0 ;MEM MGT ENABLED FLAG (0 IF MM NOT ENABLED).
;*****
; TABLE OF WORDS WITH CORRESPONDING BIT SET FOR GENERATION OF BIT MAPS.
;*****
BITTBL:: .WORD 1 ;BIT 0 SET.
; .WORD 2 ;BIT 1 SET.
; .WORD 4 ;BIT 2 SET.
; .WORD 10 ;BIT 3 SET.
; .WORD 20 ;BIT 4 SET.
; .WORD 40 ;BIT 5 SET.
; .WORD 100 ;BIT 6 SET.
; .WORD 200 ;BIT 7 SET.
; .WORD 400 ;BIT 8 SET.
; .WORD 1000 ;BIT 9 SET.
; .WORD 2000 ;BIT 10 SET.
; .WORD 4000 ;BIT 11 SET.
; .WORD 10000 ;BIT 12 SET.
; .WORD 20000 ;BIT 13 SET.
; .WORD 40000 ;BIT 14 SET.
; .WORD 100000 ;BIT 15 SET.
;*****
;* GPR SAVE AREAS ZERO AND ONE.
;*****
GPRS0B:: ;BASE OF GPR SAVE AREA NUMBER ZERO.
; .WORD 0 ;WORD 1, STORAGE FOR R1.
; .WORD 0 ;WORD 2, STORAGE FOR R2.
; .WORD 0 ;WORD 3, STORAGE FOR R3.
; .WORD 0 ;WORD 4, STORAGE FOR R4.
; .WORD 0 ;WORD 5, STORAGE FOR R5.
;*****
; STORAGE AREA FOR THE BMP CODE QUEUE.
;*****
BMPCQP:: .WORD 0 ;POINTER USED TO ACCESS THE NEXT CELL IN QUE.
BMPCQB:: .BLKW 64. ;STORAGE FOR 32 CELLS, TEST# PLUS BMP CODE.
BMPCQE:: ;LAST ADDRESS PLUS 2 OF THE BMP CODE QUEUE.

```

1361
1362
1363
1364 002620
1365 002620 000000
1366 002622 000000
1367 002624 000000
1368 002626 000000
1369 002630 000000
1370 002632 000000
1371 002634 000000
1372 002636 000000
1373 002640 000000
1374 002642 000000
1375 002644 000000
1376 002646 000000
1377 002650 000000
1378 002652 000000
1379 002654 000000
1380 002656 000000
1381 002660
1382
1383
1384
1385 002660
1386 002660
1387 003260
1388 003460
1389 003660
1390 003660
1391
1392
1393
1394 003720
1395
1396
1397
1398
1399
1400
1401 003760
1402 003760 000000
1403 003762 000002
1404 003764 000004
1405 003766 000006
1406 003770 000010
1407 003772 000012
1408 003774 000014
1409 003776 000016
1410 004000 000020
1411 004002 000022
1412 004004 000024
1413 004006 000026
1414 004010 000030
1415 004012 000032
1416 004014 000034
1417 004016 000036

```

;*****
; STORAGE AREA FOR THE CONTENTS OF THE DUT STAT REGISTER STATES.
;*****
STSTB::
;BASE OF DUT STAT STORAGE TABLE.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 0.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 1.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 2.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 3.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 4.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 5.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 6.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 7.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 8.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 9.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 10.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 11.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 12.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 13.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 14.
.WORD 0 ;STORAGE FOR STAT REGISTER FOR LINE 15.

STSTE::
;END OF DUT STAT STORAGE TABLE.
;*****
; GENERAL TABLE AND BUFFER AREA--513 WORDS.
;*****
BUFBAS::
;BASE OF MEMORY BUFFER.
ERLTBL:: .BLKW 128. ;FIRST HALF OF GENERAL TABLE OR BUFFER.
BUFHID:: .BLKW 64. ;SECOND HALF OF GENERAL TABLE OR BUFFER.
BUF3QT:: .BLKW 64. ;LAST QUARTER OF THE BUFFER AREA.
BUFEND::
;END OF GENERAL PURPOSE MEMORY BUFFER.
ENDETB:: .BLKW 16. ;BUFFER OVERFLOW SPACE.
;*****
; RECEPTION TABLE OF COUNTERS
;*****
RXCNTB:: .BLKW 16. ;RECEPTION CHARACTER COUNTERS TABLE.
;*****
;* TABLE FOR STORAGE OF RX/TX LINE NUMBER ASSOCIATIONS.
;* THE ASSOCIATIONS ARE STORED AS LINE NUMBER TIMES 2 FOR USE AS OFFSETS
;* WHEN ACCESSING A TABLE OF WORDS.
;* NOTE: DO NOT WRITE A NON-ZERO VALUE INTO THE UPPER BYTE OF ANY ENTRY.
;*****
TXRXLB::
;BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
.WORD 0 ;TX/RX LINE OFFSET FOR RX/TX LINE 0.
.WORD 2. ;TX/RX LINE OFFSET FOR RX/TX LINE 1.
.WORD 4. ;TX/RX LINE OFFSET FOR RX/TX LINE 2.
.WORD 6. ;TX/RX LINE OFFSET FOR RX/TX LINE 3.
.WORD 8. ;TX/RX LINE OFFSET FOR RX/TX LINE 4.
.WORD 10. ;TX/RX LINE OFFSET FOR RX/TX LINE 5.
.WORD 12. ;TX/RX LINE OFFSET FOR RX/TX LINE 6.
.WORD 14. ;TX/RX LINE OFFSET FOR RX/TX LINE 7.
.WORD 16. ;TX/RX LINE OFFSET FOR RX/TX LINE 8.
.WORD 18. ;TX/RX LINE OFFSET FOR RX/TX LINE 9.
.WORD 20. ;TX/RX LINE OFFSET FOR RX/TX LINE 10.
.WORD 22. ;TX/RX LINE OFFSET FOR RX/TX LINE 11.
.WORD 24. ;TX/RX LINE OFFSET FOR RX/TX LINE 12.
.WORD 26. ;TX/RX LINE OFFSET FOR RX/TX LINE 13.
.WORD 28. ;TX/RX LINE OFFSET FOR RX/TX LINE 14.
.WORD 30. ;TX/RX LINE OFFSET FOR RX/TX LINE 15.

```

```

1418 004020
1419
1420
1421
1422
1423
1424
1425 004020
1426 004020      000
1427 004021      001
1428 004022      002
1429 004023      003
1430 004024      004
1431 004025      005
1432 004026      006
1433 004027      007
1434 004030      010
1435 004031      011
1436 004032      012
1437 004033      013
1438 004034      014
1439 004035      015
1440 004036      016
1441 004037      017
1442 004040
1443
1444
1445
1446
1447
1448
1449
1450
1451 004040
1452 004040      004
1453 004041      006
1454 004042      000
1455 004043      002
1456 004044      014
1457 004045      016
1458 004046      010
1459 004047      012
1460 004050      024
1461 004051      026
1462 004052      020
1463 004053      022
1464 004054      034
1465 004055      036
1466 004056      030
1467 004057      032
1468
1481 004060
      004060
      004060 000000
      004062 000000
      004064 000000
      004066 000000

```

```

TXRXLE::                                ;END OF TX/RX LINE NUMBER ASSOCIATION TABLE.
      .EVEN                               ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
;*****
;* TABLE FOR STORAGE OF RX/TX LINE NUMBER ASSOCIATIONS.
;* THE ASSOCIATIONS ARE STORED AS LINE NUMBERS WHICH CAN BE USED AS SUCH OR
;* AS OFFSETS WHEN ACCESSING A TABLE OF BYTES.
;*****
TXRLNB::                                ;BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
      .BYTE 0                             ;TX/RX LINE FOR RX/TX LINE 0.
      .BYTE 1                             ;TX/RX LINE FOR RX/TX LINE 1.
      .BYTE 2                             ;TX/RX LINE FOR RX/TX LINE 2.
      .BYTE 3                             ;TX/RX LINE FOR RX/TX LINE 3.
      .BYTE 4                             ;TX/RX LINE FOR RX/TX LINE 4.
      .BYTE 5                             ;TX/RX LINE FOR RX/TX LINE 5.
      .BYTE 6                             ;TX/RX LINE FOR RX/TX LINE 6.
      .BYTE 7                             ;TX/RX LINE FOR RX/TX LINE 7.
      .BYTE 8                             ;TX/RX LINE FOR RX/TX LINE 8.
      .BYTE 9                             ;TX/RX LINE FOR RX/TX LINE 9.
      .BYTE 10                            ;TX/RX LINE FOR RX/TX LINE 10.
      .BYTE 11                            ;TX/RX LINE FOR RX/TX LINE 11.
      .BYTE 12                            ;TX/RX LINE FOR RX/TX LINE 12.
      .BYTE 13                            ;TX/RX LINE FOR RX/TX LINE 13.
      .BYTE 14                            ;TX/RX LINE FOR RX/TX LINE 14.
      .BYTE 15                            ;TX/RX LINE FOR RX/TX LINE 15.
TXRLNE::                                ;END OF TX/RX LINE NUMBER ASSOCIATION TABLE.
      .EVEN                               ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
;*****
;* TABLE OF TX/RX LINE NUMBER ASSOCIATIONS IN STAGGERED LOOPBACK.
;* THE ASSOCIATIONS ARE STORED AS LINE NUMBER TIMES 2 FOR USE AS OFFSETS
;* WHEN ACCESSING A TABLE OF WORDS.
;* THIS IS A TABLE OF DATA FOR READING ONLY. USE TO LOAD THE ABOVE TABLE.
;* NOTE: MUST CONVERT FROM BYTES TO WORDS WHEN LOADING ABOVE TABLE.
;*****
STGTRB::                                ;BASE OF STAGGERED TX/RX LINE NUMBER TABLE.
      .BYTE 4                             ;TX/RX LINE OFFSET FOR RX/TX LINE 0.
      .BYTE 6                             ;TX/RX LINE OFFSET FOR RX/TX LINE 1.
      .BYTE 0                             ;TX/RX LINE OFFSET FOR RX/TX LINE 2.
      .BYTE 2                             ;TX/RX LINE OFFSET FOR RX/TX LINE 3.
      .BYTE 12                            ;TX/RX LINE OFFSET FOR RX/TX LINE 4.
      .BYTE 14                            ;TX/RX LINE OFFSET FOR RX/TX LINE 5.
      .BYTE 8                             ;TX/RX LINE OFFSET FOR RX/TX LINE 6.
      .BYTE 10                            ;TX/RX LINE OFFSET FOR RX/TX LINE 7.
      .BYTE 20                            ;TX/RX LINE OFFSET FOR RX/TX LINE 8.
      .BYTE 22                            ;TX/RX LINE OFFSET FOR RX/TX LINE 9.
      .BYTE 16                            ;TX/RX LINE OFFSET FOR RX/TX LINE 10.
      .BYTE 18                            ;TX/RX LINE OFFSET FOR RX/TX LINE 11.
      .BYTE 28                            ;TX/RX LINE OFFSET FOR RX/TX LINE 12.
      .BYTE 30                            ;TX/RX LINE OFFSET FOR RX/TX LINE 13.
      .BYTE 24                            ;TX/RX LINE OFFSET FOR RX/TX LINE 14.
      .BYTE 26                            ;TX/RX LINE OFFSET FOR RX/TX LINE 15.
      .EVEN                               ;GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.
ERRTBL
ERRTYP::                                .WORD 0
ERRNBR::                                .WORD 0
ERRMSG::                                .WORD 0
ERRBLK::                                .WORD 0

```

L#ERRTBL::

J3

1482
1483

.EVEN

```

1485 .SBTTL GPR HANDLING ROUTINES FOR SUBROUTINE CALLS.
1486 ;*****
1487 ;* THERE ARE 4 ROUTINES AND MACRO DEFINITIONS USED FOR THE HANDLING OF
1488 ;* GPR VALUES DURING SUBROUTINE CALLS WITHIN THIS PROGRAM. THE FOUR
1489 ;* ROUTINES/MACRO CALLS HAVE THE FOLLOWING NAMES:
1490 ;*
1491 ;* SAVE - MACRO DEFINITION USED AT THE BEGINNING OF A SUBROUTINE TO
1492 ;* SAVE THE GPR CONTENTS FOR LATER RESTORATION.
1493 ;* PASS - MACRO DEFINITION USED AT THE END OF A SUBROUTINE TO RESTORE
1494 ;* THE PREVIOUSLY SAVED GPR CONTENTS AND TO LEAVE THE CONTENTS
1495 ;* OF THE SPECIFIED GPR(S) INTACT (NOT RESTORED).
1496 ;* PREG05 - SUBROUTINE WHICH IS CALLED FROM THE SAVE AND PASS MACRO
1497 ;* EXPANSIONS WHICH ACTUALLY PERFORMS THE ACTIONS ON THE GPRS.
1498 ;*
1499 ;* DURING A SUBROUTINE WHICH USES THESE GPR SAVE ROUTINES THE VALUES
1500 ;* OF THE GPRS ARE STORED ON THE STACK IN THE FOLLOWING STACK FRAME:
1501 ;*
1502 ;* SP -> RET PC INTO PREG05 ROUTINE.
1503 ;* SP+2 -> GPR R0 CONTENTS.
1504 ;* SP+4 -> GPR R1 CONTENTS.
1505 ;* SP+6 -> GPR R2 CONTENTS.
1506 ;* SP+8 -> GPR R3 CONTENTS.
1507 ;* SP+10 -> GPR R4 CONTENTS.
1508 ;* SP+12 -> GPR R5 CONTENTS.
1509 ;* SP+14 -> RET PC INTO CALLER OF SUB'TNE WHICH CALLED PREG05.
1510 ;*
1511 ;* EACH LEVEL OF SUB'TNE CALLING USES 8 WORDS OF STACK OVERHEAD.
1512 ;* THE SAVE AND PASS MACROS CAN ALSO BE USED IN "STRAIGHT LINE CODE"
1513 ;* TO SAVE AND RESTORE THE GPR VALUES. IN ANY CASE, AFTER THE
1514 ;* ISSUING OF A PASS CALL THE GPRS WILL BE RESTORED TO THE VALUES
1515 ;* THEY HAD PRIOR TO THE LAST SAVE CALL (EXCEPT FOR THE EXCEPTED,
1516 ;* OR PASSED INTACT, GPRS SPECIFIED AS PARAMETERS TO THE PASS CALL)
1517 ;* AND THE SP WILL ALSO BE RESTORED TO ITS CONDITION BEFORE THE LAST
1518 ;* SAVE CALL. THE PROGRAMMER MUST BE SURE THAT THE SP HAS THE SAME
1519 ;* VALUE WHEN THE PASS MACRO IS CALLED AS IT HAD IMMEDIATELY AFTER
1520 ;* THE SAVE MACRO WAS CALLED.
1521 ;*****

```

1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537

.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 PREG05
;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	ROSLOT==	2	;OFFSET FOR R0.

1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562

```

.SBTTL GLOBAL MACRO DEFINITION - SAVE -
;*****
;* THIS MACRO IS USED AT THE BEGINNING OF A SUBROUTINE TO SAVE THE
;* CONTENTS OF THE GPRS R0 THRU R5.
;*
;* INPUTS: SP - UNCHANGED SINCE SUBROUTINE WAS ENTERED
;* R5SLOT - OFFSET TO STACK SLOT FOR R5 (EQUATED TO 14 OCTAL)
;*
;* OUTPUTS: GPR SAVE AREA ON THE STACK IS LOADED WITH THE CONTENTS OF GPRS
;* TOP OF STACK - LOADED WITH THE RETURN ADDRESS INTO PREG05
;*
;* CALLING SEQUENCE: SAVE
;*
;* COMMENTS: NO ARGUMENTS ARE ALLOWED.
;* THE PASS MACRO SHOULD BE CALLED TO RESTORE THE GPR VALUES.
;*
;* SUBORDINATE ROUTINES CALLED: PREG05.
;*****
.MACRO SAVE
.LIST
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
.NLIST
.ENDM SAVE

```

1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611

```

.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 PREG05 TO RESTORE THE GPRS TO THEIR SAVED VALUES.
;*
;* INPUTS: ONLY ALLOWED ARGUMENTS ARE "R0" THRU "R5".
;* ROSLOT 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,ROSLOT(SP) ;PUT R0 IN STACK SLOT.
;* MOV R3,R3SLOT(SP) ;PUT R3 IN STACK SLOT.
;* JSR PC,8(SP)+ ;RETURN TO PREG05 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 PREG05, 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
.NLIST
.ENDM
PASS

```

```

1613 .SBTTL GLOBAL SUBROUTINE - PREG05 -
1614 ;*****
1615 ;* PRESERVE REGISTERS R0 THROUGH R5 FOR SUBROUTINE CALLS.
1616 ;*
1617 ;* INPUTS: THE RETURN ADDRESS BACK INTO THE CALLING ROUTINE MUST BE IN
1618 ;* GPR R5. (I.E.- MACROS USE "JSR R5,PREG05".)
1619 ;*
1620 ;* OUTPUTS: REGISTERS R0 THROUGH R5 ARE SAVED ON THE STACK.
1621 ;*
1622 ;*CALLING SEQUENCE: SAVE ;MACRO EXPANSION CALLS PREG05.
1623 ;* [SUBROUTINE CODE]...
1624 ;* PASS ;MACRO EXPANSION RECALLS PREG05.
1625 ;*
1626 ;*COMMENTS: THIS ROUTINE IS RE-ENTRANT.
1627 ;*
1628 ;* PARAMETERS MAY BE PASSED OUT OF A SUBROUTINE BY MODIFYING THE
1629 ;* REGISTER SAVE AREA ON THE STACK. USE THE PASS GPRN MACRO
1630 ;* TO RETURN GPR VALUES INTACT.
1631 ;* USE THE RNSLOT OFFSETS FROM THE SP TO PASS OTHER PARAMETERS.
1632 ;* [EXAMPLE: MOV VALUE,R0SLOT(SP) ]
1633 ;* MAKE SURE THE SP IS AT ITS ORIGINAL VALUE WHEN YOU DO THIS.
1634 ;*
1635 ;*SUBORDINATE ROUTINES CALLED: NONE.
1636 ;*****
1637
1638 004070 PREG05: ;R5 HAS BEEN LOADED ON THE STACK BY THE SUBROUTINE CALL
1639 004070 010446 MOV R4,-(SP) ;SAVE R4
1640 004072 010346 MOV R3,-(SP) ;SAVE R3
1641 004074 010246 MOV R2,-(SP) ;SAVE R2
1642 004076 010146 MOV R1,-(SP) ;SAVE R1
1643 004100 010046 MOV R0,-(SP) ;SAVE R0
1644 004102 010546 MOV R5,-(SP) ;PUSH RETURN PC ON TOP OF STACK
1645 004104 016665 000014 MOV R5SLOT(SP),R5 ;RESTORE R5 TO VALUE IT HAD BEFORE CALLS
1646
1647 004110 004736 JSR PC,B(SP)+ ;CALL THE SUBROUTINE AT THE RETURN ADDRESS
1648 ;FROM THE PREG05 CALL, PUTTING THE PRESENT
1649 ;PC ON THE STACK AS A RETURN ADDRESS INTO
1650 ;THIS (PREG05) ROUTINE.
1651
1652
1653 ;***
1654 ;THE FOLLOWING CODE IS EXECUTED WHEN THE CALLING ROUTINE DOES A
1655 ;"RETURN" [JSR PC,B(SP)+] USING THE PC DEPOSITED ON THE STACK ABOVE.
1656 ;---
1657 004112 012605 PREGRT:: MOV (SP)+,R5 ;PUT RETURN PC IN R5.
1658 004114 012600 MOV (SP)+,R0 ;RESTORE R0.
1659 004116 012601 MOV (SP)+,R1 ;RESTORE R1.
1660 004120 012602 MOV (SP)+,R2 ;RESTORE R2.
1661 004122 012603 MOV (SP)+,R3 ;RESTORE R3.
1662 004124 012604 MOV (SP)+,R4 ;RESTORE R4.
1663
1664 004126 000205 RTS R5 ;RETURN TO THE SUBROUTINE WHICH CALLED PREG05.
1665 ;RESTORING R5 IN THE PROCESS.

```

1667
1669
1670
1671
1672
1673
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686

1687
1693
1694
1695
1696

T2/

1697
1698
1705

```
.SBTTL GLOBAL TEXT SECTION
;*****
;
;           FVTSKL1.P11
;*****
```

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

```
; NAMES OF DEVICES SUPPORTED BY PROGRAM
```

```
;          DEVTYP <DHU-11>
```

```
L#DVTYP::
          .ASCIZ /DHU-11/
          .EVEN
```

```
; TEST DESCRIPTION
```

```
;          DESCRIPT      <DHU-11 FUNC TST PART2>
```

```
L#DESC::
          .ASCIZ /DHU-11 FUNC TST PAR
```

004130			
004130			
004130	104	110	125
004133	055	061	061
004136	000		
004140			
004140			
004140	104	110	125
004143	055	061	061
004146	040	106	125
004151	116	103	040
004154	124	123	124
004157	040	120	101
004162	122	124	062
004165	000		

```
.EVEN
```

```
.EVEN
```

1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1730
1731

;
; FVTA.FMT
;

;
; FORMAT STATEMENTS USED IN PRINT CALLS
;


```

1797 007244 EM6401:: .ASCIZ /BREAK GENERATION TEST FAILED/
1798 007301 EM6402:: .ASCIZ / BREAK NOT RECEIVED ON LINE(S):/
1799 007342 EM6601:: .ASCIZ /NO OVERRUN ERROR TEST FAILED/
1800 007377 EM6602:: .ASCIZ / OVERRUN ERROR REPORTED WHEN NONE FORCED/
1801 007451 EM6701:: .ASCIZ /OVERRUN ERROR TEST FAILED/
1802 007503 EM6702:: .ASCIZ / NO OVERRUN ERROR REPORTED, OVERRUN FORCED/
1803 007560 EM7801:: .ASCIZ /MODEM CONTROL DTR BIT TEST FAILED/
1804 007622 EM7802:: .ASCIZ / DTR BIT FAULTY ON LINE:/
1805 007653 EM7901:: .ASCIZ /MODEM CONTROL RTS BIT TEST FAILED/
1806 007715 EM7902:: .ASCIZ / RTS BIT FAULTY ON LINE:/
1807 007746 EM8001:: .ASCIZ /DSR MODEM STATUS SIGNAL TEST FAILED/
1808 010012 EM8002:: .ASCIZ / DSR MODEM STATUS SIGNAL DEFECTIVE/
1809 010056 EM8101:: .ASCIZ /RI MODEM STATUS SIGNAL TEST FAILED/
1810 010121 EM8102:: .ASCIZ / RI MODEM STATUS SIGNAL DEFECTIVE/
1811 010164 EM8201:: .ASCIZ /CTS MODEM STATUS SIGNAL TEST FAILED/
1812 010230 EM8202:: .ASCIZ / CTS MODEM STATUS SIGNAL DEFECTIVE/
1813 010274 EM8301:: .ASCIZ /DCD MODEM STATUS SIGNAL TEST FAILED/
1814 010340 EM8302:: .ASCIZ / DCD MODEM STATUS SIGNAL DEFECTIVE/
1815 010404 EM8401:: .ASCIZ /DTR MODEM CONTROL SIGNAL INTERACTIONS TEST FAILED/
1816 010466 EM8402:: .ASCIZ /DTR/
1817 010472 EM8403:: .ASCIZ /DSR/
1818 010476 EM8404:: .ASCIZ /RI/
1819 010501 EM8405:: .ASCIZ /DCD/
1820 010505 EM8406:: .ASCIZ /CTS/
1821 010511 EM8501:: .ASCIZ /RTS MODEM CONTROL SIGNAL INTERACTIONS TEST FAILED/
1822 010573 EM8502:: .ASCIZ /RTS/
1823 010577 EM9009:: .ASCIZ /EXPECTED OR CORRECT/
1824 010623 EM9010:: .ASCIZ /ACTUAL OR MEASURED /
1825 010647 EM9017:: .ASCII / FIFO WILL NOT PURGE (DATA.VALID STUCK SET),/
1826 010724 .ASCIZ / REMAINDER OF TEST SKIPPED./
1827 010760 EM9026:: .ASCIZ / LPR CONTENTS: /
1828 011004 EM9104:: .ASCIZ / UNEXPECTED DATA FOUND IN FIFO FROM LINE: /
1829 011060 EM9301:: .ASCIZ /BMP CODE REPORT/
1830 011100 EM9302:: .ASCIZ /BMP CODE FOUND IN TEST /
1831 011130 EM9303:: .ASCIZ /THE LAST BMP CODE WAS FOUND IN TEST /
1832 011175 EM9304:: .ASCIZ /UNEXPECTED BMP CODES FOUND DURING THIS PASS/
1833
1834 .EVEN
1835 .LIST BIN

```


1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877 011252
011252
1878 011252
011252 004567 172612
1879
1880 011256 012700 000100
1881 011262 046700 170732
1882 011266 001036
1883
1884
1885
1886
1887 011270 032705 000001
1888 011274 001410
1889 011276
011276 012746 011370
011302 012746 000001
011306 010600
011310 104414
011312 062706 000004
1890 011316 032705 000002
1891 011322 001410
1892 011324
011324 012746 011446
011330 012746 000001
011334 010600
011336 104414
011340 062706 000004
1893 011344
011344 012746 011525
011350 012746 000001
011354 010600
011356 104415
011360 062706 000004

```

.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). IF THE "EXTENDED ERROR INFO" OPTION HAS BEEN SELECTED
;* THEN THIS SUBROUTINE WILL REPORT THE TYPE OF ACCESS (READ OR WRITE OR
;* BOTH) WHICH CAUSED A BUS TIME-OUT TRAP (004 TRAP). A MESSAGE INDICATING
;* THAT THE DMU MAY BE AT THE WRONG UNIBUS 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.
;*****

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

MOV #BIT06,R0 ;SET-UP THE BIT MAP FOR 'REPORT EXT'D ERROR INFO'
BIC OPTION,R0 ;TRY AND CLEAR THE FLAG.
BNE 6# ;EXIT IF OPTION NOT SELECTED.

;+
; REPORT EXTENDED ERROR INFOMATION
;-

                                BIT #BIT0,R5 ;TEST FOR READ ERROR.
                                BEQ 2# ;SKIP READ ERROR MSG IF NO READ ERROR.
                                PRINTB #MSG1 ;PRINT READ ERROR MESSAGE.
                                MOV #MSG1,-(SP)
                                MOV #1,-(SP)
                                MOV SP,R0
                                TRAP C#PNTB
                                ADD #4,SP

2#:                                BIT #BIT1,R5 ;TEST FOR WRITE ERROR.
                                BEQ 4# ;SKIP WRITE ERROR MSG IF NO WRITE ERROR.
                                PRINTB #MSG2 ;PRINT WRITE ERROR MESSAGE.
                                MOV #MSG2,-(SP)
                                MOV #1,-(SP)
                                MOV SP,R0
                                TRAP C#PNTB
                                ADD #4,SP

4#:                                PRINTX #MSG3 ;SUGGEST THAT DMU MAY BE AT WRONG ADDRESS.
                                MOV #MSG3,-(SP)
                                MOV #1,-(SP)
                                MOV SP,R0
                                TRAP C#PNTX
                                ADD #4,SP

```

```

1894 011364          64:  PASS          ;RESTORE THE GPR CONTENTS.
      011364 004736          JSR          PC,0(SP).          ;RETURN TO PREG05 SUBRT.
1895 011366          ENDMMSG
      011366          L10002:
      011366 104423          TRAP  C#MSG
1896
1897 011370          045  101  102  MSG1:: .ASCIZ /#ABUS TIME-OUT TRAP CAUSED BY READ ATTEMPT.#N/
      011373          125  123  040
      011376          124  111  115
      011401          105  055  117
      011404          125  124  040
      011407          124  122  101
      011412          120  040  103
      011415          101  125  123
      011420          105  104  040
      011423          102  131  040
      011426          122  105  101
      011431          104  040  101
      011434          124  124  105
      011437          115  120  124
      011442          056  045  116
      011445          000
1898 011446          045  101  102  MSG2:: .ASCIZ /#ABUS TIME-OUT TRAP CAUSED BY WRITE ATTEMPT.#N/
      011451          125  123  040
      011454          124  111  115
      011457          105  055  117
      011462          125  124  040
      011465          124  122  101
      011470          120  040  103
      011473          101  125  123
      011476          105  104  040
      011501          102  131  040
      011504          127  122  111
      011507          124  105  040
      011512          101  124  124
      011515          105  115  120
      011520          124  056  045
      011523          116  000
1899 011525          045  101  104  MSG3:: .ASCIZ /#ADHU MAY BE AT THE WRONG UNIBUS ADDRESS.#N#N/
      011530          110  125  040
      011533          115  101  131
      011536          040  102  105
      011541          040  101  124
      011544          040  124  110
      011547          105  040  127
      011552          122  117  116
      011555          107  040  125
      011560          116  111  102
      011563          125  123  040
      011566          101  104  104
      011571          122  105  123
      011574          123  056  045
      011577          116  045  116
      011602          000
1900
1901          .EVEN

```

```

1903 .SBTTL GLOBAL ERROR REPORTING ROUTINE - ER0503 -
1904 ;*****
1905 ;* THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS AN ADDITIONAL ERROR
1906 ;* MESSAGE WHOSE ADDRESS IS PASSED AS AN INPUT PARAMETER, PROVIDED
1907 ;* EXTENDED ERROR REPORTING HAS BEEN REQUESTED.
1908 ;*
1909 ;* INPUTS: R1 - ADDRESS OF THE MESSAGE TO PRINT.
1910 ;*
1911 ;* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
1912 ;*
1913 ;* CALLING SEQUENCE: LOAD THE ADDRESS OF THE MESSAGE IN R1.
1914 ;* INCLUDE THE LABEL "ER0503" AS THE MESSAGE POINTER
1915 ;* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
1916 ;*
1917 ;* COMMENTS: THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
1918 ;*
1919 ;* SUBORDINATE ROUTINES USED: NONE.
1920 ;*****
1921
1922 011604 BGNMSG ER0503
1923 011604 ER0503::
1924 011604 012700 000100 MOV #BIT06,R0 ;TRY TO CLEAR THE
1925 011610 046700 170404 BIC OPTION,R0 ;EXT'D ERROR REPORTING FLAG
1926 011614 001011 BNE 2$ ;EXIT IF FLAG NOT SET.
1927
1928
1929 011616 PRINTB #EF0503,R1 ;PRINT THE MESSAGE.
1930 011616 010146 MOV R1,-(SP)
1931 011620 012746 004217 MOV #EF0503,-(SP)
1932 011624 012746 000002 MOV #2,-(SP)
1933 011630 010600 MOV SP,R0
1934 011632 104414 TRAP C#PNTB
1935 011634 062706 000006 ADD #6,SP
1936
1937 2$: ENDMSG
1938
1939 011640 L10003: TRAP C#MSG
1940 011640 104423

```

1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955 011642
011642
1956 011642 004567 172222
011642
1957
1958 011646 012700 000100
1959 011652 046700 170342
1960 011656 001024
1961
1962
1963 011660
011660 010146
011662 012746 004217
011666 012746 000002
011672 010600
011674 104414
011676 062706 000006
1964
1965 011702 016702 172156
1966 011706
011706 010246
011710 012746 004277
011714 012746 000002
011720 010600
011722 104414
011724 062706 000006
1967
1968 011730
011730 004736
1969 011732
011732
011732 104423

```

.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, PROVIDED EXTENDED ERROR INFORMATION HAS BEEN
;* REQUESTED, OTHERWISE ONLY A "TEST FAILURE" MESSAGE WILL BE PRINTED.
;*
;* 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 LABEL "ER1603" AS THE MESSAGE POINTER
;* PARAMETER IN THE DRS ERROR REPORT MACRO CALL.
;*
;* COMMENTS:
;*
;* SUBORDINATE ROUTINES CALLED: NONE.
;*****
BGNMSG ER1603
ER1603::
SAVE ;SAVE THE CONTENTS OF THE GPRS.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.

MOV #BIT06,R0 ;TRY TO CLEAR THE
BIC OPTION,R0 ;EXT'D ERROR REPORTING FLAG
BNE 2# ;EXIT IF FLAG NOT SET.

PRINTB #EF0503,R1 ;PRINT BASIC MESSAGE ON OPERATORS CONSOLE.
MOV R1,-(SP)
MOV #EF0503,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C#PNTB
ADD #6,SP

MOV ERRMSG,R2 ;GET THE "TEST MESSAGE".
PRINTB #EF1601,R2 ;PRINT "TEST ABORTED" MESSAGE.
MOV R2,-(SP)
MOV #EF1601,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C#PNTB
ADD #6,SP

2#: PASS ;RESTORE THE CONTENTS OF THE GPRS.
JSR PC,8(SP)+ ;RETURN TO PREG05 SUBRT.

L10004:
TRAP C#MSG

```

1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998 011734
011734
1999 011734
011734 004567 172130
2000
2001
2002
2003
2004 011740 032767 000100 170252
2005 011746 001433
2006
2007
2008 011750 005002
2009 011752 012703 000020
2010 011756
011756 010146
011760 012746 004217
011764 012746 000002
011770 010600
011772 104414
011774 062706 000006
2011 012000 000241
2012 012022 006205
2013 012004 103011
2014 012006
012006 010246
012010 012746 004447
012014 012746 000002
012020 010600
012022 104414

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER6401 -
*****
;* THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ADDITIONAL ERROR
;* INFORMATION AFTER THE ERROR MESSAGE HEADER, PROVIDED EXTENDED ERROR
;* REPORTING HAS BEEN ENABLED.
;* THIS SUBROUTINE IS PASSED A GPR CONTAINING FLAGS WHICH INDICATE
;* THE LINE(S) FOR WHICH THE ERROR CONDITION SHOULD BE REPORTED.
;*
;* INPUTS: R1 - ADDRESS OF THE MESSAGE TO BE PRINTED BY THIS ROUTINE.
;* R5 - CONTAINS THE ERROR FLAGS, (1 FLAG PER LINE).
;*
;* OUTPUTS: MESSAGES ARE PRINTED AT THE OPERATOR CONSOLE.
;*
;* CALLING SEQUENCE: LOAD THE ADDRESS OF THE MESSAGE IN R1.
;* INCLUDE THE LABEL "ER6401" AS THE MESSAGE POINTER
;* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
;*
;* COMMENTS: THE OUTPUT FORMAT OF THIS MESSAGE IS:
;* TEXT MESSAGE
;* #NN
;* #NN
;*
;* WHERE EACH "#NN" IS THE NUMBER OF A LINE WITH THE ERROR.
;*
;* SUBORDINATE ROUTINES USED: NONE.
*****
```

```
BGNMSG ER6401
SAVE ER6401::
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.

;* EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;*
;* BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
;* BEQ 60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
;* DURING THE SOFTWARE QUESTIONS.

CLR R2 ;CLEAR LINE NUMBER TO ZERO.
MOV #NUMLNS,R3 ;SET UP MAX LINE COUNT.
PRINTB #EF0503,R1 ;PRINT MESSAGE.

MOV R1,-(SP)
MOV #EF0503,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C:PNTB
ADD #6,SP

2# CLC ;CLEAR CARRY.
ASR R5 ;SHIFT FLAG OUT INTO CARRY BIT.
BCC 4# ;SKIP ERROR REPORT IF CLEAR.
PRINTB #EF6401,R2 ;PRINT MESSAGE.

MOV R2,-(SP)
MOV #EF6401,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C:PNTB
```



```

2021 .SBTTL GLOBAL ERROR REPORTING ROUTINE - ER7801 -
2022 ;*****
2023 ;* THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS AN ADDITIONAL ERROR
2024 ;* MESSAGE WHOSE ADDRESS IS PASSED AS AN INPUT PARAMETER. A LINE NUMBER
2025 ;* IS INCLUDED AT THE END OF THE MESSAGE. THE MESSAGE IS PRINTED ONLY IF
2026 ;* EXTENDED ERROR REPORTING IS REQUESTED.
2027 ;*
2028 ;* INPUTS: R1 - ADDRESS OF THE MESSAGE TO PRINT.
2029 ;* R3 - NUMBER OF LINE ON WHICH ERROR OCCURRED.
2030 ;*
2031 ;* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
2032 ;*
2033 ;* CALLING SEQUENCE: LOAD THE ADDRESS OF THE MESSAGE IN R1.
2034 ;* LOAD THE LINE NUMBER INTO R3.
2035 ;* INCLUDE THE LABEL "ER7801" AS THE MESSAGE POINTER
2036 ;* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
2037 ;*
2038 ;* COMMENTS: THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
2039 ;*
2040 ;* SUBORDINATE ROUTINES USED: NONE.
2041 ;*****
2042
2043 012042 BGNMSG ER7801
2044 012042 ER7801::
2045
2046 ;*
2047 ;* EXIT IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
2048 012042 032767 000100 170150 BIT #BIT06.OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
2049 012050 001412 BEQ 2# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
2050 ;* DURING THE SOFTWARE QUESTIONS.
2051
2052 012052 PRINTB #EF7801,R1,R3 ;PRINT THE MESSAGE.
2053 012052 010346 MOV R3,-(SP)
2054 012054 010146 MOV R1,-(SP)
2055 012056 012746 004516 MOV #EF7801,-(SP)
2056 012062 012746 000003 MOV #3,-(SP)
2057 012066 010600 MOV SP,R0
2058 012070 104414 TRAP C#PNTB
2059 012072 062706 000010 ADD #10,SP
2060
2061 2# : ENDMSG
2062
2063 L10006: TRAP C#MSG

```

2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084 012100
012100
2085 012100
012100 004567 171764
2086
2087
2088
2089
2090 012104 032767 000100 170106
2091 012112 001517
2092
2093
2094 012114
012114 010346
012116 010146
012120 012746 004554
012124 012746 000003
012130 010600
012132 104414
012134 062706 000010
2095
2096 012140 010167 000204
2097 012144 005001
2098 012146 012704 002620
2099 012152 010177 170060
2100 012156 017700 170062
2101 012162 011405
2102 012164 040005
2103 012166 042400

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER8401 -
;*****
;* THIS ERROR REPORTING SUBROUTINE IS INTENDED TO REPORT INTERACTIONS
;* WHICH HAVE BEEN FOUND BETWEEN A MODEM SIGNAL AND OTHER MODEM SIGNALS.
;* IT ANALYZES THE MODEM STATUS WHICH IS STORED IN THE STAT STORAGE AREA
;* AND REPORTS ANY DISCREPANCIES WHICH ARE FOUND BETWEEN THIS STORED DATA
;* AND THE PRESENT STATE OF THE STAT REGISTERS. SPECIFIED BITS ON THE
;* LINE ASSOCIATED WITH THE SPECIFIED LINE ARE IGNORED.
;*
;* INPUTS: R1 - ADDRESS OF SIGNAL NAME MESSAGE.
;* R2 - BIT MAP OF BITS TO IGNORE ON SPECIFIED LINE.
;* R3 - NUMBER OF SPECIFIED LINE.
;* CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
;* NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
;* FLSA - 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: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
;*
;* CALLING SEQUENCE: INCLUDE THE LABEL "ER8401" 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 ER8401
SAVE JSR ER8401::;PRESERVE THE CONTENTS OF THE GPRS.
R5,PREGOS ;CALL REGISTER SAVE SUBRT.

;* EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;*-
BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
BEQ 601 ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
;DURING THE SOFTWARE QUESTIONS.

PRINTB #EF8401,R1,R3 ;PRINT THE BASIC MESSAGE.

MOV R3,-(SP)
MOV R1,-(SP)
MOV #EF8401,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C:PNTB
ADD #10,SP

MOV R1,448 ;SAVE THE ADDRESS OF THE SIGNAL NAME MESSAGE.
CLR R1 ;CLEAR THE LINE COUNTER.
MOV #STSTB,R4 ;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
20: MOV R1,BCSRA ;SET UP THE CSR IND.ADR.REG FIELD.
MOV #FSLSA,R0 ;GET THE CONTENTS OF THIS LINE'S STAT REGISTER.
MOV (R4),R5 ;GET THE PREVIOUS CONTENTS FROM STORAGE.
BIC R0,R5
BIC (R4),R0
```

```

2104 012170 050005          BIS    R0,R5          ;XOR PRESENT AND STORED STAT VALUES.
2105 012172 012700 043777  MOV    #43777,R0     ;PREPARE TO MASK OUT UNUSED BITS.
2106 012176 120163 004020  CMPB  R1, TXRLNB(R3) ;IS THIS LINE ASSOCIATED WITH SPECIFIED LINE?
2107 012202 001002          BNE    4#            ;DON'T MASK OUT SPECIFIED BITS IF IT IS NOT.
2108 012204 056600 000006          BIS    R2SLOT(SP),R0 ;MASK OUT SPECIFIED BITS.
2109 012210 040005 4#:      BIC    R0,R5          ;GET BIT MAP OF UNDESIRED CHANGES.
2110 012212 032705 100000          BIT    #BIT15,R5     ;CHECK FOR DSR SIGNAL INTERACTION.
2111 012216 001404          BEQ    6#            ;SKIP PRINTING LINE IF NO DSR INTERACTION.
2112 012220 012702 010472  MOV    #EM8403,R2    ;SELECT DSR ERROR MESSAGE.
2113 012224 004767 000064  JSR    PC,40#        ;PRINT THE LINE OF THE ERROR MESSAGE.
2114 012230 032705 020000 6#:      BIT    #BIT13,R5 ;CHECK FOR RI SIGNAL INTERACTION.
2115 012234 001404          BEQ    8#            ;SKIP PRINTING LINE IF NO RI INTERACTION.
2116 012236 012702 010476  MOV    #EM8404,R2    ;SELECT RI ERROR MESSAGE.
2117 012242 004767 000046  JSR    PC,40#        ;PRINT THE LINE OF THE ERROR MESSAGE.
2118 012246 032705 010000 8#:      BIT    #BIT12,R5 ;CHECK FOR DCD SIGNAL INTERACTION.
2119 012252 001404          BEQ    10#           ;SKIP PRINTING LINE IF NO DCD INTERACTION.
2120 012254 012702 010501  MOV    #EM8405,R2    ;SELECT DCD ERROR MESSAGE.
2121 012260 004767 000030  JSR    PC,40#        ;PRINT THE LINE OF THE ERROR MESSAGE.
2122 012264 032705 004000 10#:     BIT    #BIT11,R5    ;CHECK FOR CTS SIGNAL INTERACTION.
2123 012270 001404          BEQ    12#           ;SKIP PRINTING LINE IF NO CTS INTERACTION.
2124 012272 012702 010505  MOV    #EM8406,R2    ;SELECT CTS ERROR MESSAGE.
2125 012276 004767 000012  JSR    PC,40#        ;PRINT THE LINE OF THE ERROR MESSAGE.
2126
2127 012302 005201 12#:     INC    R1            ;SELECT NEXT LINE.
2128 012304 020127 000020  CMP    R1,#NUMLNS    ;ALL LINES DONE?
2129 012310 002720          BLT    2#            ;LOOP IF NOT ALL LINES DONE.
2130 012312 000417          BR    60#           ;EXIT THIS ROUTINE.
2131
2132          ;* LOCAL ERROR MESSAGE LINE PRINTING ROUTINE.
2133          ;-
2134 012314 40#:     PRINTX #EF8402,44#,R3,R2,R1
          012314 010146          MOV    R1,-(SP)
          012316 010246          MOV    R2,-(SP)
          012320 010346          MOV    R3,-(SP)
          012322 016746 000022  MOV    44#,-(SP)
          012326 012746 004646  MOV    #EF8402,-(SP)
          012332 012746 000005  MOV    #5,-(SP)
          012336 010600          MOV    SP,R0
          012340 104415          TRAP  C#PNTX
          012342 062706 000014  ADD    #14,SP
2135 012346 000207          RTS    PC
2136 012350 000000 44#:     .WORD 0
2137 012352 004736 60#:     PASS          ;LOCAL STORAGE FOR ADDRESS OF SIGNAL NAME.
          ;RESTORE ALL THE GPRS TO THE PRESERVED VALUES.
          ;RETURN TO PREG05 SUBRT.
          JSR    PC,@(SP)
2138 012354          ENDMSG
          L10007: TRAP  C#MSG
          012354 104423

```

```

2140 .SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9001 -
2141 ;*****
2142 ;* THIS IS AN ERROR REPORTING SUBROUTINE WHICH REPORTS AN UNEXPECTED
2143 ;* CODE WHICH HAS BEEN FOUND IN THE DUT CSR. THIS CODE CAN BE A BMP
2144 ;* CODE, A SELF-TEST CODE, OR A MODEM STATUS CODE.
2145 ;*
2146 ;* INPUTS: R1 - ADDRESS OF MESSAGE TO PRINT FIRST.
2147 ;* R2 - SINGLE BYTE CODE WHICH HAS BEEN READ FROM THE DUT.
2148 ;* R4 - LINE NUMBER ASSOCIATED WITH THE CODE.
2149 ;*
2150 ;* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
2151 ;*
2152 ;* CALLING SEQUENCE: INCLUDE THE LABEL "ER9001" AS THE MESSAGE POINTER
2153 ;* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
2154 ;*
2155 ;* COMMENTS: THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
2156 ;*
2157 ;* SUBORDINATE ROUTINES USED: NONE.
2158 ;*****
2159
2160 012356 BGNMSG ER9001
2161 012356 ER9001::
2162
2163 ;* EXIT IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
2164 ;*
2165 012356 032767 000100 167634 BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
2166 012364 001433 BEQ 2# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
2167 ;* DURING THE SOFTWARE QUESTIONS.
2168
2169 012366 PRINTB #EF9001,R1 ;REPORT TYPE OF CODE FOUND.
2170 012366 010146 MOV R1,-(SP)
2170 012370 012746 004763 MOV #EF9001,-(SP)
2170 012374 012746 000002 MOV #2,-(SP)
2170 012400 010600 MOV SP,R0
2170 012402 104414 TRAP C#PNTB
2170 012404 062706 000006 ADD #6,SP
2171 012410 PRINTX #EF9002,R4 ;REPORT THE LINE NUMBER OF THE CODE.
2171 012410 010446 MOV R4,-(SP)
2171 012412 012746 005045 MOV #EF9002,-(SP)
2171 012416 012746 000002 MOV #2,-(SP)
2171 012422 010600 MOV SP,R0
2171 012424 104415 TRAP C#PNTX
2171 012426 062706 000006 ADD #6,SP
2172 012432 PRINTX #EF9003,R2 ;REPORT THE CODE WHICH WAS FOUND.
2172 012432 010246 MOV R2,-(SP)
2172 012434 012746 005117 MOV #EF9003,-(SP)
2172 012440 012746 000002 MOV #2,-(SP)
2172 012444 010600 MOV SP,R0
2172 012446 104415 TRAP C#PNTX
2172 012450 062706 000006 ADD #6,SP
2173 012454 2# : ENDMSG
2173 012454 L10010:
2173 012454 104423 TRAP C#MSG

```

2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197 012456
012456
2198
2199
2200
2201
2202 012456 032767 000100 167534
2203 012464 001462
2204
2205
2206 012466 006203
2207 012470 042702 177400
2208 012474
012474 010346
012476 010146
012500 012746 005227
012504 012746 000003
012510 010600
012512 104414
012514 062706 000010
2209 012520
012520 010246
012522 012746 010623
012526 012746 005146
012532 012746 000003
012536 010600
012540 104415
012542 062706 000010
2210 012546 005704
2211 012550 100414
2212 012552
012552 010446
012554 012746 010577
012560 012746 005146
012564 012746 000003

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9002 -
;*****
;* THIS IS AN ERROR REPORTING SUBROUTINE WHICH IS INTENDED FOR USE IN THE
;* TRANSMISSION 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

ER9002::

;- EXIT IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED

```
BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
BEQ 62# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
;DURING THE SOFTWARE QUESTIONS.
```

```
ASR R3 ;CALCULATE THE LINE NUMBER.
BIC #177400,R2 ;MASK OUT ALL BUT DATA IN READ CHAR.
PRINTB #EF9006,R1,R3 ;PRINT THE FIRST LINE OF THE MESSAGE.
```

```
MOV R3,-(SP)
MOV R1,-(SP)
MOV #EF9006,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C#PNTB
ADD #10,SP
```

PRINTX #EF9004,#EM9010,R2 ;PRINT ACTUAL DATA.

```
MOV R2,-(SP)
MOV #EM9010,-(SP)
MOV #EF9004,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C#PNTX
ADD #10,SP
```

```
TST R4 ;CHECK FOR "NONE" CODE SET IN EXPECTED DATA.
BHI 2# ;BRANCH TO PRINT "NONE" MESSAGE IF FLAG SET.
PRINTX #EF9004,#EM9009,R4 ;PRINT EXPECTED DATA.
```

```
MOV R4,-(SP)
MOV #EM9009,-(SP)
MOV #EF9004,-(SP)
MOV #3,-(SP)
```

```

012570 010600
012572 104415
012574 062706 000010
2213 012600 000412
2214 012602
012602 012746 010577
012606 012746 005176
012612 012746 000002
012616 010600
012620 104415
012622 062706 000006
2215 012626 004767 002046
2216 012632
012632
012632 104423

21: BR 60: ;EXIT THIS ROUTINE.
PRINTX #EF9005,#EM9009 ;PRINT MESSAGE INDICATING NO EXPECTED DATA.

60: JSR PC,PRTLPR ;PRINT CONTENTS OF THE LPR REGISTER.
62: ENDMSG

L10011: TRAP C#MSG
MOV SP,RO
TRAP C#PNTX
ADD #10,SP
MOV #EM9009,-(SP)
MOV #EF9005,-(SP)
MOV #2,-(SP)
MOV SP,RO
TRAP C#PNTX
ADD #6,SP

```

```

2218 .SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9101 -
2219 ;*****
2220 ;* THIS IS A GENERAL ERROR REPORTING SUBROUTINE WHICH REPORTS A MESSAGE
2221 ;* WHICH TAKES A SINGLE, 2 DIGIT DECIMAL ARGUMENT AFTER THE END OF AN
2222 ;* ASCII MESSAGE.
2223 ;*
2224 ;* INPUTS: R1 - VALUE TO BE PRINTED AFTER MSG AS 2 DECIMAL DIGITS.
2225 ;* R2 - ADDRESS OF MESSAGE TO PRINT FIRST.
2226 ;*
2227 ;* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
2228 ;*
2229 ;* CALLING SEQUENCE: INCLUDE THE LABEL "ER9101" AS THE MESSAGE POINTER
2230 ;* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
2231 ;*
2232 ;* COMMENTS: THE MESSAGE IS PRINTED AS BASIC ERROR INFORMATION.
2233 ;*
2234 ;* SUBORDINATE ROUTINES USED: NONE.
2235 ;*****
2236
2237 012634 BGNMSG ER9101
012634 ER9101::
2238
2239 012634 012700 000100 MOV #BIT06,R0 ;TRY TO CLEAR THE
2240 012640 046700 167354 BIC OPTION,R0 ;EXT'D ERROR REPORTING FLAG
2241 012644 001012 BNE 2# ;EXIT IF FLAG NOT SET.
2242
2243
2244 012646 PRINTB #EF9006,R2,R1 ;REPORT THE STRING FOLLOWED BY THE NUMBER.
012646 010146 MOV R1,-(SP)
012650 010246 MOV R2,-(SP)
012652 012746 005227 MOV #EF9006,-(SP)
012656 012746 000003 MOV #3,-(SP)
012662 010600 MOV SP,R0
012664 104414 TRAP C#PNTB
012666 062706 000010 ADD #10,SP
2245
2246 012672 2#: ENDMSG
012672
012672 104423 L10012: TRAP C#MSG

```

2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269 012674
012674
2270 012674 004567 171170
012674
2271
2272 012700 012700 000100
2273 012704 046700 167310
2274 012710 001064
2275
2276 012712
012712 010146
012714 012746 004217
012720 012746 000002
012724 010600
012726 104414
012730 062706 000006
2277 012734 012703 002420
2278 012740 012705 011100
2279 012744 012301
2280 012746 012304
2281 012750 004767 000056
2282 012754 020302
2283 012756 103772
2284
2285
2286
2287
2288
2289
2290 012760 020227 002614
2291 012764 001036
2292 012766 005762 000002
2293 012772 001433
2294 012774 012301
2295 012776 011304
2296 013000 012705 011130

```

.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.
; PROVIDED EXTENDED ERROR REPORTING HAS BEEN ENABLED.
;
; 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::
2270             SAVE                ;SAVE THE GPRS ON THE STACK.
                JSR R5,PREG05      ;CALL REGISTER SAVE SUBRT.
                MOV #BIT06,R0      ;TRY TO CLEAR THE
                BIC OPTION,R0      ;EXT'D ERROR REPORTING FLAG
                BNE 60#            ;EXIT IF FLAG NOT SET.
2276             PRINTB #EF0503,R1 ;REPORT UNEXPECTED BMP CODES FOUND.
                MOV R1,-(SP)
                MOV #EF0503,-(SP)
                MOV #2,-(SP)
                MOV SP,R0
                TRAP C:PNTB
                ADD #6,SP
2277             MOV #BMPQCB,R3     ;GET THE START ADDRESS OF THE BMP CODE QUEUE.
                MOV #EM9302,R5     ;GET THE MESSAGE TO BE REPORTED.
2279             MOV (R3)+,R1      ;GET THE NUMBER OF THE TEST THAT WAS EXECUTING.
2280             MOV (R3)+,R4      ;GET BMP CODE THAT WAS REPORTED OFF THE QUEUE.
                JSR PC,50#         ;GO REPORT THE BMP CODE.
                CMP R3,R2         ;CHECK IF ALL CODES HAVE BEEN REPORTED.
                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.
;
;
2290             CMP R2,#BMPQCB-4 ;CHECK IF THE POINTER IS AT THE LAST LOCATION.
                BNE 60#           ;EXIT IF NOT AT THE LAST LOCATION.
                TST 2(R2)        ;CHECK FOR A BMP CODE IN THE LAST CELL
                BEQ 60#           ;EXIT IF NO OVERFLOW HAS OCCURED, CELL EMPTY.
                MOV (R3)+,R1     ;GET THE TEST NUMBER OFF THE QUEUE.
                MOV (R3),R4      ;GET THE BMP CODE OFF THE QUEUE.
                MOV #EM9303,R5   ;SELECT THE MESSAGE TO BE REPORTED.

```

```

2297 013004          PRINTX  #EF9302          ;REPORT OVERFLOW CONDITION.
      013004 012746 005343
      013010 012746 000001
      013014 010600
      013016 104415
      013020 062706 000004
2298 013024 004767 000002
2299 013030 000414
2300
2301 013032          504: PRINTX  #EF9301,R5,R1,R4 ;PRINT THE MESSAGE.
      013032 010446
      013034 010146
      013036 010546
      013040 012746 005265
      013044 012746 000004
      013050 010600
      013052 104415
      013054 062706 000012
2302 013060 000207
2303 013062          604: RTS      PC          ;RETURN.
      013062 004736          PASS          ;RESTORE THE GPR CONTENTS.
2304
2305 013064          JSR      PC,B(SP)+    ;RETURN TO PREG05 SUBRT.
      013064
      013064 104423          ENDMSG
2305 013064          L10013: TRAP  C#MSG

```

2307
2309
2310
2311
2312
2313
2315
2316
2317
2318
2319
2320

```
.SBTTL GLOBAL SUBROUTINES SECTION  
:*****  
:  
: FVTSKL3.P11  
:*****  
  
: **  
: THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES  
: THAT ARE USED IN MORE THAN ONE TEST.  
: --
```

```

2322 .SBTTL GLOBAL SUBROUTINE - ALTFLD -
2323 ;* *****
2324 ;* - ALTER DEVICE REGISTER FIELDS ROUTINE -
2325 ;* THIS SUBROUTINE ALTERS THE SPECIFIED FIELD OF THE SPECIFIED DEVICE
2326 ;* REGISTER FOR THE SPECIFIED LINES. THIS ROUTINE CAN BE USED TO SET
2327 ;* OR CLEAR BITS WITHIN SELECTED FIELDS OF SELECTED REGISTERS.
2328 ;* USE EXAMPLES: SET RX.BAUD.RATE FIELDS ON LINES 3 AND 6.
2329 ;* CLEAR TX.DMA BITS ON ALL LINES.
2330 ;*
2331 ;* INPUTS: R1 - ADDRESS OF THE REGISTERS TO ALTER.
2332 ;* R2 - BIT FIELDS SET TO DESIRED STATES.
2333 ;* R3 - BIT MAP OF LINES FOR WHICH TO ALTER REGISTER.
2334 ;* R4 - MASK OF BITS TO ALTER (1 INDICATES CHANGE BIT).
2335 ;* CSRA - CONTAINS THE ADDRESS OF THE DEVICE CSR.
2336 ;* IESTAT - SAVED STATES OF THE INTERRUPT ENABLE BITS.
2337 ;*
2338 ;* OUTPUTS: DEVICE REGISTERS - SPECIFIED REGISTER FIELDS ALTERED.
2339 ;* CSR IND.ADR.REG FIELD - DESTROYED.
2340 ;*
2341 ;* CALLING SEQUENCE: JSR PC,ALTFLD
2342 ;*
2343 ;* COMMENTS: THIS ROUTINE READS THE SPECIFIED REGISTERS FOR ALL LINES
2344 ;* WITH NUMBERS LOWER THAN THE HIGHEST SPECIFIED LINE.
2345 ;* THIS ROUTINE DOES NOT READ THE CSR.
2346 ;*
2347 ;* SUBROUTINES CALLED: NONE.
2348 ;*
2349 ;*
2350 013066 004567 170776 ALTFLD:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
2351 ; JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2352 ;
2353 ; SET UP TO LOOP FOR EACH LINE:
2354 ; PREPARE THE WORD TO BE ORED INTO THE REGISTER CONTENTS.
2355 ; SET UP THE WORD TO WRITE INTO THE IND.ADR.REG FIELD OF THE CSR.
2356 ;
2357 013072 010400 ; MOV R4,R0 ;CALCULATE THE NEW CONTENTS OF THE
2358 013074 005100 ; COM R0 ; REGISTER FIELDS WHICH ARE TO BE
2359 013076 040002 ; BIC R0,R2 ; ALTERED BY THIS ROUTINE.
2360 013100 016705 167160 ; MOV IESTAT,R5 ;SET UP TO WRITE IND.ADR.REG FIELD TO 0.
2361 ;
2362 ; LOOP ONCE FOR EACH LINE, ALTERING THE SPECIFIED FIELD IN THE SPECIFIED
2363 ; REGISTER IF THE LINE HAS BEEN SELECTED FOR ALTERING.
2364 ; EXIT THE LOOP IF NO MORE LINES TO ALTER, OR IF WE HAVE ALTERED THE MAX
2365 ; ALLOWABLE NUMBER OF LINES (AS SPECIFIED BY NUMLNS).
2366 ;
2367 013104 000241 ; CLC ;PREPARE FOR ROTATE, "TST R5" DOES THIS BELOW.
2368 013106 006003 20: ; ROR R3 ;GET THE LINE SELECT BIT FOR THIS LINE.
2369 013110 103006 ; BCC 40 ;SKIP SETUP IF LINE IS NOT SELECTED.
2370 013112 010577 167120 ; MOV R5,BCSRA ;SET OUT CSR IND.ADR.REG FIELD TO THIS LINE.
2371 013116 011100 ; MOV (R1),R0 ;GET THE PRESENT CONTENTS OF THE REG TO ALTER.
2372 013120 040400 ; BIC R4,R0 ;CLEAR THE BIT FIELDS WE ARE TO ALTER.
2373 013122 050200 ; BIS R2,R0 ;OR IN THE NEW STATES OF THE FIELDS.
2374 013124 010011 ; MOV R0,(R1) ;WRITE THE NEW REGISTER CONTENTS TO THE REG.
2375 013126 005205 40: ; INC R5 ;SET LINE NUMBER TO THE NEXT LINE.
2376 013130 005703 ; TST R3 ;CHECK FOR UNHANDLED LINES, CLEAR CARRY FLAG.
2377 013132 001365 ; BNE 20 ;LOOP IF SELECTED LINE(S) IS NOT HANDLED.
    
```

```
2378  
2379 013134          604:  PASS          ;RESTORE GPRS.  
      013134 004736          PC,@(SP)+ ;RETURN TO PREGOS SUBRT.  
2380 013136 000207          RTS    PC    JSR    ;RETURN TO CALLING ROUTNE.
```

```

2382 .SBTTL GLOBAL SUBROUTINE - ASLNTL -
2383 ;* *****
2384 ;* - SETUP ASSOCIATED LINE NUMBER TABLES ROUTINE -
2385 ;* THIS ROUTINE SETS UP THE TWO TABLES WHICH ARE CONTAIN INFORMATION
2386 ;* ABOUT THE TX/RX LINE WHICH IS ASSOCIATED WITH A PARTICULAR RX/TX
2387 ;* LINE. ONE TABLE IS A TABLE OF WORDS WHICH CONTAINS WORD OFFSET
2388 ;* VALUES AND THE OTHER TABLE IS A TABLE OF BYTES WHICH CONTAINS
2389 ;* LINE NUMBER VALUES.
2390 ;*
2391 ;* INPUTS: LOPBCK - STORAGE FOR THE TYPE OF LOOPBACK ON THE DUT.
2392 ;* NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
2393 ;* STGTRB - LABEL AT BASE OF STAGGERED LINE ASSOCIATION TBL.
2394 ;* TXRLNB - LABEL AT BASE OF BYTE TX/RX LINE NUMBER TABLE.
2395 ;* TXRXLB - LABEL AT BASE OF WORD TX/RX LINE NUMBER TABLE.
2396 ;* TXRXLE - LABEL AT END OF WORD TX/RX LINE NUMBER TABLE.
2397 ;*
2398 ;* OUTPUTS: TXRXL, TXRLN - TABLES INITIALIZED FOR SELECTED LOOPBACK.
2399 ;*
2400 ;* CALLING SEQUENCE: JSR PC,ASLNTL
2401 ;*
2402 ;* COMMENTS:
2403 ;*
2404 ;* SUBORDINATE ROUTINES CALLED: NONE.
2405 ;* -- *****
2406
2407 013140 ASLNTL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
013140 004567 170724 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2408 013144 126727 167062 000002 CMPB LOPBCK,#2 ;TEST FOR STAGGERED LOOPBACK.
2409 013152 001411 BEQ #4 ;GO SET UP STAGGERED TABLE IF STAGGERED LPBCK.
2410 ;*
2411 ;* SET UP THE WORD TABLE FOR NON-STAGGERED LOOPBACK.
2412 ;*
2413 013154 005005 CLR R5 ;CLEAR THE LINE COUNTER
2414 013156 010565 003760 MOV R5, TXRXLB(R5) ;SET UP A WORD OF THE TABLE.
2415 013162 005205 INC R5
2416 013164 005205 INC R5 ;SET LINE COUNTER TO NEXT LINE OFFSET.
2417 013166 020527 000040 CMP R5,#2*NUMLNS ;TEST FOR ALL LINES DONE.
2418 013172 002771 BLT #2 ;LOOP UNTIL ALL LINES DONE.
2419 013174 000411 BR #8 ;GO SET UP THE BYTE TABLE.
2420 ;*
2421 ;* SET UP THE WORD TABLE FOR STAGGERED LOOPBACK.
2422 ;*
2423 013176 012701 004040 4# : MOV #STGTRB,R1 ;SET UP THE SOURCE POINTER.
2424 013202 012702 003760 MOV #TXRXLB,R2 ;SET UP THE DESTINATION POINTER.
2425 013206 112122 6# : MOVB (R1)+,(R2)+ ;MOVE A BYTE INTO THE TABLE.
2426 013210 105022 CLR# (R2)+ ;CLEAR THE UPPER BYTE OF WORD TABLE ENTRY.
2427 013212 020227 004020 CMP R2,#TXRXLE ;COMPARE POINTER WITH END ADR OF TABLE.
2428 013216 002773 BLT #6 ;LOOP IF NOT AT END YET.
2429 ;*
2430 ;* SET UP THE BYTE TABLE BASED ON THE WORD ASSOCIATION TABLE.
2431 ;*
2432 013220 012701 003760 8# : MOV #TXRXLB,R1 ;SET UP THE SOURCE POINTER.
2433 013224 012702 004020 MOV #TXRLNB,R2 ;SET UP THE DESTINATION POINTER.
2434 013230 012103 10# : MOV (R1)+,R3 ;GET THE WORD OFFSET VALUE FROM WORD TABLE.
2435 013232 006203 ASR R3 ;DIVIDE BY 2 TO GET LINE NUMBER VALUE.
2436 013234 110322 MOVB R3,(R2)+ ;LOAD THE BYTE LINE NUMBER INTO TABLE.
2437 013236 020127 004020 CMP R1,#TXRXLE ;COMPARE SOURCE POINTER WITH ADR OF TABLE END.

```

```
2438 013242 002772      BLT      10$      ;LOOP IF NOT AT END OF TABLE YET.  
2439  
2440 013244      60$:  PASS      JSR      ;RESTORE GPRS.  
      013244 004736      PC,0(SP)+  
2441 013246 000207      RTS      PC      ;RETURN TO PREG05 SUBRT.
```

```

2443 .SBTTL GLOBAL SUBROUTINE - CALMSL -
2444 ;* *****
2445 ;* - CALIBRATE MILLI SECOND LOOP COUNT SUBROUTINE -
2446 ;* THIS SUBROUTINE CALIBRATES THE TIMING LOOP WHICH IS USED IN THE MSLOOP
2447 ;* ROUTINE. THIS SUBROUTINE CALCULATES A VALUE FOR THE MSLCNT VARIABLE
2448 ;* WHICH IS THE NUMBER OF SOFTWARE LOOPS WHICH TAKES 1 MS TO EXECUTE IN
2449 ;* THE MSLOOP ROUTINE. THIS ROUTINE CALIBRATES THE COUNT BY USING THE
2450 ;* LINE TIME CLOCK (LTC), SO IF NO LTC IS AVAILABLE THE DEFAULT VALUE FOR
2451 ;* THE DELAY COUNT MUST BE USED.
2452 ;*
2453 ;*
2454 ;* INPUTS: MSLCNT - DEFAULT 1 MS DELAY LOOP COUNT VALUE, OR
2455 ;* VALUE FROM PREVIOUS CALIBRATION.
2456 ;* MSTICK - NUMBER OF MS PER LTC CLOCK TICK.
2457 ;* TIMER1 - TIMER COUNTER CHANGED BY LTC INTERRUPT SERVICE RTN.
2458 ;* CLKHRZ - NUMBER OF LTC CLICKS PER SECOND (50 OR 60).
2459 ;*
2460 ;* OUTPUTS: CARRY - SET IF LTC IS AVAILABLE, AND NEW CALIBRATION PERFORMED.
2461 ;* MSLCNT - NEW 1 MS DELAY LOOP COUNT VALUE IF LTC AVAILABLE, OR
2462 ;* UNCHANGED IF NO LTC IS AVAILABLE.
2463 ;*
2464 ;* CALLING SEQUENCE: JSR PC,CALMSL
2465 ;*
2466 ;* COMMENTS:
2467 ;*
2468 ;* SUBORDINATE ROUTINES CALLED: UNSDIV,OOPS.
2469 ;*-- *****
2470
2471 013250 CALMSL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
2472 013250 004567 170614 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2473 013254 005067 000210 CLR 62# ;CLEAR THE 2ND TIME FLAG.
2474 ;*
2475 ;* SYNCHRONIZE WITH THE LTC.
2476 013260 012705 000001 2# : MOV #1,R5 ;SET OUTER LOOP COUNTER TO 1 LOOP.
2477 ;* ;INCREASE THE VALUE LOADED INTO THIS COUNTER IF THE < **
2478 ;* ;FOLLOWING LOOP FAILS ON FUTURE, FASTER PROCESSORS. < **
2479 013264 005000 CLR R0 ;CLEAR THE WAIT FOR CLOCK INT COUNTER.
2480 013266 012767 000001 167026 MOV #1,TIMER1 ;SET UP COUNT OF 1 TO SYNCH WITH LTC.
2481 013274 005767 167022 4# : TST TIMER1 ;CHECK FOR COUNTER HAVING GONE TO ZERO.
2482 013300 001410 BEQ 6# ;JUMP OUT OF LOOP IF LTC HAS INTERRUPTED.
2483 013302 005200 INC R0 ;COUNT THIS ITERATION OF THE INNER LOOP.
2484 013304 001373 BNE 4# ;LOOP IF COUNTER HAS NOT TURNED OVER.
2485 013306 005305 DEC R5 ;DECREMENT THE INNER LOOP COUNTER.
2486 013310 003371 BGT 4# ;LOOP IF OUTER LOOP COUNT NOT UP.
2487 ;*
2488 ;* IF WE GOT NO LTC INTERRUPT, INDICATE THAT THERE IS NO LTC AVAILABLE.
2489 ;* LTC MUST BE FLAKEY, OR NOT REALLY AN LTC AT ALL.
2490 ;*--
2491 013312 005067 167002 CLR CLKHRZ ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
2492 013316 000241 CLC ;INDICATE FAILURE FOR RETURN.
2493 013320 000461 BR 60# ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
2494 ;*
2495 ;* WE ARE NOW SYNCHRONIZED WITH THE LTC.
2496 ;* SET UP FOR THE CALIBRATION LOOP.
2497 ;*--
2498 013322 012704 002322 6# : MOV #TIMER1,R4 ;WILL TEST TIMER1 IN THE LOOP BELOW.

```

```

2499 013326 005001          CLR    R1          ;CLEAR THE OUTER LOOP COUNTER.
2500 013330 005002          CLR    R2          ;INDICATE TO CHECK ALL BITS OF TIMER1.
2501 013332 005003          CLR    R3          ;INDICATE TO CHECK FOR TIMER1 CLEAR.
2502 013334 012714 000001    MOV    #1,(R4)      ;LOAD TIMER1 WITH COUNT OF 1.
2503
2504 013340 016705 166770    8#:   MOV    MSLCNT,R5 ;LOAD MS LOOP COUNT.
2505 013344 011400 10#:   MOV    (R4),R0      ;GET THE TIMER1 VALUE.
2506 013346 010067 000120    MOV    R0,64#      ;SAVE WORD (LIKE IN THE REAL LOOP).
2507 013352 040200          BIC    R2,R0        ;LEAVE ALL THE BITS.
2508 013354 020003          CMP    R0,R3        ;COMPARE AGAINST ZERO.
2509 013356 000261          SEC                    ;SET CARRY IN CASE OF SUCCESS.
2510 013360 001406          BEQ                    ;EXIT LOOP IF TIMER1 HAS CLEARED.
2511 013362 005305          DEC    R5          ;COUNT DOWN THE INSIDE MS LOOP COUNT.
2512 013364 001367          BNE    10#         ;LOOP IF MS NOT UP.
2513 013366 005301          DEC    R1          ;DECREMENT THE MS TIME COUNT.
2514 013370 001363          BNE    8#          ;KEEP LOOPING.
2515 013372 004767 001054    JSR    PC,OOPS      ;WE OVERFLOWED, SOMETHING IS WRONG, ABORT.
2516
2517
2518
2519
2520
2521
2522 013376 005401 12#:   NEG    R1          ;GET NUMBER OF OUTER LOOPS.
2523 013400 016702 166730    MOV    MSLCNT,R2    ;GET THE NUMBER OF INNER LOOPS PER OUTER LOOP.
2524 013404 010203          MOV    R2,R3        ;COPY NUMBER OF LOOPS FOR MULTIPLY.
2525 013406 160502          SUB    R5,R2        ;CALC # OF INNER LOOPS DONE IN LAST OUTER LOOP
2526 013410 010204          MOV    R2,R4        ; AND ADD TO ACCUMULATOR LSWORD.
2527 013412 005005          CLR    R5          ;CLEAR ACCUMULATOR MSWORD.
2528 013414 005301 14#:   DEC    R1          ;CHECK R1 FOR 0 CONDITION
2529 013416 100403          BMI    16#         ; SKIP MULTIPLICATION IF ZERO
2530 013420 060304          ADD    R3,R4        ;MULTIPLY NUMBER OF INNER
2531 013422 005505          ADC    R5          ; LOOPS PER OUTER LOOP BY
2532 013424 000773          BR    14#         ;NUMBER OF OUTER LOOPS PERFORMED.
2533
2534
2535
2536 013426 016701 166700    16#:   MOV    MSTICK,R1   ;# OF MS PER LTC TICK IS DIVISOR.
2537 013432 010403          MOV    R4,R3        ;LSWORD OF LOOP COUNT IS LSWORD OF DIVIDEND.
2538 013434 010502          MOV    R5,R2        ;MSWORD OF LOOP COUNT IS MSWORD OF DIVIDEND.
2539 013436 004767 002662    JSR    PC,UNSDIV    ;DIVIDE NUMBER OF LOOPS BY MS PER LTC TICK.
2540 013442 103402          BCS    18#         ;BYPASS OOPS IF WE'RE OK.
2541 013444 004767 001002    JSR    PC,OOPS      ;CLOCK ROUTINES ARE NOT LONG ENOUGH, OR BUG.
2542 013450 010167 166660 18#:   MOV    R1,MSLCNT   ;SET NEW VALUE FOR MS LOOP COUNT.
2543 013454 005167 000010    COM    62#         ;SET THE 2ND ITERATION FLAGS IF 1ST ITERATION.
2544 013460 001277          BNE    2#          ;BRANCH IF ONLY ONE ITERATION DONE.
2545 013462 000261          SEC                    ;SET THE SUCCESS FLAG FOR EXIT.
2546
2547 013464 004736 60#:   PASS          ;RESTORE GPWS.
2548 013466 000207          RTS    PC          PC,9(SP); RETURN TO PREG05 SUBRT.
2549
2550 013470 000000 62#:   .WORD 0          ;C RRY - SUCCESS FLAG. SET IF SUCCESS.
2551 013472 000000 64#:   .WORD 0          ;2ND CALIBRATION ITERATION FLAGS.
;DUMMY WORD FOR STORAGE OF THE READ WORD.

```

```

2553 .SBTTL GLOBAL SUBROUTINE - CHKBMP -
2554 ;* *****
2555 ;* - CHECK IF CHARACTER IS A BMP CODE -
2556 ;* THIS SUBROUTINE IS USED TO CHECK FOR BMP CODES.
2557 ;* IF A BMP CODE IS DETECTED, IT WILL BE SAVED ON THE QUEUE TO BE REPORTED
2558 ;* LATER. THE CARRY IS USED AS A FLAG TO INDICATE A CODE HAS BEEN FOUND.
2559 ;*
2560 ;* INPUTS: R2 - CONTAINS THE DATA TO BE CHECKED.
2561 ;*
2562 ;* OUTPUTS: R1 - CONTAINS THE MESSAGE TO BE REPORTED.
2563 ;* ERRBLK - CONTAINS THE ERROR REPORTING ROUTINE.
2564 ;* CARRY BIT IS USED TO INDICATE A BMP CODE FOUND, CARRY SET.
2565 ;*
2566 ;* CALLING SEQUENCE: JSR PC,CHKBMP
2567 ;*
2568 ;* COMMENTS:
2569 ;*
2570 ;* SUBORDINATE ROUTINES CALLED: SAVBMP.
2571 ;*
2572 ;* *****
2573 CHKBMP:: SAVE JSR R5,PREG05 ;SAVE CONTENTS OF GPRS R0 THRU R5.
013474 004567 170370 ;CALL REGISTER SAVE SUBRT.
013474 170301 ;SET UP THE FLAGS OF A BMP CODE.
2574 013500 012700 170301 MOV #170301,R0 ;TRY TO CLEAR THE BMP CODE FLAGS.
2575 013504 040200 BIC R2,R0 ;IF NOT A BMP CODE, EXIT WITH FAILURE.
2576 013506 001011 BNE 2# ;SAVE THE BMP CODE ON THE QUEUE.
2577 013510 004767 001766 JSR PC,SAVBMP ;PASS THE MESSAGE TO BE REPORTED.
2578 013514 012701 006666 MOV #EM5303,R1 ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
2579 013520 012767 011642 170340 MOV #ER1603,ERRBLK ;PASS FLAG TO INDICATE SUCCESS, BMP CODE FOUND.
2580 013526 000261 SEC ;EXIT.
2581 013530 000401 BR 60# ;PASS FLAG TO INDICATE FAILURE.
2582 013532 00C241 2# CLC ;RESTORE GPRS, EXCEPT
2583 013534 000004 60# PASS R1 ;PUT R1 IN STACK SLOT.
013534 010166 000004 MOV R1,R1SLOT(SP) ;RETURN TO PREG05 SUBRT.
013540 004736 JSR PC,@(SP) ;R1 - CONTAINS THE ADDRESS OF ERROR MESSAGE.
2584 ;CARRY BIT - SET INDICATES SUCCESS.
2585
2586 013542 000207 RTS PC

```

```

2588 .SBTTL GLOBAL SUBROUTINE - CKTRAP -
2589 ;*****
2590 ;* CHECK TRAP ROUTINE -
2591 ;* THIS SUBROUTINE IS USED TO CHECK FOR A BUS TIME-OUT TRAP (004 TRAP)
2592 ;* WHICH IS CAUSED BY AN ACCESS TO A NON-EXISTENT MEMORY OR I/O LOCATION.
2593 ;* IF THE TRAP DOES NOT OCCUR, THIS ROUTINE RETURNS A SUCCESS INDICATION.
2594 ;*
2595 ;* INPUTS: R0 - SOURCE ADDRESS FOR MOVE.
2596 ;* R1 - DESTINATION ADDRESS FOR MOVE.
2597 ;* (R0) - SOURCE FOR THE MOVE.
2598 ;*
2599 ;* OUTPUTS: (R1) - WRITTEN TO THE CONTENTS OF (R0).
2600 ;* CARRY FLAG - SET ON RETURN IF NO 004 TRAP DETECTED.
2601 ;* TP4FLG - NONZERO IF TRAP OCCURRED, CLEARED OTHERWISE.
2602 ;*
2603 ;* CALLING SEQUENCE: JSR PC,CKTRAP
2604 ;*
2605 ;* COMMENTS: IF THIS SUBROUTINE CAUSES A TRAP, EITHER THE ADDRESS WHICH
2606 ;* IS LABELED ADRPTR WILL BE THE TRAP PC ADDRESS ON THE STACK.
2607 ;*
2608 ;* SUBORDINATE ROUTINES CALLED: NONE.
2609 ;*****
2610
2611 013544 CKTRAP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
2612 013544 004567 170320 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2613 013550 005067 166532 CLR TP4FLG ;CLEAR THE 004 TRAP FLAGS.
2614 013554 011011 MOV (R0),(R1) ;PERFORM THE MOVE IN QUESTION.
2615 013556 005767 166524 ADRPTR:: TST TP4FLG ;CHECK FOR OCCURENCE OF TRAP.
2616 013562 000261 SEC ;INDICATE SUCCESS.
2617 013564 001401 BEQ 60$ ;EXIT WITH SUCCESS IF TRAP DID NOT OCCUR.
2618 013570 000241 CLC ;INDICATE FAILURE.
2619 013572 004736 60$: PASS ;RESTORE GPRS.
2619 013572 000207 RTS PC JSR PC,(SP)+ ;RETURN TO PREG05 SUBRT.

```

```

2621 .SBTTL GLOBAL SUBROUTINE - CLNRST -
2622 ;*****
2623 ;* - CLEAN RESET OF THE DEVICE UNDER TEST -
2624 ;* THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
2625 ;* THE DUT'S SELF-TEST IS SKIPPED, AND THE FIFO IS PURGED OF ANY ERROR
2626 ;* CODES, ETC.
2627 ;* IF THE RESET DOES NOT SUCCESSFULLY COMPLETE, THEN THE CARRY BIT IS
2628 ;* PASSED BACK TO THE CALLING ROUTINE (CLEAR).
2629 ;*
2630 ;* INPUTS: CSRA - CONTAINS THE ADDRESS OF THE CSR
2631 ;* TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
2632 ;* ERRNBR - ERROR NUMBER FOR POSSIBLE ERROR REPORT.
2633 ;* ERRTAB - ERRTP,ERNBR,AND ERRMSG SET UP CORRECTLY.
2634 ;*
2635 ;* OUTPUTS: THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
2636 ;* CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
2637 ;* ERRBLK - VALUE MAY BE DESTROYED.
2638 ;* IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
2639 ;* TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.
2640 ;*
2641 ;* CALLING SEQUENCE: JSR PC,CLNRST
2642 ;*
2643 ;* COMMENTS: THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS ERRNBR.
2644 ;* THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.
2645 ;*
2646 ;* SUBORDINATE ROUTINES CALLED: DELAY,MSLGET,PUFIFO,RESETT.
2647 ;*****
2648
2649 013574 CLNRST:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
013574 004567 170270 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2650
2651 ;*
2652 ;* RESET THE DUT.
2653 ;* THIS ROUTINE REPORTS ERRORS WITH NUMBERS FROM ERRNBR THRU ERRNBR+2.
2654 013600 004767 001524 JSR PC,RESETT ;RESET THE DUT TO A KNOWN STATE.
2655 013604 103002 BCC 600 ;EXIT ROUTINE WITH ABORT TEST INDICATOR.
2656
2657 ;*
2658 ;* PURGE THE FIFO OF ERROR CODES. SAVE ANY BMP CODES FOUND.
2659 013606 004767 001150 JSR PC,PUFIFO ;PURGE THE FIFO.
2660
2661 013612
2662 013612 004736 600: ;EXIT THE TEST USING RESETT OR PUFIFO STATUS.
;RESTORE GPRS, PASS THE FOLLOWING INTACT:
;PC,B(SP). ;RETURN TO PREG05 SUBRT.
2663 ;CARRY BIT:IF CLEAR, THEN ABORT THE TEST.
2664 013614 000207 RTS PC

```

```

2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682 013616
      013616 004567 170246
2683 013622 012701 000020
2684 013626 005020
2685 013630 005301
2686 013632 001375
2687 013634
      013634 004736
2688 013636 000207

```

```

.SBTTL GLOBAL SUBROUTINE - CLR16W -
; * *****
; * - CLEAR SIXTEEN WORDS ROUTINE -
; * THIS SUBROUTINE CLEARS 16 WORDS STARTING WITH THE SPECIFIED WORD.
; *
; * INPUTS: RO - ADDRESS OF THE FIRST WORD TO CLEAR.
; *
; * OUTPUTS: (RO) TO (RO+15) - 16 WORDS OF MEMORY ARE CLEARED TO 0.
; *
; * CALLING SEQUENCE: JSR PC,CLR16W
; *
; * COMMENTS:
; *
; * SUBORDINATE ROUTINES CALLED: NONE.
; -- *****

CLR16W:: SAVE
      MOV #16.,R1 JSR R5,PREG05 ;SAVE CONTENTS OF GPRS RO THRU R5.
24: CLR (R0)+ ;CALL REGISTER SAVE SUBRT.
      DEC R1 ;SET THE LOOP COUNTER TO 16.
      BNE 24 ;CLEAR A WORD OF MEMORY.
60: PASS ;COUNT THIS LOOP.
      ;LOOP IF NOT 16 WORD CLEARED.
      ;RESTORE GPRS.
      JSR PC,8(SP)+ ;RETURN TO PREG05 SUBRT.
      RTS PC

```

2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714 013640
013640 004567 170224
2715 013644 005003
2716 013646 012704 002620
2717 013652 010377 166360
2718 013656 017700 166362
2719 013662 011405
2720 013664 040005
2721 013666 042400
2722 013670 050005
2723 013672 012700 043777
2724 013676 120301
2725 013700 001001
2726 013702 050200
2727 013704 040005
2728 013706 001006
2729 013710 005203
2730 013712 020327 000020
2731 013716 002755
2732 013720 000261
2733 013722 000401
2734
2735 013724 000241
2736
2737 013726
013726 004736
2738 013730 000207

```

.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.
;* FLSA - 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.
;--*****
CMPMST:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
CLR R3 ;CLEAR THE LINE COUNTER.
MOV #STSTB,R4 ;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
24: MOV R3,BCSRA ;SET UP THE CSR IND.ADR.REG FIELD.
MOV #FLSA,R0 ;GET THE CONTENTS OF THIS LINE'S STAT REGISTER.
MOV (R4),R5 ;GET THE PREVIOUS CONTENTS FROM STORAGE.
BIC R0,R5
BIC (R4),R0
BIS R0,R5 ;XOR PRESENT AND STORED STAT VALUES.
MOV #43777,R0 ;PREPARE TO MASK OUT UNUSED BITS.
CMPB R3,R1 ;TEST FOR THIS BEING SPECIFIED LINE.
BNE 104 ;DON'T MASK OUT SPECIFIED BITS IF IT IS NOT.
BIS R2,R0 ;MASK OUT SPECIFIED BITS.
BIC R0,R5 ;GET BIT MAP OF UNDESIRED CHANGES.
BNE 504 ;EXIT WITH FAILURE IF CHANGES OCCURRED.
INC R3 ;SELECT NEXT LINE.
CMP R3,#NUMLNS ;ALL LINES DONE?
BLT 24 ;LOOP IF NOT ALL LINES DONE.
SEC ;INDICATE SUCCESS.
BR 604 ;EXIT THIS ROUTINE WITH SUCCESS.

504: CLC ;INDICATE FAILURE.

604: PASS
RTS PC JSR ;RESTORE GPRS.
PC,B(SP) ;RETURN TO PREG05 SUBRT.
; CARRY - SUCCESS FLAG (SET IF SUCCESS).

```

```

2740 .SBTTL GLOBAL SUBROUTINE - DELAY -
2741 ;*****
2742 ;* - DELAY SUBROUTINE -
2743 ;* THIS SUBROUTINE IS USED TO DELAY A VARIABLE NUMBER OF MILLI-SECONDS.
2744 ;*
2745 ;* INPUTS: R4 - CONTAINS THE NUMBER OF MS TO DELAY.
2746 ;* MSLCNT.
2747 ;*
2748 ;* OUTPUTS: NONE.
2749 ;*
2750 ;* CALLING SEQUENCE: JSR PC,DELAY
2751 ;*
2752 ;* COMMENTS: IF NO HARDWARE CLOCK INTERRUPTS ARE OCCURING, CONTROL-CS WILL
2753 ;* NOT BE HONORED FOR THE DURATION OF THE DELAY.
2754 ;*
2755 ;* SUBORDINATE ROUTINES CALLED: NONE.
2756 ;*****
2757
2758 013732 DELAY:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
013732 004567 170132 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2759 013736 010401 MOV R4,R1 ;PASS NUMBER OF MS DELAY AS TIME-OUT VALUE.
2760 013740 012702 177777 MOV #-1,R2 ;TELL MSLOOP ROUTINE TO CHECK ALL BITS.
2761 013744 005003 CLR R3 ;TELL MSLOOP RTN TO CHECK FOR ALL BITS CLEAR.
2762 013746 012704 013770 MOV #62#,R4 ;TELL MSLOOP TO CHECK DUMMY NON-ZERO WORD.
2763 013752 004767 000460 JSR PC,MSLOOP ;DELAY THE REQUESTED # OF MS.
2764 013756 103002 BCC 60# ;EXIT ROUTINE IF WE TIMED-OUT.]
2765 013760 004767 000466 JSR PC,OOPS ;IF NO TIME-OUT, BAD PROGRAM OR HOST MACHINE.
2766 013764 PASS ;RESTORE GPRS.
013764 004736 JSR PC,B(SP)+ ;RETURN TO PREG05 SUBRT.
2767 013766 000207 RTS PC
2768
2769 013770 177777 62# : .WORD -1 ;DUMMY, NON-ZERO WORD.

```

```

2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801 013772
      013772 004567 170072
2802 013776 012704 000200
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813 014002
      014002 104440
      014004 010005
2814 014006
      014006 012700 000340
      014012 104441
2815 014014 056701 166244
2816 014020 010177 166212
2817 014024 105777 166222
2818 014030 000241
2819 014032 100411
2820 014034 010377 166214
2821 014040 010277 166204
2822 014044 110477 166202

```

```

.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.
;*          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 ASSUMES MEMORY MANAGEMENT IS DISABLED AND
;*           CLEARS THE TWO MSB OF THE DMA ADDRESS, I.E. BITS 0 AND 1
;*           OF THE TBUFFAD2 REG.
;*
;* SUBORDINATE ROUTINES CALLED: NONE.
;-- *****
DODMA:: SAVE R5,REG05 ;SAVE CONTENTS OF GPRS R0 THRU R5.
        MOV #200,R4 ;CALL REGISTER SAVE SUBRT.
        ;PREPARE TO CLEAR UPPER 6 BITS OF DMA BUFF ADR.

;*
;* WRITE THE DMA PARAMETERS OUT TO THE DUT DMA REGISTERS.
;* DISABLE INTERRUPTS.
;* SET UP DUT CSR IND.ADR.REG FIELD.
;* WRITE THE DMA TRANSMIT CHARACTER COUNT.
;* WRITE THE LEAST SIGNIFICANT 16 BITS OF THE DMA BUFFER START ADDRESS.
;* WRITE THE MOST SIGNIFICANT 6 BITS OF THE ADDRESS,
;* SETTING THE DMA_START BIT, AND INITIATING THE DMA TRANSMISSION.
;--
60: GETPRI R5 ;GET THE PRESENT PROCESSOR PRIORITY.
        TRAP C#GPRI
        MOV RO,R5
        SETPRI #PRI07 ;DISABLE ALL HARDWARE INTERRUPTS.
        MOV #PRI07,RO
        TRAP C#SPRI

        BIS IESTAT,R1 ;PREPARE FOR SETUP OF LINE NUMBER IN DUT CSR.
        MOV R1,#CSRA ;SET UP THE DUT CSR IND.ADR.REG FIELD.
        TSTB #TXAD2A ;TEST THE DUT DMA_START BIT.
        CLC ;INDICATE FAILURE IN CASE DMA.H0 BIT IS SET.
        BMI 60 ;EXIT WITH FAILURE IF DMA.H0 BIT IS SET.
        MOV R3,#TXBFCA ;WRITE THE DMA CHARACTER COUNT.
        MOV R2,#TXAD1A ;WRITE THE LS 16 BITS OF BUFFER ADDRESS.
        MOV R4,#TXAD2A ;WRITE MS 6 BITS OF ADR AND START DMA TX.

```

```

2823 014050          SETPRI R5          ;RESTORE THE PROCESSOR PRIORITY.
      014050 010500
      014052 104441
2824 014054 000261          SEC          ;INDICATE SUCCESS.
2825
2826 014056          60: PASS          ;RESTORE GPRS.
      014056 004736          JSR          PC,8(SP)+ ;RETURN TO PREG05 SUBRT.
2827 014060 000207          RTS PC          ; CARRY - SUCCESS FLAG (SET IF SUCCESS).

```

```

2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848 014062
      014062 004567 170002
2849
2850
2851
2852 014066 005001
2853 014070 012703 000020
2854 014074 016700 166130
2855 014100 012705 000001
2856 014104 030500
2857 014106 001006
2858 014110 006305
2859 014112 005201
2860 014114 020103
2861 014116 002772
2862 014120 000241
2863 014122 000401
2864 014124 000261
2865
2866 014126
      014126 010166 000004
      014132 010566 000014
      014136 004736
2867
2868
2869
2870 014140 000207

```

```

.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.
;-- *****
FINACT:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
              R5,PREG05 ;CALL REGISTER SAVE SUBRT.
;+
; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
;--
      CLR R1 ;CLEAR THE LINE NUMBER COUNTER.
      MOV #NUMLNS,R3 ;GET MAX LINE NUMBER.
      MOV ACTLNS,R0 ;GET THE ACTIVE LINE BIT MAP.
      MOV #1,R5 ;SET UP A LINE BIT MASK.
20: BIT R5,R0 ;LOOK FOR AN ACTIVE LINE.
     BNE 40 ;BRANCH TO BEGIN TEST IF A LINE HAS BEEN FOUND.
     ASL R5 ;SHIFT THE BIT MASK FOR THE NEXT LINE.
     INC R1 ;INCREMENT THE LINE NUMBER COUNTER.
     CMP R1,R3 ;CHECK IF ALL LINES HAVE BEEN TRIED.
     BLT 20 ;LOOP TO TRY THE NEXT LINE.
     CLC ;CLEAR CARRY BIT, NO ACTIVE LINE FOUND.
40: BR 60 ;EXIT WITH FAILURE.
     SEC ;SET CARRY, SUCCESS.
60: PASS R1,R5 ;RESTORE GPRS, EXCEPT
      MOV R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
      MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
      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.
RTS PC

```

2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901

014142
014142 004567 167722
014146 012702 002660
014152 005003
014154 110322
014156 005203
014160 020227 003260
014164 103773
014166
014166 004736
014170 000207

```

.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,INDATP
;+
;+ COMMENTS:
;+
;+ SUBORDINATE ROUTINES CALLED: NONE.
;-- *****
INDATP:: SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
;CALL REGISTER SAVE SUBRT.
                JSR      R5,PREG05
                MOV     #BUFBAS,R2
;INITIALIZE THE DATA PATTERN IN THE GENERAL
; DATA BUFFER TO A 256 BYTE PATTERN.
24:             CLR     R3
                MOVB   R3,(R2)+
;
                INC    R3
;SELECT THE NEXT CHARACTER.
                CMP    R2,#BUFMID
;CHECK IF WE HAVE 256 DATA PATTERNS.
                BLO   24
;
604:           PASS
;RESTORE GPRS.
                JSR      PC,@(SP)+
;RETURN TO PREG05 SUBRT.
                RTS    PC

```

```

2903 .SBTTL GLOBAL SUBROUTINE - INDTPX -
2904 ;* *****
2905 ;* - INITIALISE DATA PATTERN WITHOUT XON OR XOFF -
2906 ;* THIS SUBROUTINE IS USED TO INITIALISE AN INCREMENTAL BYTE DATA PATTERN
2907 ;* IN THE GENERAL BUFFER AREA.
2908 ;* THE DATA PATTERN WILL BE FROM 0 TO 255, BUT WILL EXCLUDE THE FOLLOWING
2909 ;* TWO CHARACTERS; (ASCII DC1, DC3) XON AND XOFF. THIS WILL CAUSE THE
2910 ;* LAST TWO DATA CHARACTERS TO BE THE SAME AS THE FIRST TWO.
2911 ;*
2912 ;* INPUTS: BUFBAS - ADDRESS OF THE START OF THE GENERAL BUFFER AREA.
2913 ;*          BUFMID - ADDRESS OF THE 255 TH LOCATION.
2914 ;*
2915 ;* OUTPUTS: THE FIRST 255 LOCATIONS OF THE GENERAL BUFFER AREA CONTAIN DATA
2916 ;*
2917 ;* CALLING SEQUENCE: JSR PC,INDTPX
2918 ;*
2919 ;* COMMENTS:
2920 ;*
2921 ;* SUBORDINATE ROUTINES CALLED: NONE.
2922 ;*
2923 ;*
2924 014172 014172 004567 167672 INDTPX:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
; R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2925 ;*
2926 ;* INITIALIZE THE 256 BYTE DATA PATTERN.
2927 ;* ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
2928 ;* NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
2929 ;*
2930 014176 012702 002660 MOV #BUFBAS,R2 ;INITIALIZE THE DATA PATTERN IN THE GENERAL
2931 014202 005003 CLR R3 ; DATA BUFFER TO A 256 BYTE PATTERN.
2932 014204 110322 2#: MOVB R3,(R2)+ ;
2933 014206 105203 INCB R3 ;SELECT THE NEXT CHARACTER.
2934 014210 122703 000021 CMPB #21,R3 ;CHECK FOR AN XON CHARACTER.
2935 014214 001001 BNE 4# ;BRANCH IF CHAR NOT AN XON.
2936 014216 105203 INCB R3 ;FORCE THE NEXT CHARACTER.
2937 014220 122703 000023 4#: CMPB #23,R3 ;CHECK FOR AN XOFF CHARACTER.
2938 014224 001001 BNE 6# ;BRANCH IF NOT AN XOFF CHARACTER.
2939 014226 105203 INCB R3 ;FORCE THE NEXT CHARACTER.
2940 014230 020227 003260 6#: CMP R2,#BUFMID ;CHECK IF WE HAVE 256 DATA PATTERNS.
2941 014234 103763 BLO 2# ;
2942 ;
2943 014236 004736 60#: PASS JSR ;RESTORE GPRS.
014236 004736 PC,8(SP)+ ;RETURN TO PREG05 SUBRT.
2944 014240 000207 RTS PC

```

2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975

014242
014242 004567 167622
014246 042701 177760
014252 006301
014254 016100 002344
014260
014260 010066 000002
014264 004736
014266 000207

```

.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.
;*****
LINBIT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;*****
;***** JSR RS,PREGOS ;CALL REGISTER SAVE SUBRT.
;***** BIC #177760,R1 ;MASK OUT ALL BUT 4 LSBITS OF THE LINE #.
;***** ASL R1 ;MULTIPLY LINE # BY 2 TO GET WORD TABLE OFFSET.
;***** MOV BITTBL(R1),R0 ;GET THE SINGLE BIT BIT MAP.
600: ;RESTORE GPRS, EXCEPT THE FOLLOWING.
;***** PASS R0 ;RO,ROSL0T(SP) ;PUT RO IN STACK SLOT.
;***** MOV ;PC,@(SP) ;RETURN TO PREGOS SUBRT.
;***** JSR ;RO - BIT MAP WITH LINE # BIT SET.
;*****
RTS PC

```

```

2977 .SBTTL GLOBAL SUBROUTINE - MAPCNT -
2978 ;* *****
2979 ;* - COUNT BITS IN BIT MAP ROUTINE -
2980 ;* THIS SUBROUTINE COUNTS THE NUMBER OF BITS WHICH ARE SET IN A BIT MAP.
2981 ;*
2982 ;* INPUTS: R2 - THE BIT MAP FOR WHICH TO COUNT THE BITS.
2983 ;*
2984 ;* OUTPUTS: R2 - COUNT OF THE NUMBER OF BITS THAT WERE SET.
2985 ;*
2986 ;* CALLING SEQUENCE: JSR PC,MAPCNT
2987 ;*
2988 ;* COMMENTS:
2989 ;*
2990 ;* SUBORDINATE ROUTINES CALLED: NONE.
2991 ;*
2992 ;* *****
2993 014270 MAPCNT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
014270 004567 167574 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2994 014274 010201 MOV R2,R1
2995 014276 001405 BEQ 601 ;EXIT WITH ZERO IF NO BITS ARE SET IN MAP.
2996
2997 014300 005002 CLR R2 ;CLEAR THE BIT COUNT.
2998 014302 000261 SEC ;COUNT THE LAST BIT TO BE SHIFTED OUT.
2999
3000 014304 005502 21: ADC R2 ;COUNT THE BIT IF IT WAS SET.
3001 014306 006301 ASL R1 ;SHIFT ANOTHER BIT OUT OF THE MAP.
3002 014310 001375 BNE 21 ;LOOP IF ALL BITS NOT SHIFTED OUT OF MAP.
3003
3004 014312 010266 000006 601: PASS R2 ;RESTORE GPRS, EXCEPT THE FOLLOWING:
014312 004736 MOV R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
014316 000207 JSR PC,B(SP) ;RETURN TO PREG05 SUBRT.
3005 014320 000207 RTS PC ; R2 - COUNT OF BITS SET IN BIT MAP.

```

```

3007 .SBTTL GLOBAL SUBROUTINE - MSLGET -
3008 *****
3009 ;* - MILLI SECONDS LOOP WHICH RETURNS READ WORD AND REMAINING TIME -
3010 ;* THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE. IT IS USED
3011 ;* TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD. THE
3012 ;* CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
3013 ;* DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
3014 ;* THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
3015 ;* ROUTINE AND THEN ONCE EACH MILLI-SECOND THERE AFTER.
3016 ;* UPON RETURN, THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION
3017 ;* IS RETURNED BY THIS SUBROUTINE.
3018 ;*
3019 ;* INPUTS: R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
3020 ;* R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
3021 ;* R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
3022 ;* R4 - ADDRESS OF THE WORD TO TEST.
3023 ;* MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.
3024 ;*
3025 ;* OUTPUTS: R0 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
3026 ;* R1 - REMAINING NUMBER OF MS IN TIME-OUT TIME.
3027 ;* CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).
3028 ;*
3029 ;* CALLING SEQUENCE: JSR PC,MSLGET
3030 ;*
3031 ;* COMMENTS: THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
3032 ;* CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
3033 ;* ON THE SYSTEM.
3034 ;* THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
3035 ;* DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
3036 ;* LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
3037 ;* IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
3038 ;* THE DESIRED CONDITION BEFORE RETURNING. IT INDICATES SUCCESS
3039 ;* IF THE CONDITION IS NET, FAILURE OTHERWISE.
3040 ;*
3041 ;*
3042 ;* SUBORDINATE ROUTINES CALLED: NONE.
3043 ;*
3044 ;* *****
3045 014322 MSLGET:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
014322 004567 167542 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3046 ;*
3047 ;* SET UP MASK FOR REMOVING UNUSED BITS IN THE TEST WORD, AND CLEAR UNUSED
3048 ;* BITS IN THE DESIRED STATE WORD TO ALLOW DIRECT COMPARISON.
3049 ;*
3050 014326 005102 COM R2 ;GET MASK OF UNUSED BITS.
3051 014330 040203 BIC R2,R3 ;MASK OUT UNUSED BITS IN DESIRED STATE WORD.
3052 ;*
3053 ;* HANDLE THE TEST AND EXIT IF WE HAVE A 0 TIME-OUT VALUE.
3054 ;*
3055 014332 005701 TST R1 ;TEST THE TIME-OUT VALUE FOR ZERO.
3056 014334 001011 BNE 2# ;IF NON-ZERO TIME-OUT, GO LOOP AND TEST.
3057 014336 011400 MOV (R4),R0 ;GET THE WORD TO TEST BEFORE EXITING.
3058 014340 010067 000070 MOV R0,62# ;SAVE VALUE SO WE CAN RETURN IT.
3059 014344 040200 BIC R2,R0 ;MASK OUT UNTESTED BITS OF WORD.
3060 014346 020003 CMP R0,R3 ;COMPARE AGAINST DESIRED STATE WORD.
3061 014350 000261 SEC ;INDICATE SUCCESS IN CASE WORDS ARE EQUAL.
3062 014352 001420 BEQ 6# ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.

```

```

3063 014354 000241          CLC          ;INDICATE FAILURE (TIME-OUT).
3064 014356 000416          BR          6#          ;EXIT WITH FAILURE, WORDS AREN'T EQUAL.
3065
3066          ;+
3067          ; NON-ZERO TIME-OUT VALUE. LOOP, WAITING FOR CONDITION OR TIME-OUT.
3068 014360 016705 165750 2#:      MOV      MSLCNT,R5      ;LOAD MS LOOP COUNT.
3069 014364 011400 4#:      MOV      (R4),R0      ;GET THE WORD TO TEST.
3070 014366 010067 000042      MOV      R0,62#      ;SAVE WORD IN CASE THIS IS THE LAST.
3071 014372 040200          BIC      R2,R0      ;MASK OUT UNTESTED BITS OF WORD.
3072 014374 020003          CMP      R0,R3      ;COMPARE AGAINST DESIRED STATE WORD.
3073 014376 000261          SEC          ;SET CARRY IN CASE OF SUCCESS.
3074 014400 001405          BEQ      6#          ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.
3075 014402 005305          DEC      R5      ;COUNT DOWN THE INSIDE MS LOOP COUNT.
3076 014404 001367          BNE      4#          ;LOOP IF MS NOT UP.
3077 014406 005301          DEC      R1      ;DECREMENT THE MS TIME COUNT.
3078 014410 001363          BNE      2#          ;IF TIME NOT UP, LOOP TO COUNT ANOTHER MS.
3079 014412 000241          CLC          ;CLEAR CARRY, WE TIMED-OUT.
3080
3081          ;+
3082          ; HAVE EITHER FOUND CONDITION, OR TIMED-OUT (POSSIBLY FROM 0 TIME-OUT VALUE).
3083          ; RESTORE THE LAST CONTENTS READ FROM THE TEST WORD. EXIT ROUTINE.
3084 014414 016700 000014 6#:      MOV      62#,R0      ;PASS OUT THE LAST READ WORD.
3085 014420 010066 000002 60#:     PASS      R0,R1      ;RESTORE GPRS, EXCEPT THE FOLLOWING:
          ;RO,ROSLOT(SP)      ;PUT RO IN STACK SLOT.
          ;R1,R1SLOT(SP)      ;PUT R1 IN STACK SLOT.
          ;PC,B(SP)+          ;RETURN TO PREG05 SUBRT.
          ;RO - LAST READ WORD CHECKED FOR CONDITION.
          ;R1 - REMAINING TIME (0 IF TIME-OUT OCCURED).
          ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.
          MOV      MOV
          JSR
3086
3087
3088 014432 000207          RTS      PC
3089
3090          ;+
3091          ; LOCAL STORAGE.
3092 014434 000000 62#:      .WORD      0          ;STORAGE FOR THE LAST READ WORD.

```

```

3094 .SBTTL GLOBAL SUBROUTINE - MSLOOP -
3095 ;*****
3096 ;* - TEST LOOP SUBROUTINE -
3097 ;* THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE. IT IS USED
3098 ;* TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD. THE
3099 ;* CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
3100 ;* DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
3101 ;* THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
3102 ;* ROUTINE AND THEN ONCE EACH MILLI-SECOND THEREAFTER.
3103 ;*
3104 ;* INPUTS: R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
3105 ;* R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
3106 ;* R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
3107 ;* R4 - ADDRESS OF THE WORD TO TEST.
3108 ;* MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.
3109 ;*
3110 ;* OUTPUTS: CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).
3111 ;*
3112 ;* CALLING SEQUENCE: JSR PC,MSLOOP
3113 ;*
3114 ;* COMMENTS: THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
3115 ;* CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
3116 ;* ON THE SYSTEM.
3117 ;* THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
3118 ;* DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
3119 ;* LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
3120 ;* IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
3121 ;* THE DESIRED CONDITION BEFORE RETURNING. IT INDICATES SUCCESS
3122 ;* IF THE CONDITION IS MET, FAILURE OTHERWISE.
3123 ;*
3124 ;* SUBORDINATE ROUTINES CALLED: MSLGET.
3125 ;*****
3126
3127 014436 MSLOOP:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
014436 004567 167426 R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3128
3129 ;*
3130 ;* CALLING THE MSLGET ROUTINE FROM THE MSLOOP ROUTINE ISOLATES THE CALLER OF
3131 ;* MSLOOP FROM THE RETURNED TEST WORD AND REMAINING TIME-OUT VALUES.
3132 ;*
3133 014442 004767 177654 JSR PC,MSLGET ;CALL THE MULTI-PURPOSE MS LOOP AND SEARCH RTN.
3134
3135 014446 004736 606: PASS ;RESTORE GPRS.
014446 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
3136 014450 000207 RTS PC ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.

```

```

3138 .SBTTL GLOBAL SUBROUTINE - OOPS -
3139 ;** *****
3140 ;* - PROGRAM ABORT SUBROUTINE -
3141 ;* THIS SUBROUTINE IS USED TO ABORT THE PROGRAM WHEN A FATAL ERROR IS
3142 ;* DETECTED IN THE PROGRAM OR THE HOST SYSTEM HARDWARE. AN ERROR MESSAGE
3143 ;* IS PRINTED GIVING SOME INFORMATION ABOUT THE NATURE OF THE ABORT.
3144 ;*
3145 ;* INPUTS: R1 - ERROR CODE GIVING REASON FOR ABORT.
3146 ;*
3147 ;* OUTPUTS: AN ERROR MESSAGE IS PRINTED.
3148 ;* A LIST OF RETURN PC VALUES FOR ALL SUBROUTINE CALLS IS PRINTED.
3149 ;*
3150 ;* CALLING SEQUENCE: JSR PC,OOPS
3151 ;*
3152 ;* COMMENTS:
3153 ;*
3154 ;* SUBORDINATE ROUTINES CALLED: NONE.
3155 ;-- *****
3156
3157 014452 OOPS:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
014452 004567 167412 ; JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3158 ; REPORT "HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED." ERROR.
3159 014456 ERRSF 101,EM0101
014456 104454 TRAP C#ERSF
014460 000145 .WORD 101
014462 014516 .WORD EM0101
014464 000000 .WORD 0
3160 ; REPORT "PROGRAM HUNG, WAITING FOR A CONTROL-C."
3161 014466 PRINTF #EM0102
014466 012746 014602 MOV #EM0102,-(SP)
014472 012746 000001 MOV #1,-(SP)
014476 010600 MOV SP,R0
014500 104417 TRAP C#PNTF
014502 062706 000004 ADD #4,SP
3162 014506 20: BREAK ;LOOK FOR OPERATOR CONTROL-C INPUT.
014506 104422 TRAP C#BRK
3163 014510 000776
3164 014512 600: BR 20 ;INFINITE LOOP.
014512 004736 ;DON'T NEED THIS, BUT SOMEBODY MAY CHANGE THIS
3165 014514 000207 ; PC,8(SP)+ ;RETURN TO PREG05 SUBRT.
3166 ; RTS PC ; ROUTINE IN THE FUTURE, SO BE CONSISTANT.
3167 014516 110 117 123 EM0101:: .ASCIZ /HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED./
014521 124 040 103
014524 117 115 120
014527 125 124 105
014532 122 040 110
014535 101 122 104
014540 127 101 122
014543 105 040 117
014546 122 040 123
014551 117 106 124
014554 127 101 122
014557 105 040 102
014562 125 107 040
014565 105 116 103
014570 117 125 116
014573 124 105 122

```

	014576	105	104	056
	014601	000		
3168	014602	045	116	045
	014605	101	120	122
	014610	117	107	122
	014613	101	115	040
	014616	110	125	116
	014621	107	054	040
	014624	127	101	111
	014627	124	111	116
	014632	107	040	106
	014635	117	122	040
	014640	101	040	103
	014643	117	116	124
	014646	122	117	114
	014651	055	103	056
	014654	040	074	052
	014657	052	052	052
	014662	052	052	052
	014665	052	052	052
	014670	052	052	052
	014673	045	116	045
3169	014676	116	000	

EMC102:: .ASCIZ /N#APROGRAM HUNG, WAITING FOR A CONTROL-C. <*****N#N/

.EVEN

3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194 014700
014700 004567 167164
3195 014704 016701 165326
3196 014710 016702 165326
3197 014714 042703 177760
3198 014720 056703 165340
3199 014724 010311
3200 014726 011204
3201
3202 014730
014730 010446
014732 012746 010760
014736 012746 005246
014742 012746 000003
014746 010600
014750 104415
014752 062706 000010
3203 014756
014756 004736
3204 014760 000207

```

.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.
;--*****
PRTLPR::SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV CSRA,R1 ;GET THE CSR ADDRESS.
MOV LPRA,R2 ;GET THE LPR ADDRESS.
BIC @177760,R3 ;CLEAR ANY UNWANTED BITS.
BIS IESTAT,R3 ;SET STATE OF TX AND RX INTERRUPT ENABLE BITS.
MOV R3,(R1) ;SELECT LINE.
MOV (R2),R4 ;GET CONTENTS OF THE LPR.
;PRINT MESSAGE "CONTENTS OF THE LPR:NNNNN"
PRINTX @EF9019,@EM9026,R4;PRINT OUT MESSAGE ON OPERATORS CONSOLE.
MOV R4,-(SP)
MOV @EM9026,-(SP)
MOV @EF9019,-(SP)
MOV @3,-(SP)
MOV SP,R0
TRAP C:PNTX
ADD @10,SP
601: PASS ;RESTORE GPRS.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC

```

```

3206 .SBTTL GLOBAL SUBROUTINE - PUFIFO -
3207 :*****
3208 : - PURGE THE FIFO
3209 : THIS ROUTINE TRIES TO REMOVE ALL THE CHARACTERS FROM THE FIFO.
3210 : ANY BMP CODES THAT ARE FOUND ARE SAVED ON THE BMP CODE QUEUE.
3211 :
3212 : INPUTS: RBUFA- CONTAINS THE ADDRESS OF THE RECEIVER.
3213 :
3214 :
3215 : OUTPUTS: CARRY BIT - INDICATES THE STATE OF THE FIFO, SET:= PURGED.
3216 : BMPCQ - THE CONTENTS OF THE BMP CODE QUEUE MAY BE UPDATED.
3217 :
3218 : CALLING SEQUENCE: JSR PC,PUFIFO
3219 :
3220 : COMMENTS:
3221 :
3222 : SUBORDINATE ROUTINES CALLED: SAVBMP.
3223 :*****
3224
3225 PUFIFO::SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3226 014762 004567 167102 MOV #512.,R1 ;SET MAXIMUM TRY COUNT OF 512.
3227 014772 016704 165242 MOV RBUFA,R4 ;GET ADDRESS OF THE RECEIVER BUFFER REGISTER.
3228
3229 014776 011402 2#: MOV (R4),R2 ;GET THE CONTENTS OF THE RECEIVER BUFFER REG.
3230 015000 100016 BPL 6# ;EXIT IF THE FIFO IS EMPTY, DATA_VALID CLR.
3231
3232 : CHECK IF THE READ CHARACTER IS ACTUALLY A BMP CODE.
3233 : IF IT IS, THEN SAVE IT ON THE BMP CODE QUEUE TO BE REPORTED LATER.
3234
3235 015002 012700 070000 MOV #70000,R0 ;GENERATE A BIT MAP OF CHAR ERROR BITS
3236 015006 040200 BIC R2,R0 ; WHICH ARE NOT SET FOR CHAR.
3237 015010 001006 BNE 4# ;THROW CHAR AWAY IF NOT BMP OR SELFTEST CODE.
3238
3239 : CHECK IF THE READ DATA IS MODEM STATUS , BMP OR SELFTEST?.
3240
3241 015012 012700 000301 MOV #301,R0 ; CHECK IF BMP.
3242 015016 040200 BIC R2,R0 ;TRY TO CLEAR BMP FLAGS IN THE READ DATA.
3243 015020 001002 BNE 4# ;IF IT IS MODEM OR SELFTEST CODE THROW IT AWAY.
3244 015022 004767 000454 JSR PC,SAVBMP ;SAVE BMP CODE ON THE QUEUE.
3245
3246 015026 005301 4#: DEC R1 ;DECREMENT THE TRY COUNT.
3247 015030 001362 BNE 2# ;LOOP TO TRY AGAIN.
3248 015032 000241 CLC ;CLEAR CARRY, TO INDICATE FIFO NOT PURGED.
3249 015034 000401 BR 60# ;EXIT WITH CARRY CLEAR.
3250 015036 000261 6#: SEC ;SET CARRY, TO INDICATE FIFO PURGED.
3251
3252 015040 004736 60#: PASS ;RESTORE GPRS.
;PC,@(SP), ;RETURN TO PREG05 SUBRT.
3253 JSR
3254 015042 000207 RTS PC ;CARRY BIT, SET INDICATES FIFO PURGED.

```

3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284 015044
015044 004567 167020
3285 015050 016746 167006
3286 015054 012705 001000
3287
3288
3289
3290
3291 015060 017702 165154
3292 015064 100063
3293
3294
3295
3296 015066 012700 070000
3297 015072 040200
3298 015074 001012
3299
3300
3301
3302
3303 015076 012767 012356 166762
3304 015104 012700 000300
3305 015110 040200
3306 015112 001003
3307 015114 004767 000362
3308 015120 000430
3309
3310
3311

```
.SBTTL GLOBAL SUBROUTINE                - PUFIFR -
;*****
;*   - PURGE FIFO REPORT ANY ERRORS FOUND.
;*   THIS ROUTINE REMOVES ALL DATA FROM THE FIFO. ANY BMP CODES THAT ARE
;*   FOUND ARE SAVE ON THE QUEUE TO BE REPORTED LATER IN THE BMP REPORT TEST.
;*   ANY UNEXPECTED DATA (IE ANY NON-STATUS INFORAMTION) THAT ARE FOUND,
;*   ARE REPORTED AS AN ERROR.
;*   IF THE FIFO WILL NOT PURGE AFTER 512 ATTEMPTS, THEN THE CURRENT TEST
;*   THAT CALLED THIS ROUTINE RECEIVES A FAILURE FLAG THAT SHOULD BE USED
;*   TO ABORT THE TEST.
;*
;* INPUTS:      ERRIBL - ERRTYPE, ERRMSG, ERRNBR ARE SET UP CORRECTLY.
;*              RBUFA- CONTAINS THE ADDRESS OF THE RECEIVER.
;*
;* OUTPUTS:     CARRY BIT - ABORT TEST FLAG, CLR = ABORT TEST, SET = OK.
;*              ERRBLK - VALUE WILL BE DASTROYED.
;*              BMPCQP - THE BMP CODE QUEUE POINTER MAY BE UPDATED.
;*              THE CONTENTS OF THE BMP CODE QUEUE MAY BE UDATED.
;*
;* CALLING SEQUENCE:  JSR      PC,PUFIFR
;*
;* COMMENTS:      THIS ROUTINE REPORTS ERRORS WITH NUMBERS INITIAL ERRNBR
;*                THRU TO ERRNBR+2.
;*                THE ERRNBR IS RESTORED TO ITS INITIAL VALUE BEFORE RETURNING.
;*
;* SUBORDINATE ROUTINES CALLED: ER1603,ER9001,ER9002,SAVBMP.
;*****
PUFIFR::SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGIS.ZR SAVE SUBRT.
MOV   ERRNBR,-(SP) ;SAVE THE CONTENTS OF THE ERROR NUMBER.
MOV   #512.,R5    ;SET MAXIMUM READ COUNTER TO 2*FIFO SIZE.
;
; READ DATA FROM THE FIFO UNTIL DATA VALID IS CLEAR OF READ COUNTER IS ZERO.
; REPORT ANY BMP OR UNEXPECTED DATA AS ERRORS.
;
;
20:  MOV   @RBUFA,R2 ;GET THE CONTENTS OF THE RECEIVER BUFFER REG.
     BPL   8#       ;EXIT IF DATA VALID CLEAR, IE. FIFO PURGED.
;
; CHECK IF READ DATA IS STATUS OR UNEXPECTED CHARACTER.
;
;
     MOV   #70000,R0 ;GENERATE A BIT MAP OF CHAR ERROR BITS
     BIC   R2,R0    ; WHICH ARE NOT SET FOR CHAR.
     BNE   4#       ;SKIP BMP CHECK IF IT IS UNEXPECTED DATA.
;
; CHECK IF THE READ DATA IS MODEM STATUS , BMP OR SELFTEST?.
; IF IT IS A BMP CODE THEN SAVE IT ON THE QUEUE.
;
;
     MOV   @ER9001,ERRBLK ;SET UP THE CORRECT ERROR REPORTING ROUTINE.
     MOV   #300,R0       ; CHECK IF BMP OR SELFTEST?.
     BIC   R2,R0        ;TRY TO CLEAR BMP FLAGS IN THE READ DATA.
     BNE   4#           ;SKIP BMP ERROR REPORT IF MODEM OR SELFTEST?.
     JSR   PC,SAVBMP    ;SAVE THE BMP CODE ON THE QUEUE.
     BR   6#           ;BRANCH TO CHECK READ COUNT.
;
; CHECK IF THE READ DATA IS MODEM, SELFTEST OR UNEXPECTED DATA.
;
;

```

```

3312 015122 032702 000001      4:      BIT      #BIT0,R2      ;TEST THE MODEM STATUS INDICATION BIT.
3313 015126 001425              BEQ      6:      ;DO NOT REPORT ANY ERROR IF MODEM STATUS.
3314 015130 012701 011004      MOV      #EM9104,R1 ;PASS THE CORRECT ERROR MESSAGE TO REPORT.
3315 015134 010203              MOV      R2,R3      ;EXTRACT THE LINE NUMBER FROM
3316 015136 000303              SWAB     R3          ; THE READ DATA.
3317 015140 042703 177760      BIC      #177760,R3 ;
3318 015144 006303              ASL      R3          ;FORM LINE NUMBER TIMES 2 FOR ER9002 ROUTINE.
3319 015146 052704 100000      BIS      #BIT15,R4 ;SET THE "NONE" EXPECTED MESSAGE FLAG.
3320 015152 005267 166704      INC      ERRNBR     ;SET ERROR NUMBER TO INTIAL ERRNR+1.
3321 015156 012767 012456 166702 MOV      #ER9002,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
3322                                ;REPORT ERROR "UNEXPECTED DATA FOUND IN FIFO".
3323 015164                                ;ERROR                                >>>>> ERROR <<<<<.
                                ;                                TRAP      C#ERROR
                                ;                                >>>>> ERROR <<<<<.
3324                                ;+
3325                                ; EXIT WITH FAILURE IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
3326                                ;-
3327 015166 032767 000100 165024      BIT      #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
3328 015174 001415              BEQ      7:      ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
3329                                ;DURING THE SOFTWARE QUESTIONS.
3330
3331 015176 005367 166660      DEC      ERRNBR     ;RESTORE ERROR NUMBER TO INTIAL ERRNBR.
3332
3333 015202 005305      6:      DEC      R5          ;DECREMENT READ COUNTER.
3334 015204 001325      BNE     2:      ;LOOP TO READ NEXT CHAR FROM FIFO IF COUNT > 0.
3335
3336                                ;+
3337                                ; THE FIFO WILL NOT CLEAR, REPORT THE ERROR AND INDICATE THAT THE TEST IS TO
3338                                ; BE ABORTED.
3338                                ;-
3339 015206 062767 000002 166646      ADD      #2,ERRNBR  ;SET ERROR NUMBER TO INTIAL ERRNBR+2.
3340 015214 012767 011642 166644      MOV      #ER1603,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
3341 015222 012701 010647      MOV      #EM9017,R1 ;PASS THE MESSAGE TO BE REPORTED.
3342                                ;REPORT THE ERROR "FIFO WILL NOT PURGE, (DATA VALID STUCK SET)"
3343                                ;"?????? TEST ABORTED".
3344 015226                                ;ERROR                                >>>>> ERROR <<<<<.
                                ;                                TRAP      C#ERROR
                                ;                                >>>>> ERROR <<<<<.
3345 015230 000241      7:      CLC          ;INDICATE THE TEST IS TO BE ABORTED.
3346 015232 000401      BR      10:     ;EXIT THIS ROUTINE AND ABORT THE CURRENT TEST.
3347
3348 015234 000261      8:      SEC          ;SET THE CARRY, DO NOT ABORT THE TEST.
3349
3350 015236 012667 166620      10:     MOV      (SP)+,ERRNBR ;RESTORE INITIAL ERROR NUMBER.
3351 015242 004736      60:     PASS     ;RESTORE GPRS.
                                ;RETURN TO PREGOS SUBRT.
3352                                ;CARRY BIT, SET INDICATES FIFO PURGED, DO NOT
3353                                ; ABORT THE TEST.
3354 015244 000207      RTS      PC

```

```

3356 .SBTTL GLOBAL SUBROUTINE - READBX -
3357 ;** *****
3358 ;* - READ CHARACTERS FROM THE FIFO AND CHECKS FOR BMPS AND XONS-
3359 ;* THIS SUBROUTINE IS USED IN THE FIHAVL.TST.
3360 ;* IT READS THE SPECIFIED NUMBER OF CHARACTERS FROM THE FIFO AND CHECKS
3361 ;* FOR BMP CODES AND XON CHARACTERS.
3362 ;*
3363 ;* INPUTS: R0 - CONTAINS THE NUMBER OF CHARS TO READ FROM THE FIFO.
3364 ;*
3365 ;* OUTPUTS: R1 - CONTAINS ADDRESS OF ERROR MESSAGE TO BE REPORTED
3366 ;* CLEAR IF NO ERROR FOUND.
3367 ;* CARRY USED TO INDICATE IF FIFO WAS FOUND EMPTY, CARRY CLEAR.
3368 ;*
3369 ;* CALLING SEQUENCE: JSR PC,READ
3370 ;*
3371 ;* COMMENTS:
3372 ;*
3373 ;* SUBORDINATE ROUTINES CALLED: CHKBMP.
3374 ;-- *****
3375
3376 015246 004567 166616 READBX:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
015246 005001 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3377 015252 005001 CLR R1 ;CLEAR GPR THAT HOLDS THE ADDRESS OF ERRMSG.
3378 015254 016703 164760 MOV RBUFA,R3 ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
3379 015260 011302 2#: MOV (R3),R2 ;READ A CHARACTER FROM THE FIFO.
3380 015262 100015 BPL 8# ;BRANCH IF FIFO IS EMPTY.
3381 ;*
3382 ;* CHECK IF THE READ CHARACTER IS A BMP CODE.
3383 ;* IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
3384 ;* ABORT THE TEST.
3385 ;--
3386 015264 004767 176204 JSR PC,CHKBMP ;CHECK IF CHARACTER IS A BMP CODE.
3387 015270 103410 BCS 6# ;BRANCH IF A BMP CODE WAS FOUND.
3388 015272 120227 000021 CMPB R2,#21 ;CHECK IF IT IS AN XON.
3389 015276 001003 BNE 4# ;BRANCH IF NOT AN XON.
3390 015300 012701 007005 MOV #EM5402,R1 ;PASS THE MESSAGE TO BE REPORTED.
3391 015304 000402 BR 6# ;GO EXIT TEST.
3392 015306 005300 4#: DEC R0 ;DECREMENT THE READ COUNT.
3393 015310 001363 BNE 2#
3394 015312 000261 6#: SEC ;SET CARRY TO INDICATE SUCCESS.
3395 015314 000401 BR 60# ;EXIT
3396 015316 000241 8#: CLC ;CLEAR CARRY BIT TO INDICATE FAILURE.
3397
3398 015320 010166 000004 60#: PASS R1 ;RESTORE GPRS,
015320 004736 MOV R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
015324 000207 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
3399 015326 000207 RTS PC

```

```

3401 .SBTTL GLOBAL SUBROUTINE - RESETT -
3402 ;*****
3403 ;* - RESET DEVICE UNDER TEST -
3404 ;* THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
3405 ;* IF RESET DOES NOT SUCCESSFULLY COMPLETE, IE. TIME-OUT OCCURS, THEN
3406 ;* AN ABORT TEST ERROR MESSAGE IS REPORTED.
3407 ;*
3408 ;* INPUTS: CSRA - CONTAINS THE ADDRESS OF THE CSR
3409 ;* TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
3410 ;* ERRRTBL- ERRRTYP,ERRNBR,AND ERRMSG SET UP CORRECTLY.
3411 ;*
3412 ;* OUTPUTS: THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
3413 ;* CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
3414 ;* ERRBLK - VALUE MAY BE DESTROYED.
3415 ;* IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
3416 ;* TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.
3417 ;*
3418 ;* CALLING SEQUENCE: JSR PC,RESETT
3419 ;*
3420 ;* COMMENTS: THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS INITIAL ERRNBR
3421 ;* THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.
3422 ;*
3423 ;* SUBORDINATE ROUTINES CALLED: DELAY,MSLGET.
3424 ;*****
3425
3426 015330 RESETT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
3427 015330 004567 166534 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3428 ; MOV #BIT05,R2 ;SET BIT MASK OF MASTER RESET BIT.
3429 ;*
3430 ; TEST THE STATE OF THE MASTER RESET BIT IN THE CSR.
3431 ; IF MR IS SET THEN WAIT FOR SELF-TEST TO COMPLETE.
3432 ; IF TIME-OUT OCCURS, REPORT THE ERROR AND PASS-OUT ABORT TEST INDICATOR.
3433 015340 016704 164672 ;
3434 015344 030214 ; MOV CSRA,R4 ;GET THE ADDRESS OF THE DUT'S CSR.
3435 015346 001406 ; BIT R2,(R4) ;CHECK STATE OF MASTER RESET BIT.
3436 015350 005003 ; BEQ 2# ;DON'T DELAY IF MR IS ALREADY CLEAR.
3437 015352 012701 011610 ; CLR R3 ;SET UP DESIRED STATE OF MASTER RESET BIT.
3438 015356 004767 176740 ; MOV #5000.,R1 ;PASS TIME-OUT VALUE OF 5 SECONDS.
3439 015362 103012 ; JSR PC,MSLGET ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.
3440 ; BCC 4# ;GO REPORT ERROR IF TIMEOUT OCCURRED.
3441 ;*
3442 ; SET MASTER RESET BIT IN CSR. CLEAR TX AND RX ENABLE BITS, ETC.
3443 ; SKIP THE SELFTST.
3444 ; TIME-OUT OF 5 SECS, JUST IN CASE THE SELF-TEST EXECUTES.
3445 ;
3446 015364 010277 164646 2#: MOV R2,@CSRA ;SET MASTER RESET BIT, DISABLE TX AND RX INTS.
3447 015370 004767 000266 ; JSR PC,SKPSTS ;TRY TO SKIP THE SELFTST.
3448 ;*
3449 ; SET SELF-TEST TIME-OUT OF 5 SECONDS, AND WAIT FOR M.R TO CLEAR.
3450 ; IF TIME-OUT OCCURS, THEN REPORT THE FATAL ERROR AND PASS-OUT THE ABORT
3451 ; TEST INDICATOR.
3452 ;
3453 015374 005003 ; CLR R3 ;SET UP DESIRED STATE OF MASTER RESET BIT.
3454 015376 012701 011610 ; MOV #5000.,R1 ;PASS TIME-OUT VALUE OF 5 SECONDS.
3455 015402 004767 176714 ; JSR PC,MSLGET ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.
3456 015406 103410 ; BCS 6# ;SKIP ERROR REPORT IF MR CLEARED IN TIME.
    
```

```

3457
3458 ; SET UP ERROR MESSAGE TO REPORT "FATAL ERROR FOUND DURING RESET,TEST ABORTED".
3459 ; INDICATE TEST IS TO BE ABORTED BY CLEARING THE CARRY BIT.
3460 ;
3461 015410 012701 005661 48: MOV #EM1601,R1 ;PASS ERROR MESSAGE TO REPORT.
3462 015414 012767 011642 166444 MOV #ER1603,ERRBLK ;PASS ADDRESS OF ERROR HANDLING ROUTINE.
3463 ;REPORT ERROR "TIME-OUT OCCURRED WAITING FOR MASTER RESET TO CLEAR"
3464 ; "TEST ABORTED"
3465 015422 ERROR ; >>>> ERROR <<<<<
3466 015422 104460 ; INDICATE TEST IS TO BE ABORTED. TRAP C:ERROR
3467 015424 000241 CLC ;EXIT THIS SUBROUTINE, ABORT TEST INDICATOR.
3468 015426 000403 BR 60:
3469 ;
3470 ; CLEAR TX AND RX INTERRUPT ENABLE STATUS FLAGS IN IESTAT.
3471 ; EXIT WITH CONTINUE TEST INDICATOR SET (IE,CARRY SET).
3472 015430 005067 164630 68: CLR IESTAT ;CLEAR TX AND RX INTERRUPT STATUS FLAGS.
3473 015434 000261 SEC ;INDICATE SUCCESS, CONTINUE TEST.
3474 ;
3475 015436 004736 60: PASS ;RESTORE GPRS, PASS THE FOLLOWING INTACT:
3476 015436 JSR PC,@(SP); ;RETURN TO PREG05 SUBRT.
3477 015440 000207 RTS PC ;CARRY BIT:IF CLEAR,INDICATES ABORT TEST.
3478

```

```

3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498 015442 010046
3499 015444
      015444 104440
      015446 010046
3500 015450
      015450 012700 000340
      015454 104441
3501 015456 042767 137777 164600
3502 015464 016777 164574 164544
3503 015472
      015472 012600
      015474 104441
3504 015476 012600
3505 015500 000207

```

```

.SBTTL GLOBAL SUBROUTINE - RXIEO -
;*****
;* - RECEIVER INTERRUPT DISABLE -
;* THIS ROUTINE IS USED TO DISABLE RECEIVER INTERRUPTS IN THE DHU11.
;*
;* 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,RXIEO
;*
;* COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
;* THE DUT CSR ARE DESTROYED.
;*
;* SUBORDINATE ROUTINES CALLED: NONE.
;*****
RXIEO:: MOV RO,-(SP) ;SAVE CONTENTS OF RO ON THE STACK.
        GETPRI -(SP) ;SAVE PROCESSOR PRIORITY ON STACK.
;*****
        TRAP C#GPRI
        MOV RO,-(SP)
        TRAP C#GPRI
        MOV #PRI07,RO
        TRAP C#SPRI
        BIC #137777,IESTAT ;CLEAR RX.INT.ENBL BIT IN IESTAT.
        MOV IESTAT,@CSRA ;DISABLE RX INTERRUPTS.
        SETPRI (SP). ;ENABLE INTERRUPTS TO THE PROCESSOR AGAIN.
        MOV (SP).,RO
        TRAP C#SPRI
        MOV (SP).,RO ;RESTORE RO.
        RTS PC

```

```

3507 .SBTTL GLOBAL SUBROUTINE - SAVBMP -
3508 ;** *****
3509 ;* - SAVE BMP CODES ROUTINE -
3510 ;* THIS ROUTINE SAVES THE PARAMETER PASSED IN, ONTO THE BMP CODE QUEUE
3511 ;* TOGETHER WITH THE NUMBER OF THE CURRENTLY EXECUTING TEST.
3512 ;*
3513 ;* INPUTS: R2 - CONTAINS THE BMP CODE THAT IS TO BE PLACED ON THE QUEUE.
3514 ;* BMPCQP - CONTAINS ADDRESS OF NEXT LOCATION IN THE BMP QUEUE.
3515 ;* BMPCQB - LABEL AT BASE OF THE BMP CODE QUEUE.
3516 ;* BMPCQE - LABEL OF NEXT LOCATION AFTER THE END OF THE BMP QUEUE.
3517 ;* TSTNUM - CONTAINS THE NUMBER OF THE CURRENT TEST.
3518 ;*
3519 ;* OUTPUTS: BMPCQP - INCREMENTED BY 4.
3520 ;* THE CONTENTS OF THE BMP CODE QUEUE ARE UPDATED.
3521 ;*
3522 ;* CALLING SEQUENCE: JSR PC,SAVBMP
3523 ;*
3524 ;* COMMENTS: IF THE OVERFLOW OCCURS THEN THE LAST LOCATION WILL BE
3525 ;* OVERWRITTEN BY ANY SUBSEQUENT ATTEMPTS TO UPDATE THE QUEUE.
3526 ;*
3527 ;* SUBORDINATE ROUTINES CALLED: NONE.
3528 ;-- *****
3529
3530 015502 SAVBMP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
3531 015502 004567 166362 ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3532 015506 016704 164704 ;MOV BMPCQP,R4 ;GET THE POINTER TO THE NEXT LOCATION IN QUEUE.
3533 015512 116724 164544 ;MOVB TSTNUM,(R4). ;SAVE THE CURRENT TEST NUMBER ON THE QUEUE.
3534 015516 005204 ;INC R4 ;INCREMENT THE POINTER TO GIVE AN EVEN ADDRESS.
3535 015520 042702 177400 ;BIC #177400,R2 ;CLEAR THE UNWANTED BITS FROM THE BMP CODE.
3536 015524 010224 ;MOV R2,(R4). ;SAVE THE BMP CODE ON THE QUEUE.
3537 015526 020427 002620 ;CMP R4,#BMPCQE ;CHECK IF OVERFLOW WILL OCCUR THE NEXT TIME.
3538 015532 103402 ;BLO 2# ;GO SAVE THE POINTER IF WE WILL NOT OVERFLOW.
3539 015534 162704 000004 ;SUB #4,R4 ;RESET THE POINTER TO THE LAST LOCATION IN QUE.
3540 015540 010467 164652 2#: MOV R4,BMPCQP ;SAVE THE POINTER.
3541 015544 004736 60#: PASS ;RESTORE GPRS.
3542 015546 000207 RTS PC JSR PC,B(SP). ;RETURN TO PREG05 SUBRT.

```

3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579

015550
015550 004567 166314
015554 016701 164504
015560 012702 002620
015564 012703 000020
015570 050103
015572 010177 164440
015576 017722 164442
015602 005201
015604 020103
015606 002771
015610
015610 004736
015612 000207

```

.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.
;* FLSA - 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.
;--*****
SAVMST:: SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
;GET IE STATES FOR UPDATING IND.ADR.REG FIELD.
;SET UP STAT STORAGE POINTER TO BASE OF TABLE.
MOV IESTAT,R1
MOV @STSTB,R2
MOV @NUMLNS,R3
BIS R1,R3 ;FORM COMPLETION COMPARISON WORD.
20: MOV R1,@CSRA ;SET UP THE CSR IND.ADR.REG FIELD.
MOV @FLSA,(R2)+ ;SAVE CONTENTS OF THIS LINE'S STAT REGISTER.
INC R1 ;SET LINE COUNTER TO NEXT LINE.
CMP R1,R3 ;CHECK FOR ALL LINES DONE.
BLT 20 ;LOOP IF NOT ALL LINES DONE.
600: PASS ;RESTORE GPRS.
;PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC JSR

```

```

3581 .SBTTL GLOBAL SUBROUTINE - SETPAR -
3582 ;* *****
3583 ;* - SET TX AND CONTROL PARAMETERS -
3584 ;* THIS SUROUTINE IS USED IN THE FIHAVL.TST.
3585 ;* IT INITIALISES THE SELECTED LINE TO THE FOLLOWING STATE:
3586 ;* INTERNAL LOOPBACK, IAUTO ENABLED, LPR:38.4K, 8 BITS/CHAR, 2 STOP,
3587 ;* ODD PARITY.
3588 ;*
3589 ;* INPUTS: R1 - CONTAINS NUMBER OF THE LINE TO BE INITIALISED.
3590 ;*
3591 ;* OUTPUTS: LNCTRL AND LPR REGISTERS FOR THE SELECTED LINE ARE DESTROYED.
3592 ;*
3593 ;* CALLING SEQUENCE: JSR PC,SETPAR
3594 ;*
3595 ;* COMMENTS:
3596 ;*
3597 ;* SUBORDINATE ROUTINES CALLED: DELAY,WTWLNCR,WTWLPR.
3598 ;* -- *****
3599
3600 SETPAR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
3601 ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3602 JSR PC,LINBIT ;GET A BIT MAP FOR THIS LINE.
3603 MOV R0,R5 ;COPY THE LINE BIT MAP.
3604 MOV @206,R0 ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
3605 JSR PC,WTWLNCR ;INITILAISE THE LINE CONTROL REGISTER.
3606 MOV @177670,R0 ;PASS THE LPR CONTENTS.
3607 JSR PC,WTWLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
3608 MOV @10.,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
3609 JSR PC,DELAY ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
3610 600: PASS ;RESTORE GPRS.
3611 RTS PC JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
    
```

```

3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633 015662
      015662 004567 166202
3634 015666 012704 000012
3635 015672 004767 176034
3636
3637
3638
3639 015676 012701 000060
3640
3641
3642 015702 012703 052525
3643 015706 005301
3644 015710 016704 164322
3645 015714 010124
3646 015716 010324
3647 015720 020467 164330
3648 015724 103774
3649 015726 032701 000017
3650 015732 001365
3651
3652 015734
      015734 004736
3653 015736 000207

```

```

.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.
; -- *****
SKPSTS:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          MOV #10,R4 ;PASS DELAY VALUE OF 10 MILLI-SECONDS.
          JSR PC,DELAY ;DELAY FOR 10 MILLI-SECONDS.
; *
; * WRITE SKIP SELF-TEST CODE (52525) TO ALL THE INDEXED DUT REGISTERS.
; -
          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 DHU-11.
          MOV #52525,R3 ;INITIALISE THE SKIP SELF-TEST CODE.
40: DEC R1 ;SELECT THE NEXT SET OF DEVICE REGISTERS.
      MOV CSRA,R4 ;GET THE ADDRESS OF THE CSR OF THE DUT.
      MOV R1,(R4)+ ;SELECT A BANK OF DUT REGISTERS.
60: MOV R3,(R4)+ ;WRITE THE CODE TO A DUT REGISTER.
      CMP R4,TXBFCA ;COMPARE POINTER WITH LAST REGISTER ADDRESS.
      BLO 60 ;LOOP IF NOT ALL REGS DONE IN THIS BANK.
      BIT #17,R1 ;TEST FOR IND.ADR.REG FIELD DECREMENTED TO 0.
      BNE 40 ;LOOP UNTIL ALL REGISTERS CONTAIN THE CODE.
600: PASS ;RESTORE GPRS.
          JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
          RTS PC

```



```

3686 .SBTTL GLOBAL SUBROUTINE - TXDATP -
3687 ;* *****
3688 ;* - TRANSMIT DATA PATTERN -
3689 ;* THIS SUBROUTINE IS USED IN THE FIMAVL.TST.
3690 ;* IT TRANSMITS A SPECIFIED NUMBER OF DATA BYTES ON THE SPECIFIED LINE.
3691 ;*
3692 ;* INPUTS: R0 - CONTAINS THE NUMBER OF DATA BYTES TO TX.
3693 ;* R1 - CONTAINS LINE NUMB ON WHICH TRANSMISSION IS TO TAKE PLACE.
3694 ;* BUFBAS TO BUFMID CONTAINS A 256 BYTE DATA PATTERN.
3695 ;*
3696 ;* OUTPUTS: DATA IS SENT OUT ON THE SPECIFIED LINE.
3697 ;* CARRY SET = TX SUCCESSFUL.
3698 ;*
3699 ;* CALLING SEQUENCE: TXDATP
3700 ;*
3701 ;* COMMENTS:
3702 ;*
3703 ;* SUBORDINATE ROUTINES CALLED: DODMA.
3704 ;*-- *****
3705
3706 016052 TXDATP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
016052 004567 166012 ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3707 016056 010003 MOV R0,R3 ;PASS THE NUMBER OF CHARS TO TX.
3708 016060 012702 002660 MOV #BUFBAS,R2 ;PASS THE START OF THE DATA PATTERN TO TX.
3709 016064 004767 175702 JSR PC,DODMA ;TRANSMIT THE DATA PATTERN.
3710 016070 60: PASS ;RESTORE GPRS.
016070 004736 ;PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
3711 016072 000207 RTS PC

```

3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765

016074
016074 004567 165770
016100 010500
016102 012701 000001
016106 016702 164140
016112 005202
016114 012703 000020
016120 016704 164140
016124 005005

016126 010477 164104
016132 105712
016134 100001
016136 050105

016140 030100
016142 001402
016144 142712 000200
016150 005204
016152 006301
016154 005303
016156 001363

016160
016160 010566 000014
016164 004736

016166 000207

```
.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 TBUFAD2 REGISTER.
;+
;+ OUTPUTS: R5 - BIT'S SET INDICATE THE INITIAL STATES OF ALL TX.ENABLE BITS.
;+ TBUFAD2 - THE STATE OF THE TX.ENABLE 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.
;-- *****

TXDSBL:: SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV R5,R0 ;COPY BIT MAP OF LINES TO DISABLE TRANSMISSION.
MOV #BIT0,R1 ;INITIALIZE THE SELECTED LINE BIT MASK.
MOV TXAD2A,R2 ;GET THE ADDRESS OF THE TBUFAD2 REGISTER.
INC R2 ;GET THE ADDRESS OF THE MSBYTE OF TBUFAD2 REG.
MOV #NUMLNS,R3 ;GET MAXIMUM LINE NUMBER PLUS ONE.
MOV IESTAT,R4 ;GET THE STATES OF THE INT ENABLE BITS.
CLR R5 ;LOG POSSIBLE TX DISABLED ON ALL LINES.

;+
; SELECT EVERY LINE IN TURN, AND LOG THE STATE OF EACH TX.ENABLE BIT.
;-
20: MOV R4,BCSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
TSTB (R2) ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
BPL 40 ;SKIP NEXT INSTRUCTION IF TX.ENABLE CLEAR.
BIS R1,R5 ;LOG TX ENABLE BIT SET FOR SELECTED LINE.

;+
; CLEAR TX.ENABLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE TX DISABLE
; LINE BIT MAP.
;-
40: BIT R1,R0 ;CHECK STATE OF DISABLE LINE BIT MAP.
BEQ 60 ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
BICB #BIT7,(R2) ;CLEAR TX.ENABLE BIT ON SELECTED LINE.
INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
DEC R3 ;DECREMENT LINE NUMBER.
BNE 20 ;LOOP TO CHECK NEXT LINE.

60: PASS R5 ;RESTORE GPRS,EXCEPT
MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
;R5 - PREVIOUS STATES OF ALL TX.ENABLE BITS.

RTS PC
```

```

3767 .SBTTL GLOBAL SUBROUTINE - TXENBL -
3768 ;* *****
3769 ;* - TRANSMITTER ENABLE -
3770 ;* THIS SUBROUTINE IS USED TO ENABLE TRANSMISSION ON SELECTED LINES BY
3771 ;* SETTING THE ASSOCIATED TX.ENABLE BIT ON THE DUT.
3772 ;*
3773 ;* INPUTS: R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO SET TX.ENABLE.
3774 ;* CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
3775 ;* IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
3776 ;* NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
3777 ;* TXAD2A - CONTAINS THE ADDRESS OF THE TBUFAD2 REGISTER.
3778 ;*
3779 ;* OUTPUTS: R5 - BIT'S SET INDICATE PREVIOUSLY DISABLED LINES.
3780 ;* TBUFAD2 - THE STATE OF THE TX.ENABLE BIT MAY BE ALTERED.
3781 ;* THE CONTENTS OF THE IND.ADD.REG FIELD IN THE CSR ARE DESTROYED.
3782 ;*
3783 ;* CALLING SEQUENCE: JSR PC,TXENBL
3784 ;*
3785 ;* COMMENTS:
3786 ;*
3787 ;* SUBORDINATE ROUTINES CALLED: NONE.
3788 ;*
3789 ;* *****
3790 TXENBL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
016170 004567 165674 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
016170 010500 ;COPY BIT MAP OF LINES TO ENABLE.
3791 016174 012701 000001 MOV R5,R0
3792 016176 016702 164044 MOV #BIT0,R1 ;INITIALIZE THE SELECTED LINE BIT MASK.
3793 016202 005202 MOV TXAD2A,R2 ;GET THE ADDRESS OF THE TBUFAD2 REGISTER.
3794 016206 012703 000020 INC R2 ;GET THE ADDRESS OF THE MSBYTE OF TBUFAD2 REG.
3795 016210 016704 164044 MOV #NUMLNS,R3 ;GET MAXIMUM LINE NUMBER.
3796 016214 005005 MOV IESTAT,R4 ;GET THE STATES OF THE INT ENABLE BITS.
3797 016220 CLR R5 ;CLEAR TX.ENABLE BIT LOG OF DISABLED LINES.
3798 ;*
3799 ;* SELECT EVERY LINE IN TURN,AND LOG ANY TX.ENABLE BIT THAT IS CLEAR.
3800 ;*
3801 016222 010477 164010 2#: MOV R4,BCSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
3802 016226 105712 TSTB (R2) ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
3803 016230 100401 BMI 4# ;SKIP NEXT INSTRUCTION IF TX.ENABLE SET.
3804 016232 050105 BIS R1,R5 ;LOG TX ENABLE BIT CLEAR FOR SELECTED LINE.
3805 ;*
3806 ;* SET TX.ENABLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE TX ENABLE
3807 ;* LINE BIT MAP.
3808 ;*
3809 016234 030100 4#: BIT R1,R0 ;CHECK STATE OF TX.ENABLE LINE BIT MAP.
3810 016236 001402 BEQ 6# ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
3811 016240 152712 000200 BISB #BIT7,(R2) ;ENABLE TRANSMISSION ON SELECTED LINE.
3812 016244 005204 6#: INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
3813 016246 006301 ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
3814 016250 005303 DEC R3 ;DECREMENT LINE NUMBER.
3815 016252 001363 BNE 2# ;LOOP TO CHECK NEXT LINE.
3816 ;*
3817 016254 010566 000014 60#: PASS R5 ;RESTORE GPRS,EXCEPT
016254 004736 MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
016260 JSR PC,@(SP) ;RETURN TO PREG05 SUBRT.
3818 ;R5 - LINE BIT MAP CORRESPONDING TO THE
3819 ; PREVIOUS LINES THAT WERE DISABLED.
3820 016262 000207 RTS PC

```

```

3822 .SBTTL GLOBAL SUBROUTINE - TXIEO -
3823 ;* *****
3824 ;* - TRANSMITTER INTERRUPT DISABLE -
3825 ;* THIS ROUTINE IS USED TO DISABLE TRANSMITTER INTERRUPTS IN THE DHU11..
3826 ;*
3827 ;* INPUTS: NONE.
3828 ;*
3829 ;* OUTPUTS: THE TX.INT.ENBL BIT IS CLEARED IN THE DUT CSR.
3830 ;* IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
3831 ;* ENABLE BITS.
3832 ;*
3833 ;* CALLING SEQUENCE: JSR PC,TXIEO
3834 ;*
3835 ;* COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER*FIELD IN
3836 ;* THE DUT CSR ARE DESTROYED.
3837 ;*
3838 ;* SUBORDINATE ROUTINES CALLED: NONE.
3839 ;-- *****
3840 016264 010046 TXIEO:: MOV RO,-(SP) ;SAVE CONTENTS OF RO ON THE STACK.
3841 016266 104440 GETPRI -(SP) ;SAVE CURRENT PROCESSOR PRIORITY ON THE STACK.
3842 016270 010046 TRAP C#GPRI
3842 016272 SETPRI #PRI07 ;IGNORE ANY INTERRUPTS THAT MAY BE GENERATED.
3843 016300 042767 177677 163756 BIC #177677,IESTAT ;CLEAR TX.INT.ENBL BIT IN IESTAT.
3844 016306 016777 163752 163722 MOV IESTAT,@CSRA ;DISABLE TX INTERRUPTS.
3845 016314 012600 SETPRI (SP)+ ;ENABLE INTERRUPTS TO THE PROCESSOR AGAIN.
3846 016320 012600 MOV (SP)+,RO ;RESTORE RO.
3847 016322 000207 RTS PC TRAP C#SPRI
    
```

```

3849 .SBTTL GLOBAL SUBROUTINE - UNSDIV -
3850 ;* *****
3851 ;* - UNSIGNED DIVIDE ROUTINE -
3852 ;* THIS SUBROUTINE IS USED TO DIVIDE A 32 BIT UNSIGNED DIVIDEND BY A
3853 ;* 16 BIT UNSIGNED DIVISOR GIVING A 16 BIT QUOTIENT. ALL NUMBERS ARE
3854 ;* CONSIDERED TO BE UNSIGNED. A SUCCESS FLAG IS NOT SET ON RETURN IF
3855 ;* THE QUOTIENT WAS TOO BIG TO BE CONTAINED IN 16 BITS.
3856 ;*
3857 ;* INPUTS: R1 - THE DIVISOR, UNSIGNED, 16 BITS.
3858 ;* R2 - MOST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
3859 ;* R3 - LEAST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
3860 ;*
3861 ;* OUTPUTS: R1 - QUOTIENT, UNSIGNED, 16 BITS (177777 IF OVERFLOW).
3862 ;* CARRY - SUCCESS FLAG, SET IF COMPLETE QUOTIENT FITS IN 16 BITS.
3863 ;*
3864 ;* CALLING SEQUENCE: JSR PC,UNSDIV
3865 ;*
3866 ;* COMMENTS: IF THE DIVISOR IS 0 THE QUOTIENT IS RETURNED AS ALL ONES
3867 ;* (177777) AND THE CARRY IS CLEAR REGARDLESS OF THE DIVIDEND.
3868 ;*
3869 ;* SUBORDINATE ROUTINES CALLED: NONE.
3870 ;* -- *****
3871
3872 016324 UNSDIV:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
016324 004567 165540 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3873
3874 ;* CHECK FOR QUOTIENT GREATER THAN 16 BITS CONDITION.
3875 ;*
3876 016330 010204 ;GET MSW OF DIVIDEND FOR SUBTRACT.
3877 016332 160104 ;SUBTRACT DIVISOR FROM MSW OF DIVIDEND.
3878 016334 103403 ;IF IT DIDN'T GO, WE HAVE QUOTIENT < 16 BITS.
3879 016336 012701 177777 ;SET QUOTIENT TO ALL ONES (177777).
3880 016342 000442 ;EXIT WITH CARRY CLEAR.
3881
3882 ;* SET UP COUNTERS AND VARIOUS WORKING GPRS.
3883 ;*
3884 016344 005004 ;CLEAR THE LSM OF THE DIVISOR.
3885 016346 000241 ;CLEAR CARRY FOR THE SHIFT OF THE DIVISOR.
3886 016350 006001 ; DIVISOR BY
3887 016352 006004 ; 2(UNSIGNED)
3888 016354 012700 000020 ;SET UP INITIAL SHIFT COUNT TO 16.
3889
3890 ;* THE SUBTRACT AND SHIFT LOOP.
3891 ;*
3892 016360 010246 ;SAVE MSWORD OF DIVIDEND.
3893 016362 010346 ;SAVE LSWORD OF DIVIDEND.
3894 016364 160403 ;LSWORD DIVIDEND - LSWORD OF DIVISOR.
3895 016366 005602 ;MSWORD DIVIDEND - BORROW.
3896 016370 103402 ;IF BORROW FROM BORROW SUBTRACT, IT DIDN'T GO.
3897 016372 160102 ;MSWORD DIVIDEND - MSWORD OF DIVISOR.
3898 016374 103003 ;IF NO BORROW, IT WENT, CARRY IS CLEAR.
3899
3900 ;* IT DIDN'T GO, SO WE SHIFT A 1 INTO THE QUOTIENT (COMPLEMENTED LATER).
3901 ;* CARRY IS SET.
3902 ;*
3903 016376 012603 ;RESTORE LSWORD OF DIVIDEND.
3904 016400 012602 ;RESTORE MSWORD OF DIVIDEND.
    
```

```

3905 016402 000401          BR      10$          ;GOTO SHIFT 1 INTO THE QUOTIENT.
3906
3907          ;+
3908          ; IT WENT, SO WE RESTORE THE STACK AND SHIFT A 0 INTO QUOTIENT (WILL BE
3909          ; COMPLEMENTED LATER).  CARRY IS CLEAR.
3910 016404 012626      8$:  MOV      (SP)+,(SP)+      ;POP THE SAVED DIVIDEND OFF OF THE STACK.
3911          ;+
3912          ; SHIFT THE RESULT OF THE SUBTRACT ATTEMPT INTO THE QUOTIENT SHIFT REG.
3913          ;-
3914 016406 006105      10$:  ROL      R5          ;SHIFT NEXT BIT INTO THE INVERTED QUOTIENT.
3915 016410 000241          CLC          ;DIVIDE THE
3916 016412 006001          ROR      R1          ; DEVISOR BY
3917 016414 006004          ROR      R4          ; 2 (UNSIGNED).
3918 016416 005300          DEC      R0          ;COUNT THIS SHIFT AND SUBTRACT.
3919 016420 001357          BNE     4$          ;LOOP FOR ANOTHER SHIFT & SUB IF NOT DONE.
3920 016422 005105          COM     R5          ;GET QUOTIENT FROM INVERTED QUOTIENT.
3921          ;+
3922          ; NOW WE EITHER ROUND UP OR LEAVE QUOTIENT ALONE.
3923          ;-
3924 016424 000241          CLC          ;CLEAR THE CARRY FOR THE SHIFT OF THE DIVIDEND.
3925 016426 006103          ROL     R3          ;MULTIPLY LSWORD OF DIVIDEND BY 2, MSWORD IS 0.
3926 016430 103402          BCS     12$        ;IF CARRY FROM SHIFT, ROUND UP.
3927 016432 160403          SUB     R4,R3      ;SUBTRACT DIVISOR FROM DIVIDEND.
3928 016434 103403          BCS     14$        ;IF BORROW, DON'T ROUND UP.
3929          ;+
3930          ; ROUND UP, EXTRA SUBTRACT WENT.
3931          ;-
3932 016436 005205      12$:  INC     R5          ;INCREMENT THE QUOTIENT BY ONE.
3933 016440 001001          BNE     14$        ;IF NO OVERFLOW, WE LEAVE THE ROUND UP.
3934 016442 005305          DEC     R5          ;DON'T LET ROUNDING CAUSE OVERFLOW.
3935          ;+
3936          ; ALL DONE, PASS QUOTIENT AND EXIT.
3937          ;-
3938 016444 010501      14$:  MOV     R5,R1      ;PASS QUOTIENT BACK IN R1.
3939 016446 000261          SEC          ;INDICATE NO OVERFLOW.
3940
3941 016450          60$:  PASS     R1          ;RESTORE GPRS, LEAVE THE FOLLOWING INTACT:
          016450 010166 000004      MOV     R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
          016454 004736          JSR     PC,0(SP)+ ;RETURN TO PREGOS SUBRT.
3942          ;R1 - 16 BIT, UNSIGNED QUOTIENT.
3943 016456 000207          RTS     PC          ;CARRY - SET INDICATES NO OVERFLOW (SUCCESS).

```

```

3945 .SBTTL GLOBAL SUBROUTINE - WAIBIC -
3946 ;* *****
3947 ;* - WAIT FOR BIT CLEAR ROUTINE -
3948 ;* THIS SUBROUTINE WAITS FOR THE SPECIFIED BIT TO BECOME CLEAR. IF THE
3949 ;* SPECIFIED BIT GOES TO A CLEAR STATE WITHIN THE SPECIFIED TIME-OUT
3950 ;* PERIOD A SUCCESS INDICATION IS RETURNED BY THIS ROUTINE.
3951 ;* THE LAST VALUE WHICH IS READ LOOKING FOR THE CONDITION IS RETURNED TO
3952 ;* ALLOW THE USE OF THIS ROUTINE TO LOOK FOR DESTRUCTIVE READ CONDITIONS.
3953 ;*
3954 ;* INPUTS: R1 - TIME-OUT VALUE AND BIT NUMBER INDICATION;
3955 ;* BITS 15 THRU 12 - NUMBER OF BIT TO TEST (RANGE 0 THRU 15).
3956 ;* BITS 11 THRU 0 - TIME-OUT VALUE IN MILLI-SECONDS (4095 MAX).
3957 ;* R2 - ADDRESS OF WORD CONTAINING THE BIT TO TEST.
3958 ;* MSLCNT.
3959 ;*
3960 ;* OUTPUTS: R2 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
3961 ;* CARRY - SUCCESS FLAG (CARRY SET IF BIT CLR BEFORE TIME-OUT).
3962 ;*
3963 ;* CALLING SEQUENCE: MOV #130040,R1 ;PASS BIT 11 (13 OCTAL) AND
3964 ;* ; 32 (40 OCTAL) MS DELAY.
3965 ;* MOV #LABEL,R2 ;TEST BIT IN WORD AT "LABEL".
3966 ;* JSR PC,WAIBIC ;WAIT 32 MS FOR BIT 11 TO CLR.
3967 ;*
3968 ;* COMMENTS:
3969 ;*
3970 ;* SUBORDINATE ROUTINES CALLED: MSLGET.
3971 ;*-- *****
3972
3973 016460 WAIBIC:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
016460 004567 165404 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3974 016464 010204 MOV R2,R4 ;SET UP THE ADDRESS PARAMETER FOR MSLGET.
3975 016466 010102 MOV R1,R2
3976 016470 042701 170000 BIC #170000,R1 ;SEPERATE DELAY COUNT OUT OF PASSED PARAMETER.
3977 016474 042702 007777 BIC #7777,R2 ;SEPERATE LINE NUMBER FIELD OF PASSED PARAM.
3978 016500 000302 SWAB R2 ;PUT LINE NUMBER FIELD IN LSBYTE.
3979 016502 006202 ASR R2 ;SHIFT THE LINE NUMBER FIELD INTO THE PROPER
3980 016504 006202 ASR R2 ; POSITION TO USE IT AS A WORD TABLE OFFSET
3981 016506 006202 ASR R2 ; FOR THE TABLE LOOKUP OF THE LINE BIT MAP.
3982 016510 016202 002344 MOV BITTBL(R2),R2 ;GET BIT MAP OF LINE TO TEST FROM TABLE.
3983 016514 005003 CLR R3 ;INDICATE THAT THE BIT SHOULD BE CLR.
3984 016516 004767 175600 JSR PC,MSLGET ;WAIT FOR THE BIT TO BE CLR WITHIN TIME-OUT.
3985 ; CARRY IS CORRECT UPON MSLGET RETURN.
3986 016522 010002 MOV R0,R2 ;PASS LAST VALUE READ AS OUTPUT PARAMETER.
3987 016524 010266 000006 601: PASS R2 ;RESTORE GPRS, EXCEPT THE FOLLOWING:
016530 004736 MOV R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
3988 JSR PC,@(SP) ;RETURN TO PREG05 SUBRT.
3989 016532 000207 RTS PC ; R2 - LAST VALUE READ LOOKING FOR CONDITION.
; CARRY - SUCCESS FLAG (SET IF BIT FOUND CLR).
    
```

```

3991 .SBTTL GLOBAL SUBROUTINE - WAIBIS -
3992 ;* *****
3993 ;* - WAIT FOR BIT SET ROUTINE -
3994 ;* THIS SUBROUTINE WAITS FOR THE SPECIFIED BIT TO BECOME SET. IF THE
3995 ;* SPECIFIED BIT GOES TO A SET STATE WITHIN THE SPECIFIED TIME-OUT
3996 ;* PERIOD A SUCCESS INDICATION IS RETURNED BY THIS ROUTINE.
3997 ;* THE LAST VALUE WHICH IS READ LOOKING FOR THE CONDITION IS RETURNED TO
3998 ;* ALLOW THE USE OF THIS ROUTINE TO LOOK FOR DESTRUCTIVE READ CONDITIONS.
3999 ;*
4000 ;* INPUTS: R1 - TIME-OUT VALUE AND BIT NUMBER INDICATION:
4001 ;* BITS 15 THRU 12 - NUMBER OF BIT TO TEST (RANGE 0 THRU 15).
4002 ;* BITS 11 THRU 0 - TIME-OUT VALUE IN MILLI-SECONDS (4095 MAX).
4003 ;* R2 - ADDRESS OF WORD CONTAINING THE BIT TO TEST.
4004 ;* MSLCNT.
4005 ;*
4006 ;* OUTPUTS: R2 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
4007 ;* CARRY - SUCCESS FLAG (CARRY SET IF BIT SET BEFORE TIME-OUT).
4008 ;*
4009 ;* CALLING SEQUENCE: MOV #130040,R1 ;PASS BIT 11 (13 OCTAL) AND
4010 ;* ; 32 (40 OCTAL) MS DELAY.
4011 ;* MOV #LABEL,R2 ;TEST BIT IN WORD AT "LABEL".
4012 ;* JSR PC,WAIBIS ;WAIT 32 MS FOR BIT 11 TO SET.
4013 ;*
4014 ;* COMMENTS:
4015 ;*
4016 ;* SUBORDINATE ROUTINES CALLED: MSLGET.
4017 ;* -- *****
4018
4019 016534 WAIBIS:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
4020 016534 004567 165330 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4021 016542 010102 MOV R2,R4 ;SET UP THE ADDRESS PARAMETER FOR MSLGET.
4022 016544 042701 170000 MOV R1,R2
4023 016550 042702 007777 BIC #170000,R1 ;SEPERATE DELAY COUNT OUT OF PASSED PARAMETER.
4024 016554 000302 SWAB R2 ;SEPERATE LINE NUMBER FIELD OF PASSED PARAM.
4025 016556 006202 ASR R2 ;PUT LINE NUMBER FIELD IN LSBYTE.
4026 016560 006202 ASR R2 ;SHIFT THE LINE NUMBER FIELD INTO THE PROPER
4027 016562 006202 ASR R2 ; POSITION TO USE IT AS A WORD TABLE OFFSET
4028 016564 016202 002344 MOV BITTBL(R2),R2 ; FOR THE TABLE LOOKUP OF THE LINE BIT MAP.
4029 016570 010203 MOV R2,R3 ;GET BIT MAP OF LINE TO TEST FROM TABLE.
4030 016572 004767 175524 JSR PC,MSLGET ;INDICATE THAT THE BIT SHOULD BE SET.
4031 ;WAIT FOR THE BIT TO BE SET WITHIN TIME-OUT.
4032 016576 010002 MOV R0,R2 ; CARRY IS CORRECT UPON MSLGET RETURN.
4033 016600 010266 000006 601: PASS R2 ;PASS LAST VALUE READ AS OUTPUT PARAMETER.
4034 016604 004736 MOV R2,R2SLOT(SP) ;RESTORE GPRS, EXCEPT THE FOLLOWING:
4035 016606 000207 RTS PC JSR PC,B(SP)+ ;PUT R2 IN STACK SLOT.
;RETURN TO PREG05 SUBRT.
; R2 - LAST VALUE READ LOOKING FOR CONDITION.
; CARRY - SUCCESS FLAG (SET IF BIT FOUND SET).
    
```

4037
4038
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4050
4051
4052
4053
4054
4055
4056 016610
4057 016614 004567 165254
4058 016614 012701 170536
4059 016620 016702 163412
4060 016624 004767 177704
4061 016630 103005
4062 016632 012704 000005
4063 016636 004767 175070
4064 016642 000261
4065 016644
4066 016644 004736
4067 016646 000207

```

.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.
;*****
WAITTX:: SAVE
;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
;PASS TIME-OUT VALUE OF 350 MILLI SECS.
;PASS THE ADDRESS OF THE CSR.
;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
;BRANCH IF NO TX_ACTION, ABORT THE TEST.
;PASS DELAY OF 5 MILLI SECS.
;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
;SET CARRY TO INDICATE SUCCESS.

60: PASS
;RESTORE GPRS.
;PC,8(SP); ;RETURN TO PREG05 SUBRT.
;PASS THE CARRY BIT, SET INDICATES SUCCESS.

RTS PC

```

```

4069 .SBTTL GLOBAL SUBROUTINE - WTWLNC -
4070 ;** *****
4071 ;* - LINE CONTROL REGISTER SETUP ROUTINE -
4072 ;* THIS SUBROUTINE IS USED TO SET THE DEVICE UNDER TEST (DUT) LINE
4073 ;* CONTROL REGISTERS (LNCTRL) TO THE SPECIFIED STATE. ONLY THE LNCTRLS
4074 ;* FOR THE SPECIFIED LINES ARE ALTERED.
4075 ;*
4076 ;* INPUTS: R0 - NEW LINE PARAMETERS.
4077 ;* R5 - BIT MAP OF LINES TO BE ALTERED.
4078 ;* CSRA - CONTAINS ADDRESS OF THE DUT CSR.
4079 ;* IESTAT - CONTAINS THE CURRENT STATE OF THE TX AND RX INTERRUPT
4080 ;* ENABLE BITS IN THE CSR.
4081 ;* LNCTRA - CONTAINS ADDRESS OF THE DUT LNCTRL REGISTERS.
4082 ;*
4083 ;* OUTPUTS: LNCTRL - SPECIFIED DUT LINE CONTROL REGISTERS ARE ALTERED.
4084 ;*
4085 ;* CALLING SEQUENCE: JSR PC,WTWLNC
4086 ;*
4087 ;* COMMENTS:
4088 ;*
4089 ;* SUBORDINATE ROUTINES CALLED: ALTFLD.
4090 ;-- *****
4091
4092 016650 WTWLNC:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
016650 004567 165214 R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4093
4094 ;*
4095 ;* SET UP THE PARAMETERS FOR THE CALL TO ALTFLD.
4096 ;*
4096 016654 016701 163366 MOV LNCTRA,R1 ;SET UP THE REGISTER ADDRESS PARAMETER.
4097 016660 010002 MOV R0,R2 ;SET UP THE DESIRED REGISTER CONTENTS.
4098 016662 010503 MOV R5,R3 ;SET UP THE BIT MAP OF LINES TO ALTER.
4099 016664 012704 177777 MOV #-1,R4 ;SELECT ALL REGISTER BITS TO BE ALTERED.
4100 ;*
4101 ;* CALL THE SUBROUTINE WHICH ALTERS THE REGISTER CONTENTS.
4102 ;*
4103 016670 004767 174172 JSR PC,ALTFLD ;ALTER THE REGISTER CONTENTS.
4104 ;*
4105 016674 PASS 606: PASS JSR ;RESTORE GPRS.
016674 004736 PC,B(SP)+ ;RETURN TO PREG05 SUBRT.
4106 016676 000207 RTS PC

```

4108
4109
4110
4111
4112
4113
4114
4115
4116
4117
4118
4119
4120
4121
4122
4123
4124
4125
4126
4127
4128
4129
4130
4131 016700
016700 004567 165164
4132
4133
4134
4135 016704 016701 163332
4136 016710 010002
4137 016712 010503
4138 016714 012704 177777
4139
4140
4141
4142 016720 004767 174142
4143
4144 016724
016724 004736
4145 016726 000207

```

.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.
;-- *****
WTWLPR:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
R5,PREG05 ;CALL REGISTER SAVE SUBRT.
;+
; SET UP THE PARAMETERS FOR THE CALL TO ALTFLD.
;--
MOV LPRA,R1 ;SET UP THE REGISTER ADDRESS PARAMETER.
MOV R0,R2 ;SET UP THE DESIRED REGISTER CONTENTS.
MOV R5,R3 ;SET UP THE BIT MAP OF LINES TO ALTER.
MOV #-1,R4 ;SELECT ALL REGISTER BITS TO BE ALTERED.
;+
; CALL THE SUBROUTINE WHICH ALTERS THE REGISTER CONTENTS.
;--
JSR PC,ALTFLD ;ALTER THE REGISTER CONTENTS.
600: PASS ;RESTORE GPRS.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC

```

```

4147 .SBTTL INTERRUPT SERVICE ROUTINE - CLKINT -
4148 ;** *****
4149 ;* THIS ROUTINE IS EXECUTED CLKHRZ TIMES PER SECOND. IT DECREASES THE
4150 ;* TWO TIMER COUNTERS DOWN TO ZERO.
4151 ;*
4152 ;* INPUTS: TIMER1 - TIMER COUNTER #1.
4153 ;* TIMER2 - TIMER COUNTER #2.
4154 ;* TIMERS3 - TIMER COUNTER FOR CALL OF BREAK MACRO.
4155 ;*
4156 ;* OUTPUTS: THE 2 TIMER COUNTERS ARE DECREMENTED IF THEY ARE NOT ZERO.
4157 ;*
4158 ;* CALLING SEQUENCE: PUT #CLKINT IN THE CLOCK INTERRUPT VECTOR SLOT.
4159 ;* PUT THE DESIRED TIME PERIOD (SECONDS TIMES CLKHRZ) IN
4160 ;* EITHER TIMER1 OR TIMER2 AND POLL THE RESPECTIVE TIMER
4161 ;* COUNTER TO DETECT ITS GOING TO 0 ON TIME-OUT.
4162 ;*
4163 ;* COMMENTS: THE 2 COUNTERS WILL NOT WRAPAROUND BUT WILL STOP AT 0. THIS
4164 ;* ALLOWS THE DETECTION OF A TIME-OUT ANY TIME AFTER THE TIME-OUT
4165 ;* HAS OCCURRED UNTIL THE TIMER COUNTER IS SET TO ANOTHER VALUE.
4166 ;*
4167 ;* SUBORDINATE ROUTINES CALLED: NONE.
4168 ;-- *****
4169
4170 016730 005767 163366 CLKINT:: TST TIMER1 ;CHECK FOR TIMER1 AT ZERO.
4171 016734 001402 BEQ 2# ;BRANCH TO LEAVE IT AT ZERO IF IT IS ZERO.
4172 016736 005367 163360 DEC TIMER1 ;DECREMENT TIME COUNT.
4173 016742 005767 163356 2# TST TIMER2 ;CHECK FOR TIMER2 AT ZERO.
4174 016746 001402 BEQ 4# ;BRANCH TO LEAVE IT ALONE IF IT'S ALREADY ZERO.
4175 016750 005367 163350 DEC TIMER2 ;DECREMENT TIME COUNT.
4176 016754 005367 163346 4# DEC TIMER3 ;DECREMENT THE BREAK COUNT.
4177 016760 001006 BNE 60# ;EXIT IF NOT TIME TO CALL BREAK.
4178 016762 016767 163342 163336 MOV BCOUNT,TIMER3 ;SET UP TIME TILL NEXT BREAK.
4179 016770 010046 MOV RO,-(SP) ;SAVE CONTENTS OF RO FROM BREAK MACRO.
4180 016772 BREAK ;CHECK FOR OPERATOR CONTROL/C.
4181 016774 012600 TRAP C#BRK
4182 016776 000002 60# MOV (SP)+,RO ;RESTORE CONTENTS OF RO.
RTI
    
```

```

4184      .SBTTL GLOBAL TRAP SERVICE ROUTINE - TP4RTN -
4185      ;*****
4186      ;* BUS TIME-OUT TRAP (004 TRAP) SERVICE ROUTINE -
4187      ;* THIS ROUTINE DETERMINES IF THE 004 TRAP WAS CAUSED BY
4188      ;* AN "EXPECTED" ERROR OR NOT BY EXAMINING THE RETURN PC VALUE ON THE
4189      ;* STACK. IF THE TRAP IS UNEXPECTED, THIS ROUTINE JUMPS TO THE NORMAL
4190      ;* DIAGNOSTIC SUPERVISOR 004 TRAP HANDLING ROUTINE.
4191      ;*
4192      ;*
4193      ;* INPUTS:      SP - POINTS TO THE PC WHERE THE TRAP OCCURED.
4194      ;*              ADRPTR - LABEL AT THE ADDRESS WHERE "EXPECTED" TRAPS OCCUR.
4195      ;*              TP4FLG - 004 TRAP FLAGS.
4196      ;*
4197      ;* OUTPUTS:     TP4FLG - BIT 15 IS SET IF "EXPECTED" TRAP OCCURED.
4198      ;*
4199      ;* CALLING SEQUENCE:  PUT ADDRESS POINTED TO BY TP4RTN IN 004 VECTOR.
4200      ;*              OCCURENCE OF 004 TRAP VECTORS TO THIS ROUTINE.
4201      ;*
4202      ;* COMMENTS:     ANY 004 TRAP WHICH OCCURS AT AN ADDRESS OTHER THAN THAT LABELED
4203      ;*              ADRPTR WILL BE HANDLED BY THE NORMAL 004 TRAP SERVICE ROUTINE.
4204      ;*
4205      ;* SUBORDINATE ROUTINES CALLED: NONE.
4206      ;*****
4207
4208 017000 021627 013556      TP4RTN:: CMP      (SP),#ADRPTR      ;COMPARE EXPECTED ADR AGAINST TRAP RET PC.
4209 017004 001402              BEQ      2#              ;IF THEY MATCH, CONTINUE THIS ROUTINE.
4210 017006 000177 163272      JMP      @TP4VEC      ;IF NOT, JUMP TO NORMAL 004 TRAP SERVICE RTN.
4211 017012 052767 100000 163266 2#:  BIS      @BIT15,TP4FLG      ;SET THE 004 TRAP OCCURED FLAG.
4212 017020 000002              RTI              ;ALL DONE, GO BACK TO THE TEST.

```

```

4214
4215 *****
4216 :
4217 :           FVTA.RPT
4218 :
4219 :*****
4220
4221
4222 /
4223 .SBTTL REPORT CODING SECTION
4224
4225 :**
4226 : THE REPORT CODING SECTION CONTAINS THE
4227 : "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
4228 :--
4229
4230 017022      BGNRPT
4231 017022
4232 017022      EXIT  RPT
4233 017022 000167
4234 017024 000000
4235
4236 017026      ENDRPT
4237 017026
4238 017026 104425
4239
4240
4241
4242
4243
4244
4245
4246
4247
4248
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4310
4311
4312
4313
4314
4315
4316
4317
4318
4319
4320
4321
4322
4323
4324
4325
4326
4327
4328
4329
4330
4331
4332
4333
4334
4335
4336
4337
4338
4339
4340
4341
4342
4343
4344
4345
4346
4347
4348
4349
4350
4351
4352
4353
4354
4355
4356
4357
4358
4359
4360
4361
4362
4363
4364
4365
4366
4367
4368
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4380
4381
4382
4383
4384
4385
4386
4387
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4470
4471
4472
4473
4474
4475
4476
4477
4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500

```

```

4238 .SBTTL PROTECTION TABLE
4239
4240 ;*****
4241 ;
4242 ; FVTSKL4.P11
4243 ;
4244 ;*****
4245
4246
4247
4248
4249 ;***
4250 ; THIS TABLE IS USED BY THE RUNTIME SERVICES
4251 ; TO PROTECT THE LOAD MEDIA.
4252 ;--
4253 017030 BGNPROT
4254 017030 L#PROT::
4255 017030 177777 -1 ;OFFSET INTO P-TABLE FOR CSR ADDRESS
4256 017032 177777 -1 ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
4257 017034 177777 -1 ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
4258
4259 017036 ENDPROT
4260

```

4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295
4296 017036
017036
4297
4298 017036
017036 012700 000040
017042 104447
4299 017044
017044 103416
4300
4301 017046
017046 012700 000037
017052 104447
4302 017054
017054 103556
4303
4304 017056
017056 012700 000035
017062 104447
4305 017064
017064 103555
4306
4307 017066
017066 012700 000036
017072 104447
4308 017074
017074 103161
4309 017076 000167 000544
4310 017102
4311 017102
017102 104433
4312
4313
4314
4315 017104
017104 012700 000114
017110 104462

```

*****
:
:           FVTA.INI
:
*****

.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.
:*****
:--
:           BGNINIT
:
:           L$INIT::
:SEE IF PROGRAM JUST STARTED, BR IF YES
:           READEF  #EF.START
:
:           BCOMPLETE    NEWSTA
:
:SEE IF PROGRAM JUST RESTARTED, BR IF YES
:           READEF  #EF.RESTART
:
:           BCOMPLETE    NEWRES
:
:SEE IF THIS IS A NEW PASS, BR IF YES
:           READEF  #EF.NEW
:
:           BCOMPLETE    NEWPAS
:
:SEE IF PROGRAM WAS JUST CONTINUED
:           READEF  #EF.CONTINUE
:
:           BNCOMPLETE    GETPRM
:           JMP          ENDIT
NEWSTA:
:           BRESET
:           ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
:           TRAP        C$RESET
:
:           ;*
:           ; SET UP FOR LINE TIME CLOCK INTERRUPTS.
:           ;--
:           CLOCK    L,R1
:           ;GET THE CLOCK PARAMETERS.
:           MOV      #'L,R0
:           TRAP    C$CLCK

```

```

017112 010001
4316 017114 012167 163170      MOV      (R1)+,CLKCSR      ;STORE CLOCK CSR ADDRESS.
4317 017120 012167 163170      MOV      (R1)+,CLKBRL     ;STORE CLOCK BUS REQ INT LEVEL.
4318 017124 012167 163166      MOV      (R1)+,CLKVEC     ;STORE CLOCK INTERRUPT VECTOR.
4319 017130 012167 163164      MOV      (R1)+,CLKHRZ     ;STORE CLOCK FREQUENCY.
4320 017134 026727 163160 000062  CMP      CLKHRZ,#50.      ;TEST FOR 50HZ LINE FREQUENCY.
4321 017142 001004              BNE      2#               ;BRANCH IF CLOCK IS NOT 50HZ.
4322 017144 012767 000024 163160  MOV      #20.,MSTICK      ;INDICATE 20MS PER CLOCK TICK.
4323 017152 000403              BR       4#
4324 017154 012767 000021 163150 2#:  MOV      #17.,MSTICK      ;INDICATE 17 MS PER CLOCK TICK.
4325 017162 012767 000021 163150 4#:  SETVEC  CLKVEC,#CLKINT,PRI06 ;INITIALIZE CLOCK INTERRUPT VECTOR.
                                MOV      PRI06,-(SP)
                                MOV      #CLKINT,-(SP)
                                MOV      CLKVEC,-(SP)
                                MOV      #3,-(SP)
                                TRAP     C#SVEC
                                ADD      #10,SP
4326 017210 016700 163104      MOV      CLKHRZ,RO        ;INITIALIZE THE BREAK COUNT
4327 017214 006300              ASL      RO               ; TO CAUSE A BREAK
4328 017216 010067 163106      MOV      RO,BCOUNT        ; EVERY 2 SECONDS.
4329 017222 012700 000240      SETPRI   #PRI05           ;ALLOW CLOCK INTERRUPTS DISABLE OTHERS.
                                MOV      #PRI05,RO
                                TRAP     C#SPRI
4330
4331 ;*
4332 ; ENABLE THE LINE TIME CLOCK (LTC) CHECKING TO MAKE SURE THAT THE CSR
4333 ; IS ACCESSABLE.
4334 ; FIRST SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:
4335 017230 016767 160550 163046      MOV      4,TP4VEC         ;SAVE THE EXISTING 004 TRAP VECTOR.
4336 017236 012767 017000 160540      MOV      #TP4RTN,4        ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
4337
4338 ;*
4339 ; ENABLE LTC CHECKING FOR 004 TRAP IN CASE CSR IS NOT THERE.
4340 017244 005067 163036              CLR      TP4FLG           ;CLEAR THE 004 TRAP FLAG.
4341 017250 012767 000100 163032      MOV      #BIT6,WORD1      ;SET UP TO SET BIT6 OF THE LTC CSR.
4342 017256 012700 002310              MOV      #WORD1,RO        ;SET UP WORD1 AS THE CKTRAP MOVE SOURCE.
4343 017262 016701 163024              MOV      CLKCSR,R1        ;SET UP LTC CSR AS DESTINATION FOR CKTRAP MOVE.
4344 017266 004767 174252              JSR      PC,CKTRAP        ;MOVE AND CHECK FOR TRAP.
4345 017272 016767 163006 160504      MOV      TP4VEC,4         ;RESTORE THE NORMAL 004 TRAP VECTOR.
4346 017300 103403              BCS     6#               ;IF NO TRAP, LTC IS THERE SO CONTINUE.
4347 017302 005067 163012              CLR      CLKHRZ           ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
4348 017306 000402              BR       8#               ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
4349
4350 ;*
4351 ; CALIBRATE THE DELAY ROUTINE MILLI-SECOND DELAY COUNT VALUE.
4352 017310 004767 173734      6#:  JSR      PC,CALMSL
4353
4354 ;*
4355 ; CHECK FOR MEMORY MANAGEMENT PRESENT ON THIS MACHINE.
4356 ; IF MEM MGT IS PRESENT, DISABLE IT.
4357 017314 016767 160464 162762 8#:  MOV      4,TP4VEC         ;SAVE THE EXISTING 004 TRAP VECTOR.
4358 017322 012767 017000 160454      MOV      #TP4RTN,4        ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
4359 017330 005067 162752              CLR      TP4FLG           ;CLEAR THE 004 TRAP FLAG.
4360 017334 005067 162750              CLR      WORD1            ;PREPARE TO CLEAR THE MEM MGT SRO REGISTER.
4361 017340 012700 002310              MOV      #WORD1,RO        ;SELECT CLEARED WORD AS CKTRAP RTN SOURCE.
4362 017344 016701 162766              MOV      #MSRO,R1         ;SELECT MEM MGT SRO REGISTER AS DESTINATION.
4363 017350 005067 162764              CLR      #MMPRES          ;INDICATE NO MEM MGT PRESENT IN CASE IT ISN'T.

```

```

4364 017354 005067 162762          CLR    MMENAB          ;INDICATE MEM MGT IS NOT ENABLED.
4365 017360 004767 174160          JSR    PC,CKTRAP      ;CLEAR THF MEM MGT SRO REG AND CHECK FOR TRAP.
4366 017364 016767 162714 160412    MOV    TP4VEC,4       ;RESTORE THE NORMAL 004 TRAP VECTOR.
4367 017372 103003                    BCC    10$            ;SKIP INDICATING MEM MGT PRESENT IF IT ISN'T.
4368 017374 012767 000001 162736    MOV    #1,MMPRES      ;INDICATE THAT MEM MGT IS PRESENT.
4369 017402 005067 162664 10$:   CLR    PASCNT         ;CLR COUNTER USED IN REPORTING ROM VERSION #.
4370 017406 000167 000006          JMP    NEWPAS         ;SKIP AROUND THE BUS RESET, IT'S BEEN DONE.
4371
4372 017412                    NEWRES: BRESET       ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
      017412 104433                    TRAP    C$RESET
4373 017414 005067 162652          CLR    PASCNT         ;CLR COUNTER USED IN REPORTING ROM VERSION #.
4374 017420                    NEWPAS:
4375 017420 012767 177777 162606    MOV    #-1,UNITN     ;RESET LOGICAL DEVICE TO -1
4376
4377                    ; INCREMENT THE PASS COUNTER, CORRECT FOR ANY OVERFLOW.
4378                    ; THIS COUNTER IS USED IN THE ROM VERSION TEST.
4379                    ;-
4380 017426 005267 162640          INC    PASCNT         ;INCREMENT THE PASS COUNTER.
4381 017432 001002                    BNE    GETPRM        ;BRANCH IF WE HAVE NOT YET! OVERFLOWED.
4382 017434 005367 162632          DEC    PASCNT         ;SET PASS COUNT TO 177777 OCTAL.
4383
4384                    ; GET THE HARDWARE PARAMETERS FOR THIS UNIT.
4385 017440                    GETPRM:
4386 017440 005267 162570          INC    UNITN          ;INCREMENT LOGICAL DEVICE NUMBER
4387 017444 026767 162564 162340    CMP    UNITN,L$UNIT   ;SEE IF MAXIMUM UNIT NO. EXCEEDED
4388 017452 002362                    BGE    NEWPAS         ;BR IF YES
4389
4390 017454                    GPHARD UNITN,R1      ;GET P-TABLE POINTER INTO R1
      017454 016700 162554                    MOV    UNITN,R0
      017460 104442                    TRAP   C$GPHRD
      017462 010001                    MOV    R0,R1
4391 017464                    BCOMPLETE           ;BR IF DEVICE AVAILABLE
      017464 103401                    BCS   30$
4392 017466 000764                    BR    GETPRM         ;SKIP THIS DEVICE
4393
4394
4395                    ;***** HARDWARE PARAMETER MOVING CODE *****
4396 017470 012167 162542 30$:   MOV    (R1)+,CSRA     ;STORE DHU-11 CSR ADDRESS IN DEV.REG.ADDRESS TABLE
4397 017474 012102                    MOV    (R1)+,R2      ;GET THE RX INTERRUPT VECTOR ADDRESS.
4398 017476 010267 162522                    MOV    R2,RXVECA    ;STORE RX INT VECTOR ADDRESS.
4399 017502 062702 000004                    ADD    #4,R2         ;CALCULATE TX INTERRUPT VECTOR ADDRESS.
4400 017506 010267 162514                    MOV    R2,TXVECA    ;STORE TX INT VECTOR ADDRESS.
4401 017512 012167 162512                    MOV    (R1)+,ACTLNS ;STORE DHU-11 ACTIVE LINE BIT MAP
4402 017516 112167 162510                    MOVB  (R1)+,LOPBCK  ;STORE DHU-11 LOOPBACK MODE
4403 017522 111167 162505                    MOVB  (R1),BRLEVL   ;STORE DHU-11 INTERUPT BUS REQUEST LEVEL
4404
4405                    ;+
4406                    ; CALCULATE DEVICE REGISTER ADDRESSES,AND PUT THEM IN THE
4407                    ; DEVICE REGISTER ADDRESS TABLE.
4408 017526 016701 162504                    ;-
      MOV    CSRA,R1      ;COPY CSR ADDRESS
      INC    R1           ;INCREMENT CSR ADDRESS
4409 017532 005201                    INC    R1           ; COPY BY 2.
4410 017534 005201                    MOV    #7,R3       ;SET UP REGISTER COUNT
4411 017536 012703 000007                    MOV    #RBUFA,R2   ;GET LOCATION WHERE RBUF ADDRESS GOES IN TABLE
4412 017542 012702 002240 12$:   MOV    R1,(R2)+     ;STORE REGISTER ADDRESS IN TABLE
4413 017546 010122                    INC    R1           ;INCREMENT REGISTER ADDRESS
4414 017550 005201                    INC    R1           ; BY 2,FOR THE NEXT DEVICE REGISTER.
4415 017552 005201

```

```

4416 017554 005303          DEC      R3          ;DECREMENT REGISTER COUNT
4417 017556 001373          BNE      12$         ;LOOP IF NOT DONE
4418
4419
4420          ;+
4421          ; INITIALISE THE BMP CODE QUEUE.
4422 017560 012700 002420    ;-
4423 017564 012701 002620    MOV      #BMPQB,R0    ;GET THE START ADDRESS OF THE QUEUE.
4424 017570 010067 162622    MOV      #BMPQE,R1    ;GET THE END ADDRESS OF THE QUEUE.
4425 017574 005020          MOV      R0,BMPCQP    ;SET THE POINTER TO THE START OF THE QUEUE.
4426 017576 020001          14$:   CLR      (R0)+        ;CLEAR OUT THE CONTENTS OF THE QUEUE.
4427 017600 103775          CMP      R0,R1        ;CHECK IF END OF QUEUE HAS BEEN REACHED.
4428          BLO      14$         ;LOOP IF NOT ALL DONE.
4429
4430          ;+
4431          ; REPORT THE UNIT NUMBER IF THE SOFTWARE P-TABLE QUESTION WAS ANSWERED YES,
4432          ; AND THE MAXIMUM UNIT NUMBER IS GREATER THAN 1.
4433 017602 032767 000020 162410 ;-
4434 017610 001416          BIT      #BIT4,OPTION ;CHECK IF THE QUESTION WAS ANSWERED YES.
4435 017612 026727 162174 000001 BEQ      16$         ;SKIP REPORTING UNIT NUMBER IF IT IS DISABLED.
4436 017622 003412          CMP      L$UNIT,#1    ;CHECK MAXIMUM NUMBER OF UNITS SELECTED.
4437 017622 016746 162406          BLE     16$         ;DO NOT REPORT UNIT NUMBER IF MAX NUMBER < 1.
4438 017626 012746 004166          PRINTF #MFUNIT,UNITN ;REPORT UNIT NUMBER.
4439 017632 012746 000002          MOV      UNITN,-(SP)
4440 017636 010600          MOV      #MFUNIT,-(SP)
4441 017640 104417          MOV      #2,-(SP)
4442 017642 062706 000006          MOV      SP,R0
4443 017646          TRAP   C$PNTF
4444          ADD      #6,SP
4445
4446          16$:
4447          ENDIT: CLR      CTRLCF          ;CLR THE CTRL-C TEST ABORT FLAG.
4448          ;+
4449          ; SET THE PROCESSOR PRIORITY TO ALLOW LTC INTERRUPTS BUT NOT OTHERS.
4450          ;-
4451          SETPRI #PRI07          ;SET PROCESSOR PRIORITY TO 7.
4452 017652 012700 000340          MOV      #PRI07,R0
4453 017656 104441          TRAP   C$SPRI
4454 017660          ENDINIT
4455 017660 104411          L10016: TRAP   C$INIT
4456          TNUM == 0          ;INITIALIZE THE ASSEMBLER TEST NUMBER VARIABLE.

```

```

4449 ;*****
4450 ;
4451 ;           FVTA.ATD
4452 ;
4453 ;*****
4455
4456

```

.SBTTL AUTODROP SECTION

```

4457
4458
4459
4460 ;**
4461 ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
4462 ; THE "ADR" FLAG WAS SET. THE UNIT(S) UNDER TEST ARE CHECKED TO
4463 ; SEE IF THEY WILL RESPOND. THOSE THAT DON'T ARE IMMEDIATELY
4464 ; DROPPED FROM TESTING.
4465 ;--
4466

```

```

4467 017662          BGNAUTO
4468 017662
4475
4476 017662          ENDAUTO
      017662
      017662 104461
                                L$AUTO::
                                L10017: TRAP C$AUTO

```

4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500
4509
4510
4511
4523
4524
4525
4526

017664
017664
017664 005767 162370
017670 001401
017672 104433
017674
017674 104432
017676 000002
017700
017700
017700 104412

```
*****
:
:           FVT.CUC
:
*****
```

.SBTTL CLEANUP CODING SECTION

```
***
: THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
: AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
:--
```

```
                BGNCLN
                                L$CLEAN::
                TST    CTRLCF    ;DID WE GET HERE BY CTRL-C FROM TEST?
                BEQ    2$        ;CTRL-C FROM TEST? NO, SKIP BUS RESET.
                BRESET                ;YES, CLR ANY DMAS OR OUTSTANDING INTERRUPTS.
                                TRAP    C$RESET
2$:
                EXIT    CLN
                                TRAP    C$EXIT
                                .WORD  L10020-.
                .EVEN
                ENDCLN
                                L10020:
                                TRAP    C$CLEAN
```

4528
4529
4530
4531
4532
4533
4534
4535
4536
4537
4538
4539
4540
4541
4542
4543

```
*****  
:                               :  
:                               :  
:                               :  
:                               :  
:                               :  
*****
```

.SBTTL DROP UNIT SECTION

```
***  
: THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE  
: TO NO LONGER BE TESTED.  
:--
```

4544 017702
4545 017702

BGNDU

L#DU::

4546 017702
017702 010046
017704 012746 017726
017710 012746 000002
017714 010600
017716 104417
017720 062706 000006
4547 017724 000427

PRINTF #DROP,R0 ;REPORT UNIT THAT HAS BEEN DROPPED.

MOV R0,-(SP)
MOV #DROP,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C#PRINTF
ADD #6,SP

BR EDROP ;BRANCH AROUND THE MESSAGE.

4549 017726 045 101 040
017731 125 116 111
017734 124 045 104
017737 066 045 101
017742 040 104 122
017745 117 120 120
017750 105 104 040
017753 106 122 117
017756 115 040 106
017761 125 122 124
017764 110 105 122
017767 040 124 105
017772 123 124 111
017775 116 107 056
020000 045 116 000

DROP: .ASCIZ/##A UNIT#D6##A DROPPED FROM FURTHER TESTING.#N/

4550
4551 020004

EDROP: .EVEN

4552
4553 020004
020004 000167
020006 000000

EXIT DU

.WORD J#JMP
.WORD L10021-2-

4554
4555
4556 020010
020010
020010 104453

ENDDU

L10021:
TRAP C#DU

4558
4559
4560
4561
4562
4563
4564
4565
4566
4567
4568
4569
4570
4571
4572
4573
4574
4575
4576
4577
4578
4579
4580
4581
4582
4583

020012
020012
020012 000167
020014 000000
020016
020016
020016 104452

```

:*****
:
:           FVTA.ADD
:
:*****

```

.SBTTL ADD UNIT SECTION

```

:
: THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
: TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
: TO THE TEST CYCLE.
:--

```

BGNAU

L\$AU::

EXIT AU

```

.WORD J$JMP
.WORD L10022-2-

```

.EVEN

ENDAU

```

L10022: TRAP C$AU

```

4585
4586
4587
4588
4589
4590
4591
4592
4593
4594
4595
4596
4597
4598
4599 020020
 020020
4600
4601 020020 000001
4602 020026 012767 000001 162234
4603 020034 012767 177777 162224
4604 020034 012767 000145 164020
4605 020042 012767 005443 164014
4606 020050 012767 011252 164010
4607
4608
4609 020056 016767 157722 162220
4610 020064 012767 017000 157712
4611 020072 005005
4612
4613
4614
4615
4616
4617 020074 016700 162136
4618 020100 012701 020272
4619 020104 004767 173434
4620 020110 103402
4621 020112 052705 100001
4622 020116 042767 000017 000146 40:
4623 020124 010100
4624 020126 016701 162104
4625 020132 004767 173406
4626 020136 103403
4627 020140 052705 100002
4628 020144 000434
4629
4630
4631
4632 020146 012702 000010
4633 020152 016767 162060 000110 60:
4634 020160 012700 020270
4635 020164 012701 020272 80:
4636 020170 004767 173350
4637 020174 103402
4638 020176 052705 100001
4639 020202 010100
4640 020204 012701 020270 100:

```
.SBTTL  HARDWARE TEST          - ADRA -  
;+  
;*****  
; - REGISTER ADDRESS TEST -  
;+  
; THIS TEST VERIFIES THAT THE DEVICE REGISTERS WILL RESPOND TO THE PROPER  
; UNIBUS HANDSHAKING SIGNALS WHEN ACCESSED. IF THE DMU11 DOES NOT RESPOND  
; TO THE ACCESS ATTEMPTS (IF THE DMU11 IS AT THE WRONG ADDRESS, FOR EXAMPLE)  
; THE 004 BUS TIME-OUT TRAP IS DETECTED BY THIS ROUTINE AND AN ERROR  
; IS REPORTED. THIS TEST IS PERFORMED ON LINE 0 ONLY.  
;+  
;*****  
;--  
  
                  BGNTST  
  
                                                          T1::  
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.  
MOV  #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (1)  
MOV  #-1,CTRLCF  ;INDICATE THAT WE ARE IN A TEST.  
MOV  #101,,ERRNBR ;SET THE TEST ERROR NUMBER IN THE TABLE.  
MOV  #EM0103,ERRMSG ;SET UP THE TEST FAILURE MESSAGE IN THE TABLE.  
MOV  #ERO101,ERRBLK ;SET-UP THE ERROR ROUTINE IN THE ERROR TABLE.  
  
;+  
; SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:  
;--  
MOV  4,TP4VEC ;SAVE THE EXISTING 004 TRAP VECTOR.  
MOV  #TPARTN,4 ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.  
CLR  R5 ;CLEAR THE ERROR FLAGS.  
  
;+  
; 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.  
;--  
MOV  CSRA,R0 ;SET UP CSR AS THE CKTRAP MOVE SOURCE.  
MOV  #52,R1 ;SET UP DESTINATION LOCATION FOR CKTRAP MOVE.  
JSR  PC,CKTRAP ;MOVE AND CHECK FOR TRAP.  
BCS  4 ;IF NO TRAP, BYPASS ERROR.  
BIS  #100001,R5 ;SET FATAL READ ERROR FLAGS.  
BIC  #17,52 ;CLEAR THE I.A.R FIELD OF THE CSR DATA.  
MOV  R1,R0 ;USE OLD DESTINATION FOR SOURCE OF CKTRAP MOVE.  
MOV  CSRA,R1 ;SET UP CSR AS THE CKTRAP MOVE DESTINATION.  
JSR  PC,CKTRAP ;MOVE AND CHECK FOR TRAP.  
BCS  6 ;IF NO TRAP, BYPASS ERROR.  
BIS  #100002,R5 ;SET FATAL WRITE ERROR FLAGS.  
BR   40 ;EXIT AND REPORT FATAL ERROR.  
  
;+  
; NOW, WE TEST EACH REGISTER FOR THIS LINE.  
;--  
60: MOV  #8,,R2 ;INIT REGISTER COUNTER TO 8.  
MOV  CSRA,50 ;INITIALIZE THE REGISTER POINTER.  
80: MOV  #50,R0 ;SET UP REGISTER AS THE SOURCE FOR CKTRAP MOVE.  
MOV  #52,R1 ;SET UP LOCAL STORAGE AS THE DES FOR CKTRAP.  
JSR  PC,CKTRAP ;PERFORM THE MOVE, CHECK FOR TRAP.  
BCS  10 ;IF NO TRAP, BYPASS THE SETTING OF ERROR FLAGS.  
BIS  #100001,R5 ;SET FATAL READ ERROR FLAGS.  
100: MOV  R1,R0 ;USE OLD DEST AS SRC FOR CKTRAP MOVE.  
MOV  #50,R1 ;SET UP REGISTER AS THE DEST FOR CKTRAP MOVE.
```



```

4676 .SBTTL  HARDWARE TEST          - DMASTA -
4677 ;* *****
4678 ;* - DMA START BIT TEST -
4679 ;* THIS TEST VERIFIES THAT THE DMA_START BIT IN THE DUT'S LINE CONTROL
4680 ;* REGISTERS WILL INITIATE DMA TRANSMISSION ON THE SELECTED LINE.
4681 ;* THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, ON ALL ACTIVE LINES.
4682 ;*
4683 ;* *****
4684 020302 BGNTST
4685 020302          SETPRI  @PRI05          ;ALLOW LTC INTERRUPTS.          T2::
         020302      012700  000240          ;
         020306      104441          ;
4686         000002          TNUM == TNUM + 1          ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
4687 020310      012767  000002  161744      MOV      @TNUM,TSTNUM          ;SET UP THE TEST NUMBER.          (40)
4688 020316      012767  177777  161734      MOV      @-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
4689 020324      012767  000001  163526      MOV      @1,ERRTYP          ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
4690 020332      012767  007641  163522      MOV      @4001.,ERRNBR          ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
4691 020340      012767  005744  163516      MOV      @EM4001,ERRMSG          ;SET ERROR MESSAGE ADDRESS IN ERRRTL.
4692 020346      012767  012634  163512      MOV      @ER9101,ERRBLK          ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
4693
4694 ;*
4695 ;* RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
4696 ;* CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
4697 ;* THIS SUBROUTINE REPORTS ERROR >>>> 4001 <<<<<.
4698 020354      004767  173214          JSR      PC,CLNRST          ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
4699 020360      103145          BCC     50#          ;RESET FAILURE?, ABORT THIS TEST.
4700
4701 020362      004767  173554          JSR      PC,INDATP          ;INITIALSE THE 256 BYTE DATA PATTERN.
4702
4703 ;*
4704 ;* SET INTERNAL LOOPBACK,ENABLE RECEIVER FUNCTIONS ON ALL ACTIVE LINES.
4705 ;* SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
4706 ;* 2 STOP BITS.
4707 ;* ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
4708 020366      016705  161636          MOV      ACTLNS,R5          ;PASS THE ACTIVE LINE BIT MAP.
4709 020372      012700  000204          MOV      @204,R0          ;PASS THE LNCTRL CONTENTS.
4710 020376      004767  176246          JSR      PC,WTMLNC          ;INITIALISE THE LNCTRL REGISTERS.
4711 020402      012700  177670          MOV      @177670,R0          ;PASS THE LPR CONTENTS.
4712 020406      004767  176266          JSR      PC,WTMLPR          ;INITIALSE THE LPR REGISTERS ON ALL LINES.
4713 020412      004767  175552          JSR      PC,TXENBL          ;ENABLE TRANSMITTERS ON ALL LINES.
4714
4715 ;*
4716 ;* SET-UP OUTER LOOP TO TEST THE DMA_START BIT ON ALL ACTIVE LINES.
4717 020416      016705  161606          MOV      ACTLNS,R5          ;GET THE ACTIVE LINE BIT MAP.
4718 020422      005001          CLR      R1          ;CLEAR THE LINE NUMBER COUNTER.
4719 020424      012767  007642  163430  2# : MOV      @4002.,ERRNBR          ;SET THE ERROR NUMBER TO 4002.
4720 020432      000241          CLC          ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
4721 020434      006005          ROR     R5          ;SHIFT THE BIT MAP INTO THE CARRY BIT.
4722 020436      103112          BCC     14#          ;DO NOT TEST THE LINE IF IT IS INACTIVE.
4723 020440      004767  174316          JSR      PC,PUFIFO          ;PURGE THE FIFO.
4724 020444      103113          BCC     50#          ;GO REPORT ERROR IF FIFO WILL NOT CLEAR.
4725
4726 ;*
4727 ;* PERFORM DMA_START BIT TESTING ON EACH LINE INDIVIDUALLY.
4728 ;* TEST EACH DMA_START BIT BEFORE TX'ING DATA PATTERN, REPORT ERROR IF SET.
4729 ;* SET DMA_START BIT ON LUT, VERIFY IT IS SET, REPORT ERROR IF CLEAR.
         ;* WAIT FOR DMA TO COMPLETE.
    
```

```

4730      ; VERIFY DMA_START BIT IS CLEAR, REPORT ERROR IF SET.
4731      ; VERIFY CORRECT NUMBER OF CHARS WERE RECEIVED, REPORT ERROR IF < EXPECTED.
4732      ;-
4733 020446 005267 163410      INC      ERRNBR      ;SET ERROR NUMBER TO 4003.
4734 020452 012702 002660      MOV      @BUFBAS,R2  ;PASS THE START OF THE DATA PATTERN TO TX.
4735 020456 012703 000144      MOV      #100.,R3   ;PASS THE LENGTH OF THE DATA PATTERN.
4736 020462 004767 173304      JSR      PC,DODMA   ;TRANSMIT THE DATA PATTERN.
4737 020466 103067              BCC      12#        ;GO REPORT ERROR IF DMA_START BIT SET.
4738
4739      ;+
4740      ; TEST THE STATE OF THE DMA_START BIT ON THE LINE UNDER TEST.
4741      ; REPORT ERROR IF DMA_START BIT IS CLEAR.
4742      ;-
4742 020470 005267 163366      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4004.
4743 020474 010177 161536      MOV      R1,@CSRA   ;SELECT THE LINE CURRENTLY UNDER TEST.
4744 020500 105777 161546      TSTB    @TXAD2A    ;TEST THE STATE OF THE DMA_START BIT.
4745 020504 100060              BPL      12#        ;GO REPORT ERROR IF BIT IS CLEAR.
4746
4747      ;+
4748      ; WAIT FOR DMA TRANSMISSION TO COMPLETE.
4749      ;-
4749 020506 005267 163350      4# : INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4005.
4750 020512 010103              MOV      R1,R3      ;SAVE THE LINE NUMBER.
4751 020514 012701 170226      MOV      #170226,R1 ;TEST BIT 15, TIMEOUT OF 150 MILLI SECS.
4752 020520 016702 161512      MOV      CSRA,R2   ;PASS THE ADDRESS OF THE REGISTER TO TEST.
4753 020524 004767 176004      JSR      PC,WAIBIS ;WAIT FOR DMA TO COMPLETE.
4754 020530 103045              BCC      10#        ;GO REPORT ERROR IF TIMEOUT OCCURRED.
4755 020532 012704 000005      MOV      #5,R4     ;PASS DELAY OF 5 MILLI SECS.
4756 020536 004767 173170      JSR      PC,DELAY  ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
4757 020542 010301              MOV      R3,R1     ;RESTORE THE CURRENT LINE NUMBER.
4758
4759      ;+
4760      ; TEST THE STATE OF THE DMA_START BIT ON THE LINE UNDER TEST.
4761      ; REPORT ERROR IF DMA_START BIT IS SET.
4762      ;-
4762 020544 005267 163312      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4006.
4763 020550 010177 161462      MOV      R1,@CSRA   ;SELECT THE LINE CURRENTLY UNDER TEST.
4764 020554 105777 161472      TSTB    @TXAD2A    ;TEST THE STATE OF THE DMA_START BIT.
4765 020560 100432              BMI      12#        ;GO REPORT ERROR IF BIT IS STILL SET.
4766
4767      ;+
4768      ; VERIFY THE NUMBER OF CHARS RECEIVED = NUMBER OF CHARS EXPECTED.
4769      ; REPORT ERROR IF COUNT IS INCORRECT.
4770      ; IF MORE THAN 128 BMP CODES ARE FOUND THEN REPORT ERROR AND EXIT TEST.
4771      ;-
4771 020562 005003              CLR      R3         ;CLEAR THE READ COUNTER.
4772 020564 012704 000200      MOV      #128.,R4  ;SET UP MAX BMP CODE READ COUNT.
4773 020570 012767 007647      MOV      #4007.,ERRNBR ;SET ERROR NUMBER TO 4007.
4774 020576 017702 161436      MOV      @RBUFA,R2 ;READ THE CHARACTER FROM THE FIFO.
4775 020602 100021              BPL      12#        ;GO REPORT ERROR IF FIFO EMPTY TOO SOON.
4776 020604 012700 170301      MOV      #170301,R0 ;SET-UP BIT MASK OF A BMP CODE.
4777 020610 040200              BIC      R2,R0     ;TRY TO CLEAR THE BMP CODE MASK.
4778 020612 001007              BNE      8#        ;BRANCH IF NOT A BMP CODE.
4779 020614 005267 163242      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4008.
4780 020620 004767 174656      JSR      PC,SAVBMP ;SAVE THE BMP CODE ON THE QUEUE.
4781 020624 005304              DEC      R4         ;DECREMENT MAX BMP CODE READ COUNT.
4782 020626 001422              BEQ      50#       ;GO REPORT ERROR IF TOO MANY BMP CODES FOUND.
4783 020630 000757              BR       6#        ;DO NOT COUNT THE BMP CODE AS A VALID CHAR.
4784 020632 005203      8# : INC      R3         ;COUNT THIS CHARACTER.
4785 020634 020327 000144      CMP      R3,#100.  ;HAVE WE RECIEVED 100 CHARACTERS?.
4786 020640 002753              BLT      6#        ;LOOP UNTIL 100 (NON-BMP) CHARS ARE READ.
    
```

```

4787 020642 000410          BR      14#          ;SKIP AROUND THE ERROR REPORT.
4788
4789
4790
4791          ;+
          ; REPORT ERROR, SKIP FURTHER TESTING ON THIS LINE.
          ;-
4792 020644 010301      10#:    MOV      R3,R1          ;RESTORE THE CURRENT LINE NUMBER.
4793
4794 020646 012702 005776      12#:    MOV      @EM4002,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
4795
4796 020652          ERROR          ; "DMA_START BIT BAD ON LINE NN".
          ;          >>>> ERROR <<<<.
          ;          TRAP      C#ERROR
4797
4798
4799          ;+
          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
          ;-
4801 020654 032767 000100 161336      BIT      @BIT06.OPTION      ;EXIT WITH TEST FAILURE MESSAGE IF
4802 020662 001406          BEQ      60#          ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
4803
4804 020664 005201      14#:    INC      R1          ;INCREMENT THE LINE NUMBER COUNTER.
4805 020666 005705          TST      R5          ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
4806 020670 001255          BNE      2#          ;YES; BRANCH TO TEST THE NEXT LINE.
4807 020672 000402          BR      60#          ;NO; EXIT THIS TEST.
4808
4809 020674 004767 175040      50#:    JSR      PC,TSABRT      ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
4810 020700 005067 161354      60#:    CLR      CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
4811
4812          ENDTST
          ;
          ;          L10024:
          ;          TRAP      C#ETST

```

```

4814 .SBTTL  HARDWARE TEST      - DMABRT -
4815 ;** *****
4816 ;*          - DMA ABORT/RESTART TEST -
4817 ;* THIS TEST VERIFIES THAT EACH DMA_ABORT BIT WILL CORRECTLY HALT
4818 ;* A DMA TRANSMISSION, AND RETURN A TX_ACTION.
4819 ;* IT WILL ALSO VERIFY THAT THE ABORTED DMA TRANSMISSION CAN BE RESUMMED,
4820 ;* AND THAT A TX_ACTION IS RETURNED UPON COMPLETION.
4821 ;* THIS TEST IS PERFORMED IN INTERNAL LOOPBACK, ON ALL ACTIVE LINES.
4822 ;*
4823 ;-- *****
4824 020706 BGNST
      020706
4825 020706 SETPRI #PRI05          ;ALLOW LTC INTERRUPTS.      T3::
      020706 012700 000240
      020712 104441
4826          000003          TNUM == TNUM + 1          ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
4827 020714 012767 000003 161340 MOV #TNUM,TSTNUM          ;SET UP THE TEST NUMBER.      (41)
4828 020722 012767 177777 161330 MOV #-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
4829 020730 012767 000001 163122 MOV #1,ERRTYP          ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
4830 020736 012767 010005 163116 MOV #4101,ERRNBR          ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
4831 020744 012767 006032 163112 MOV #EM4101,ERRMSG          ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
4832 020752 012767 012634 163106 MOV #ER9101,ERRBLK          ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
4833
4834 ;*
4835 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
4836 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
4837 ; THIS SUBROUTINE REPORTS ERROR >>>> 4101 <<<<<.
4838 020760 004767 172610 JSR PC,CLNRST          ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
4839 020764 103164 BCC #0          ;RESET FAILURE?, ABORT THIS TEST.
4840
4841 020766 004767 173150 JSR PC,INDATP          ;INITIALISE 256 BYTE DATA PATTERN.
4842
4843 ;*
4844 ; SET INTERNAL LOOPBACK,ENABLE RECEIVER FUNCTIONS ON ALL ACTIVE LINES.
4845 ; SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
4846 ; 2 STOP BITS.
4847 ; ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
4848 020772 016705 161232 MOV ACTLNS,R5          ;PASS THE ACTIVE LINE BIT MAP.
4849 020776 012700 000204 MOV #204,R0          ;PASS THE LNCTRL CONTENTS.
4850 021002 004767 175642 JSR PC,WTLNC          ;INITIALISE THE LNCTRL REGISTERS.
4851 021006 012700 177670 MOV #177670,R0          ;PASS THE LPR CONTENTS.
4852 021012 004767 175662 JSR PC,WTLPR          ;INITIALSE THE LPR REGISTERS ON ALL LINES.
4853 021016 004767 175146 JSR PC,TXENBL          ;ENABLE TRANSMITTERS ON ALL LINES.
4854
4855 ;*
4856 ; PERFORM DMA_ABORT BIT TESTING ON EACH INDIVIDUAL (ACTIVE) LINE.
4857 021022 016705 161202 MOV ACTLNS,R5          ;GET THE ACTIVE LINE BIT MAP.
4858 021026 005001 CLR R1          ;CLEAR THE LINE NUMBER COUNTER.
4859 021030 012767 010006 163024 20: MOV #4102,ERRNBR          ;SET THE ERROR NUMBER TO 4102.
4860 021036 000241 CLC          ;CLEAR THE CARRY BIT PRIOR TO SHIFTING BIT MAP.
4861 021040 006005 ROR R5          ;SHIFT THE BIT MAP INTO THE CARRY BIT.
4862 021042 103127 BCC #10          ;DO NOT TEST THE LINE IF IT IS INACTIVE.
4863 021044 004767 173712 JSR PC,PUFIFO          ;PURGE THE FIFO.
4864 021050 103130 BCC #50          ;GO REPORT ERROR IF FIFO WILL NOT CLEAR.
4865
4866 ;*
4867 ; CHECK THE DMA_ABORT BIT BEFORE ENABLING DMA, REPORT ERROR IF SET.

```

```

4868 021052 005267 163004      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4103.
4869 021056 010177 161154      MOV      R1,BCSRA    ;SELECT THE LINE CURRENTLY UNDER TEST.
4870 021062 032777 000001 161156  BIT      #BIT0,DLNCTRA ;TEST THE STATE OF THE DMA_ABORT BIT.
4871 021070 001105          BNE      6#         ;GO REPORT ERROR IF BIT IS SET.
4872
4873      ;+
4874      ; ENABLE DMA TX ON SELECTED LINE, WAIT FOR DMA TO TX APPROX 1/4 OF DATA.
4875      ; ABORT THE DMA TRANSMISSION. WAIT FOR TX_ACTION TO BE RETURNED.
4876 021072 005267 162764      INC      ERRNBR      ;SET ERROR NUMBER TO 4104.
4877 021076 012702 002660      MOV      #BUFBAS,R2  ;PASS THE START OF THE DATA PATTERN TO TX.
4878 021102 012703 000400      MOV      #256.,R3    ;PASS THE LENGTH OF THE DATA PATTERN.
4879 021106 004767 172660      JSR      PC,DODMA    ;TRANSMIT THE DATA PATTERN.
4880 021112 103107      BCC      50#        ;GO REPORT ERROR IF THERE ARE TX PROBLEMS.
4881
4882      ;+
4883      ; WAIT FOR DMA TO TRANSMIT 1/4 OF THE DATA BEFORE ABORTING.
4884 021114 010177 161116      MOV      R1,BCSRA    ;SELECT THE LINE CURRENTLY UNDER TEST.
4885 021120 012704 000050      MOV      #40.,R4     ;PASS THE DELAY TIME OF 40 MILLI SECONDS.
4886 021124 004767 172602      JSR      PC,DELAY    ;WAIT FOR APPROX 1/4 OF DATA TO BE TX'D.
4887 021130 052777 000001 161110  BIS      #BIT0,DLNCTRA ;ABORT THE DMA TRANSMISSION.
4888
4889      ;+
4890      ; WAIT FOR TX_ACTION TO BE RETURNED, REPORT ERROR IF TIME-OUT OCCURS.
4891 021136 005267 162720      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4105.
4892 021142 010103      MOV      R1,R3       ;SAVE THE LINE NUMBER.
4893 021144 012701 170012      MOV      #170012,R1  ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
4894 021150 016702 161062      MOV      CSRA,R2     ;PASS THE ADDRESS OF THE REGISTER TO TEST.
4895 021154 004767 175354      JSR      PC,WAIBIS   ;WAIT FOR DMA TO COMPLETE.
4896 021160 103050      BCC      4#         ;GO REPORT ERROR IF TIMEOUT OCCURRED.
4897 021162 010301      MOV      R3,R1       ;RESTORE THE CURRENT LINE NUMBER.
4898
4899      ;+
4900      ; VERIFY DMA_START BIT CLEAR, REPORT ERROR IF SET.
4901 021164 005267 162672      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4106.
4902 021170 012702 006120      MOV      #EM4103,R2  ;SELECT MESSAGE TO BE REPORTED.
4903      ; "DMA_START BIT FOUND SET AFTER DMA ABORTED".
4904 021174 010177 161036      MOV      R1,BCSRA    ;SELECT THE LINE CURRENTLY UNDER TEST.
4905 021200 105777 161046      TSTB    #TXAD2A     ;TEST THE STATE OF THE DMA_START BIT.
4906 021204 100441      BMI      8#         ;GO REPORT ERROR IF IT IS SET.
4907
4908      ;+
4909      ; RESUME DMA TRANSMISSION BY CLEARING DMA_ABORT AND SETTING DMA_START.
4910 021206 042777 000001 161032  BIC      #BIT0,DLNCTRA ;CLEAR THE DMA_ABORT BIT.
4911 021214 052777 000200 161030  BIS      #BIT7,DTXAD2A ;SET THE DMA_START BIT.
4912
4913      ;+
4914      ; WAIT FOR DMA TRANSMISSION TO COMPLETE.
4915 021222 005267 162634      INC      ERRNBR      ;INCREMENT ERROR NUMBER TO 4107.
4916 021226 010103      MOV      R1,R3       ;SAVE THE LINE NUMBER.
4917 021230 012701 170536      MOV      #170536,R1  ;TEST BIT 15, TIMEOUT OF 350 MILLI SECS.
4918 021234 016702 160776      MOV      CSRA,R2     ;PASS THE ADDRESS OF THE REGISTER TO TEST.
4919 021240 004767 175270      JSR      PC,WAIBIS   ;WAIT FOR DMA TO COMPLETE.
4920 021244 103016      BCC      4#         ;GO REPORT ERROR IF TIMEOUT OCCURRED.
4921 021246 012704 000002      MOV      #2,R4       ;PASS TIME-OUT OF 2 MILLI SECS.
4922 021252 004767 172454      JSR      PC,DELAY    ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
4923 021256 010301      MOV      R3,R1       ;RESTORE THE CURRENT LINE NUMBER.
4924

```

```

4925 ; TEST THE STATE OF THE DMA_ABORT BIT ON THE LINE UNDER TEST.
4926 ; REPORT ERROR IF DMA_ABORT BIT IS SET.
4927 ;-
4928 021260 005267 162576          INC   ERRNBR          ;INCREMENT ERROR NUMBER TO 4108.
4929 021264 010177 160746          MOV   R1,BCSRA        ;SELECT THE LINE CURRENTLY UNDER TEST.
4930 021270 032777 000001 160750    BIT   #BIT0,&LNCTRA   ;TEST THE STATE OF THE DMA_ABORT BIT.
4931 021276 001002                    BNE   6#              ;GO REPORT ERROR IF BIT IS SET.
4932 021300 000410                    BR    10#             ;BRANCH TO CHECK FOR ANY MORE LINES TO TEST.
4933 ;+
4934 ; REPORT ERROR, SKIP FURTHER TESTING ON THIS LINE.
4935 ;-
4936 021302 010301          4#:    MOV   R3,R1          ;RESTORE THE CURRENT LINE NUMBER.
4937
4938 021304 012702 006064          6#:    MOV   #EM4102,R2      ;PASS THE ERROR MESSAGE TO BE REPORTED.
4939
4940 021310          8#:    ERROR          ; "DMA_ABORT BIT BAD ON LINE NN".
4941 021310 104460                    ; >>>> ERROR <<<<<.
4942
4943
4944 ;+
4945 021312 032767 000100 160700    BIT   #BIT06,OPTION  ;EXIT WITH TEST FAILURE MESSAGE IF
4946 021320 001406                    BEQ   60#              ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
4947
4948 ;-
4949 ; VERIFY ALL ACTIVE LINES HAVE BEEN TESTED.
4950 ;-
4951 021322 005201          10#:   INC   R1          ;INCREMENT THE LINE NUMBER COUNTER.
4952 021324 005705                    TST   R5              ;ARE THERE ANY MORE ACTIVE LINES TO TEST?.
4953 021326 001240                    BNE   2#              ;YES; BRANCH TO TEST THE NEXT LINE.
4954 021330 000402                    BR    60#             ;NO; EXIT THIS TEST.
4955
4956 021332 004767 174402          50#:   JSR   PC,TSABRT    ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
4957 021336 005067 160716          60#:   CLR   CTRLCF      ;INDICATE THAT WE ARE NOT WITHIN A TEST.
4958
4959 021342          ENDTST
4959 021342          L10025:
4959 021342 104401                    TRAP  C#ETST

```

```

4961 .SBTTL HARDWARE TEST - OAUTOI -
4962 ;*****
4963 ;* - OAUTO BIT INACTIVE TEST -
4964 ;*
4965 ;* THIS TEST VERIFIES THAT THE DUT'S OAUTO FUNCTION BEHAVES CORRECTLY
4966 ;* WHEN INACTIVE, IE OAUTO BIT CLEAR.
4967 ;* THIS TEST WILL ONLY EXECUTE IF STAGGERED LOOPBACK MODE IS SELECTED.
4968 ;* THE SPECIAL STAGGERED LOOPBACK CONNECTOR MUST BE FITTED.
4969 ;*
4970 ;*****
4971
4972 021344 BGNTST
      021344
4973 021344 126727 160662 000002 CMPB LOPBCK,#2 ;CHECK MODE SELECTED. T4::
4974 021352 001402 BEQ .+6 ;DO NOT EXIT IF STAGGERD LOPBCK MODE SELECTED.
4975 021354 000167 000556 JMP 60# ;EXIT THIS TEST.
4976 021360 SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
      021360 012700 000240
      021364 104441
      021364 000004 MOV #PRI05,R0
      021366 012767 000004 160666 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
      021374 012767 177777 160656 MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (49)
4980 021402 012767 000001 162450 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
4981 021410 012767 011445 162444 MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
4982 021416 012767 006204 162440 MOV #4901,ERRNBR ;SET ERROR NUMBER TO 4901.
4983 021424 012767 012634 162434 MOV #EM4901,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
      021424 012767 012634 162434 MOV #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
4984
4985 ;*
4986 ;* RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
4987 ;* CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
4988 ;* THIS SUBROUTINE REPORTS ERROR >>>> 4901 <<<<<.
4989 021432 004767 172136 JSR PC,CLNRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
4990 021436 103402 BCS .+6 ;DO NOT EXIT IF RESET WAS SUCCESSFUL.
4991 021440 000167 000472 JMP 60# ;EXIT THIS TEST.
4992
4993 ;*
4994 ;* SET-UP THE ASSOCIATED TX/RX LINE NUMBER TABLES.
4995 021444 004767 171470 JSR PC,ASLNTL ;INITIALISE THE ASSOCIATED TX/RX TABLES.
4996
4997 ;*
4998 ;* SET EXTERNAL LOOPBACK, DISABLE OAUTO AND ENABLE RECEIVER ON ALL ACTIVE LINES.
4999 ;* SET LPR ON ALL LINES TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY,
5000 ;* 2 STOP BITS.
5001 ;* ENABLE TRANSMITTERS ON ALL LINES.
5002 021450 016705 160554 MOV ACTLNS,R5 ;PASS THE ACTIVE LINE BIT MAP.
5003 021454 012700 000004 MOV #4,R0 ;PASS THE LNCTRL CONTENTS.
5004 021460 004767 175164 JSR PC,WTWLNLC ;INITIALISE THE LNCTRL REGISTERS.
5005 021464 012705 177777 MOV #MAPLNS,R5 ;PASS BIT MAP OF ALL LINES.
5006 021470 012700 177670 MOV #177670,R0 ;PASS THE LPR CONTENTS.
5007 021474 004767 175200 JSR PC,WTWLPR ;INITIALISE THE LPR REGISTERS ON ALL LINES.
5008 021500 004767 174464 JSR PC,TXENBL ;ENABLE TRANSMITTERS ON ALL LINES.
5009
5010 ;*
5011 ;* SET UP OUTER LOOP FOR TESTING ACTIVE LINES IN BOTH LINE GROUPS.
5012 021504 012703 100000 MOV #100000,R3 ;SET-UP LOOP CONTROL FLAG.
5013 021510 016705 160514 MOV ACTLNS,R5 ;GET THE ACTIVE LINE BIT MAP.
5014 021514 046705 160550 BIC LGRP2M,R5 ;REMOVE LINES IN GROUP 2.
    
```

```

5015 021520 010567 000404      2:      MOV      R5,45:      ;SAVE THE CURRENT LINE GROUP.
5016 021524 005067 000376      CLR      40:      ;CLEAR THE LINE NUMBER COUNTER.
5017 021530 016701 000372      4:      MOV      40:,R1    ;COPY THE LINE NUMBER.
5018 021534 000241      CLC      ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
5019 021536 006005      ROR      R5      ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
5020 021540 103064      BCC      8:      ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
5021
5022      ;*
5023      ; TEST THE STATE OF THE OAUTO BIT ON THE LINE UNDER TEST.
5024      ; REPORT ERROR IF IT IS FOUND SET, AND SKIP FURTHER TESTING OF THAT LINE.
5025 021542 012767 011446 162312      ;*
5026 021550 010177 160462      ;*      MOV      #4902.,ERRNBR ;SET THE ERROR NUMBER TO 4902.
5027 021554 032777 000020 160464      ;*      MOV      R1,BCSRA    ;SELECT THE LINE TO BE TESTED.
5028 021562 001410      ;*      BIT      #BIT4,SLNCTRA ;TEST THE STATE OF THE OAUTO BIT.
5029 021564 012702 006245      ;*      BEQ      6:      ;SKIP ERROR REPORT IF OAUTO BIT IS CLEAR.
5030      ;*      MOV      #EM4902,R2 ;PASS THE ERROR MESSAGE.
5031 021570      ;*      ERROR      ; "OAUTO BIT BAD ON LINE NN"
5032 021570 104460      ;*      ;*      >>>> ERROR #4902 <<<<<.
5033      ;*      ;*      TRAP      C:ERROR
5034
5035      ;*
5036 021572 032767 000100 160420      ;*      ;*
5037 021600 001556      ;*      ;*
5038      ;*
5039      ;*
5040 021602 000443      ;*      ;*
5041      ;*
5042      ;*
5043      ;*
5044 021604 116177 004020 160424 6:      ;*      ;*
5045 021612 112777 000023 160424      ;*      ;*
5046      ;*
5047      ;*
5048      ;*
5049 021620 005267 162236      ;*      ;*
5050 021624 012701 170012      ;*      ;*
5051 021630 016702 160402      ;*      ;*
5052 021634 004767 174674      ;*      ;*
5053 021640 103134      ;*      ;*
5054 021642 012704 000005      ;*      ;*
5055 021646 004767 172060      ;*      ;*
5056      ;*
5057      ;*
5058      ;*
5059      ;*
5060 021652 005267 162204      ;*      ;*
5061 021656 016701 000244      ;*      ;*
5062 021662 010177 160350      ;*      ;*
5063 021666 005777 160360      ;*      ;*
5064 021672 100407      ;*      ;*
5065 021674 012702 006245      ;*      ;*
5066      ;*
5067 021700      ;*      ;*
5068 021700 104460      ;*      ;*
5069      ;*

```

```

5070 ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5071 ;
5072 021702 032767 000100 160310 ; BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5073 021710 001512 ; BEQ 60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5074 ; ;DURING THE SOFTWARE QUESTIONS.
5075 ;
5076 021712 005267 000210 8# : INC 40# ;INCREMENT THE LINE NUMBER,
5077 021716 005705 ; TST R5 ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
5078 021720 001303 ; BNE 4# ;
5079 ;
5080 ;* ; DISABLE TRANSMITTERS ON THE SELECTED LINES IN THE CURRENT LINE GROUP.
5081 ;
5082 021722 016705 000202 ; MOV 45#,R5 ;RESTORE THE CURRENT LINE ACTIVE LINE GROUP.
5083 021726 004767 174142 ; JSR PC, TXDSBL ;DISABLE TRANSMITTERS ON THE SELECTED LINES.
5084 021732 016705 000172 ; MOV 45#,R5 ;GET THE CURRENT ACTIVE LINE GROUP AGAIN.
5085 021736 005067 000164 ; CLR 40# ;CLEAR THE LINE COUNTER.
5086 021742 012767 011451 162112 10# : MOV #4905.,ERRNBR ;SET ERROR NUMBER TO 4905.
5087 021750 016701 000152 ; MOV 40#,R1 ;COPY THE LINE NUMBER.
5088 021754 000241 ; CLC ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
5089 021756 006005 ; ROR R5 ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
5090 021760 103041 ; BCC 12# ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
5091 ;
5092 ;* ; TRANSMIT THE XON (ASCII DC1) ON THE ASSOCIATED LINE.
5093 ;
5094 021762 116177 004020 160246 ; MOVB TXRLNB(R1),@CSRA ;SELECT THE ASSOCIATED TX LINE.
5095 021770 112777 000021 160246 ; MOVB #21,@FDATA ;TRANSMIT THE XON CHARACTER TO THE LUT.
5096 ;
5097 ;* ; WAIT FOR TRANSMISSION TO COMPLETE.
5098 ;
5099 021776 012701 170012 ; MOV #170012,R1 ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
5100 022002 016702 160230 ; MOV CSRA,R2 ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5101 022006 004767 174522 ; JSR PC,WAIBIS ;WAIT FOR TRANSMISSION TO COMPLETE.
5102 022012 103047 ; BCC 50# ;ABORT TEST IF TIMEOUT OCCURRED.
5103 022014 012704 000005 ; MOV #5,R4 ;PASS TIME-OUT OF 5 MILLI SECS.
5104 022020 004767 171706 ; JSR PC,DELAY ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
5105 ;
5106 ;* ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
5107 ; REPORT ERROR IF TX_ENABLE BIT IS SET.
5108 ;
5109 022024 005267 162032 ; INC ERRNBR ;INCREMENT ERROR NUMBER TO 4906.
5110 022030 016701 000072 ; MOV 40#,R1 ;GET THE NUMBER OF THE LINE UNDER TEST.
5111 022034 010177 160176 ; MOV R1,@CSRA ;SELECT THE LINE CURRENTLY UNDER TEST.
5112 022040 005777 160206 ; TST @TXAD2A ;TEST THE STATE OF THE TX_ENABLE BIT.
5113 022044 100007 ; BPL 12# ;SKIP ERROR REPORT IF BIT IS CLEAR.
5114 022046 012702 006245 ; MOV #EM4902,R2 ;PASS THE MESSAGE TO BE REPORTED.
5115 ; ; "OAUTO BIT BAD ON LINE NN".
5116 022052 ; ERROR ; >>>> ERROR #4906 <<<<<.
5117 ; ; TRAP C:ERROR
5118 ;
5119 ;* ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5120 ;
5121 022054 032767 000100 160136 ; BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5122 022062 001425 ; BEQ 60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5123 ; ;DURING THE SOFTWARE QUESTIONS.
5124 ;
5125 022064 005267 000036 12# : INC 40# ;INCREMENT THE LINE NUMBER,

```

```

5126 022070 005705          TST   R5          ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
5127 022072 001323          BNE   10$          ;
5128                          ;*
5129                          ; CHECK LOOP CONTROL FLAG TO DETERMINE IF BOTH SETS OF LINES HAVE BEEN TESTED
5130                          ; IF THIS IS THE FIST TIME AROUND, RE-ENABLE TX ON ALL LINES, GENERATE ACTIVE
5131                          ; BIT MAP FOR SECOND LINE GROUP.
5132                          ;-
5133 022074 005703          TST   R3          ;HAVE BOTH LINE GROUPS BEEN TESTED?.
5134 022076 001417          BEQ   60$          ;YES, THEN EXIT THIS TEST.
5135 022100 005003          CLR   R3          ;NO, CLEAR THE LTOP CONTROL FLAG.
5136 022102 012705 177777  MOV   #MAPLNS,R5 ;PASS THE BIT MAP OF ALL AVAILABLE LINE.
5137 022106 004767 174056  JSR   PC, TXENBL ;RE-ENABLE TRANSMISSION ON ALL LINES.
5138 022112 016705 160112  MOV   ACTLNS,R5  ;GET THE ACTIVE LINE BIT MAP.
5139 022116 046705 160144  BIC   LGRP1M,R5  ;REMOVE ALL ACTIVE LINES IN GROUP 1.
5140 022122 000167 177372  JMP   2$          ;ONCE MORE AROUND AND WE ARE DONE.
5141
5142 022126 000000          40$: .WORD 0      ;STORAGE FOR CURRENT LINE NUMBER.
5143 022130 000000          45$: .WORD 0      ;STORAGE FOR CURRENT ACTIVE LINE BIT MAP.
5144 022132 004767 173602  50$: JSR   PC, TSABRT ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
5145 022136 005067 160116  60$: CLR   CTRLCF   ;INDICATE THAT WE ARE NOT WITHIN A TEST.
5146
5147 022142          ENDTST
      022142
      022142 104401

```

L10026: TRAP C#ETST

5149
 5150
 5151
 5152
 5153
 5154
 5155
 5156
 5157
 5158
 5159
 5160
 5161 022144
 022144
 5162 022144 126727 160062 000002
 5163 022152 001402
 5164 022154 000167 000556
 5165 022160
 022160 012700 000240
 022164 104441
 5166 000005
 5167 022166 012767 000005 160066
 5168 022174 012767 177777 160056
 5169 022202 012767 000001 161650
 5170 022210 012767 011611 161644
 5171 022216 012767 006277 161640
 5172 022224 012767 012634 161634
 5173
 5174
 5175
 5176
 5177
 5178 022232 004767 171336
 5179 022236 103402
 5180 022240 000167 000472
 5181
 5182
 5183
 5184 022244 004767 170670
 5185
 5186
 5187
 5188
 5189
 5190
 5191 022250 016705 157754
 5192 022254 012700 000024
 5193 022260 004767 174364
 5194 022264 012705 177777
 5195 022270 012700 177670
 5196 022274 004767 174400
 5197 022300 004767 173664
 5198
 5199
 5200
 5201 022304 012703 100000
 5202 022310 016705 157714

```

.SBTTL  HARDWARE TEST          - OAUTOA -
;*****
;                                - OAUTO BIT ACTIVE TEST -
;
; THIS TEST VERIFIES THAT THE DUT'S OAUTO FUNCTION BEHAVES CORRECTLY
; WHEN ACTIVE, IE OAUTO BIT ASSERTED HIGH.
; THIS TEST WILL ONLY EXECUTE IF THE STAGGERED LOOPBACK MODE IS SELECTED.
; THE SPECIAL STAGGERED LOOPBACK CONNECTOR MUST BE FITTED.
;*****
          BGNTST
          T5::
          CMPB  LOPBCK,#2      ;CHECK MODE SELECTED.
          BEQ   .+6            ;DO NOT EXIT IF STAGGERD LOPBCK MODE SELECTED.
          JMP   60$           ;EXIT THIS TEST.
          SETPRI @PRI05       ;ALLOW LTC INTERRUPTS.
                                MOV   #PRI05,R0
                                TRAP  C$SPRI
          TNUM == TNUM + 1    ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
          MOV   #TNUM,TSTNUM  ;SET UP THE TEST NUMBER. (50)
          MOV   #-1,CTRLCF    ;INDICATE THAT WE ARE IN A TEST.
          MOV   #1,ERRTYP     ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
          MOV   #5001,ERRNBR  ;SET ERROR NUMBER TO 5001.
          MOV   #EM5001,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
          MOV   #ER9101,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
;
; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
; THIS SUBROUTINE REPORTS ERROR >>>> 5001 <<<<.
;
          JSR   PC,CLNRST     ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
          BCS   .+6          ;DO NOT EXIT IF RESET WAS SUCCESSFUL.
          JMP   60$         ;EXIT THIS TEST.
;
; SET-UP THE ASSOCIATED TX/RX LINE NUMBER TABLES.
;
          JSR   PC,ASLNTL     ;INITIALISE THE ASSOCIATED TX/RX TABLES.
;
; SET EXTERNAL LOOPBACK,ENABLE OAUTO AND 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 LINES.
;
          MOV   ACTLNS,R5     ;PASS THE ACTIVE LINE BIT MAP.
          MOV   #24,R0       ;PASS THE LNCTRL CONTENTS.
          JSR   PC,WTWLNLC   ;INITIALISE THE LNCTRL REGISTERS.
          MOV   #MAPLNS,R5   ;PASS BIT MAP OF ALL LINES.
          MOV   #177670,R0   ;PASS THE LPR CONTENTS.
          JSR   PC,WTWLPRL   ;INITIALISE THE LPR REGISTERS ON ALL LINES.
          JSR   PC,TXENBL    ;ENABLE TRANSMITTERS ON ALL LINES.
;
; SET UP OUTER LOOP FOR TESTING ACTIVE LINES IN BOTH LINE GROUPS.
;
          MOV   #100000,R3   ;SET-UP LOOP CONTROL FLAG.
          MOV   ACTLNS,R5   ;GET THE ACTIVE LINE BIT MAP.
    
```

```

5203 022314 046705 157750          BIC    LGRP2M,R5      ;REMOVE LINES IN GROUP 2.
5204 022320 010567 000404      2#:   MOV    R5,45#    ;SAVE THE CURRENT LINE GROUP.
5205 022324 005067 000376          CLR    40#          ;CLEAR THE LINE NUMBER COUNTER.
5206 022330 016701 000372      4#:   MOV    40#,R1    ;COPY THE LINE NUMBER.
5207 022334 000241              CLC          ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
5208 022336 006005              ROR    R5          ;SHIFT ACTIVE LINE BIT MAP INTO CARR. BIT.
5209 022340 103064              BCC    8#          ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
5210
5211          ;*
5212          ; TEST THE STATE OF THE OAUTO BIT ON THE LINE UNDER TEST.
5213          ; REPORT ERROR IF IT IS FOUND CLEAR, AND SKIP FURTHER TESTING OF THAT LINE.
5214 022342 012767 011612 161512      MOV    #5002,ERRNBR ;SET THE ERROR NUMBER TO 5002.
5215 022350 010177 157662              MOV    R1,BCSRA    ;SELECT THE LINE TO BE TESTED.
5216 022354 032777 000020 157664      BIT    #BIT4,BLNCTRA ;TEST THE STATE OF THE OAUTO BIT.
5217 022362 001010              BNE    6#          ;SKIP ERROR REPORT IF OAUTO BIT IS SET.
5218 022364 012702 006245              MOV    #EM4902,R2  ;PASS THE ERROR MESSAGE.
5219          ; "OAUTO BIT BAD ON LINE NN"
5220 022370              ERROR          ; >>>>> ERROR #5002 <<<<<.
5221 022370 104460              TRAP    C#ERROR
5222
5223          ;*
5224          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5225 022372 032767 000100 157620      BIT    #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5226 022400 001556              BEQ    60#         ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5227          ; DURING THE SOFTWARE QUESTIONS.
5228
5229 022402 000443              BR     8#          ;SKIP FURTHER TESTING OF THIS LINE.
5230
5231          ;*
5232          ; TRANSMIT THE XOFF (ASCII DC3) ON THE ASSOCIATED LINE.
5233 022404 116177 004020 157624 6#:   MOVB   TXRLNB(R1),BCSRA ;SELECT THE ASSOCIATED TX LINE.
5234 022412 112777 000023 157624      MOVB   #23,#FDATA  ;TRANSMIT THE XOFF CHARACTER TO THE LUT.
5235
5236          ;*
5237          ; WAIT FOR TRANSMISSION TO COMPLETE.
5238 022420 005267 161436              INC    ERRNBR      ;INCREMENT ERROR NUMBER TO 5003.
5239 022424 012701 170012              MOV    #170012,R1 ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
5240 022430 016702 157602              MOV    CSRA,R2    ;PASS THE ADDRESS OF THE REGISTER TO TEST.
5241 022434 004767 174074              JSR    PC,WAIBIS  ;WAIT FOR TRANSMISSION TO COMPLETE.
5242 022440 103134              BCC    50#         ;ABORT TEST IF TIMEOUT OCCURRED.
5243 022442 012704 000005              MOV    #5,R4      ;PASS TIME-OUT OF 5 MILLI SECS.
5244 022446 004767 171260              JSR    PC,DELAY   ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
5245
5246          ;*
5247          ; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
5248          ; REPORT ERROR IF TX_ENABLE BIT IS SET.
5249 022452 005267 161404              INC    ERRNBR      ;INCREMENT ERROR NUMBER TO 5004.
5250 022456 016701 000244              MOV    40#,R1     ;GET THE NUMBER OF THE LINE TEST.
5251 022462 010177 157550              MOV    R1,BCSRA   ;SELECT THE LINE CURRENTLY UNDER TEST.
5252 022466 005777 157560              TST    BTXAD2A    ;TEST THE STATE OF THE TX_ENABLE BIT.
5253 022472 100007              BPL    8#          ;SKIP ERROR REPORT IF BIT IS CLEAR.
5254 022474 012702 006245              MOV    #EM4902,R2 ;PASS THE MESSAGE TO BE REPORTED.
5255          ; "OAUTO BIT BAD ON LINE NN".
5256 022500              ERROR          ; >>>>> ERROR #5004 <<<<<.
5257 022500 104460              TRAP    C#ERROR

```

```

5258
5259
5260
5261 022502 032767 000100 157510
5262 022510 001512
5263
5264
5265 022512 005267 000210
5266 022516 005705
5267 022520 001303
5268
5269
5270
5271 022522 016705 000202
5272 022526 004767 173342
5273 022532 016705 000172
5274 022536 005067 000164
5275 022542 012767 011615 161312 10:
5276 022550 016701 000152
5277 022554 000241
5278 022556 006005
5279 022560 103041
5280
5281
5282
5283 022562 116177 004020 157446
5284 022570 112777 000021 157446
5285
5286
5287
5288 022576 012701 170012
5289 022602 016702 157430
5290 022606 004767 173722
5291 022612 103047
5292 022614 012704 000005
5293 022620 004767 171106
5294
5295
5296
5297
5298 022624 005267 161232
5299 022630 016701 000072
5300 022634 010177 157376
5301 022640 005777 157406
5302 022644 100407
5303 022646 012702 006245
5304
5305 022652
022652 104460
5306
5307
5308
5309
5310 022654 032767 000100 157336
5311 022662 001425
5312
5313

; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;
; BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
; BEQ 60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
; DURING THE SOFTWARE QUESTIONS.
;
; INC 40# ;INCREMENT THE LINE NUMBER,
; TST R5 ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
; BNE 4# ;
;
; DISABLE TRANSMITTERS ON THE SELECTED LINES IN THE CURRENT LINE GROUP.
;
; MOV 45#,R5 ;RESTORE THE CURRENT LINE ACTIVE LINE GROUP.
; JSR PC, TXDSBL ;DISABLE TRANSMITTERS ON THE SELECTED LINES.
; MOV 45#,R5 ;GET THE CURRENT LINE ACTIVE LINE GROUP AGAIN.
; CLR 40# ;CLEAR THE LINE COUNTER.
; MOV #5005.,ERRNBR ;SET ERROR NUMBER TO 5005.
; MOV 40#,R1 ;COPY THE LINE NUMBER.
; CLC ;CLEAR CARRY BIT PRIOR TO SHIFTING BIT MAP.
; ROR R5 ;SHIFT ACTIVE LINE BIT MAP INTO CARRY BIT.
; BCC 12# ;SKIP TESTING THIS LINE IF IT IS INACTIVE.
;
; TRANSMIT THE XON (ASCII DC1) ON THE ASSOCIATED LINE.
;
; MOVB TXRLNB(R1),BCSRA ;SELECT THE ASSOCIATED TX LINE.
; MOVB #21,BFDATA ;TRANSMIT THE XON CHARACTER TO THE LUT.
;
; WAIT FOR TRANSMISSION TO COMPLETE.
;
; MOV #170012,R1 ;TEST BIT 15, TIMEOUT OF 10 MILLI SECS.
; MOV CSRA,R2 ;PASS THE ADDRESS OF THE REGISTER TO TEST.
; JSR PC,WAIBIS ;WAIT FOR DMA TO COMPLETE.
; BCC 50# ;ABORT TEST IF TIMEOUT OCCURRED.
; MOV #5,R4 ;PASS TIME-OUT OF 5 MILLI SECS.
; JSR PC,DELAY ;WAIT FOR CHAR TO BE RECEIVED AND PROCESSED.
;
; TEST THE STATE OF THE TX_ENABLE BIT ON THE LINE UNDER TEST.
; REPORT ERROR IF TX_ENABLE BIT IS CLEAR.
;
; INC ERRNBR ;INCREMENT ERROR NUMBER TO 5006.
; MOV 40#,R1 ;GET THE NUMBER OF THE LINE UNDER TEST.
; MOV R1,BCSRA ;SELECT THE LINE CURRENTLY UNDER TEST.
; TST #TXAD2A ;TEST THE STATE OF THE TX_ENABLE BIT.
; BMI 12# ;SKIP ERROR REPORT IF BIT IS SET.
; MOV #EM4902,R2 ;PASS THE MESSAGE TO BE REPORTED.
; ERROR ; "OAUTO BIT BAD ON LINE NN".
; >>>> ERROR #5006 <<<<<.
; TRAP C#ERROR
;
; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;
; BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
; BEQ 60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
; DURING THE SOFTWARE QUESTIONS.

```

```

5314 022664 005267 000036      12$:   INC   40$      ;INCREMENT THE LINE NUMBER,
5315 022670 005705              TST   R5          ;CHECK IF THERE ARE ANY MORE LINES TO TEST.
5316 022672 001323              BNE   10$          ;
5317                               ;*
5318                               ; CHECK LOOP CONTROL FLAG TO DETERMINE IF BOTH SETS OF LINES HAVE BEEN TESTED
5319                               ; IF THIS IS THE FIST TIME AROUND, RE-ENABLE TX ON ALL LINES, GENERATE ACTIVE
5320                               ; BIT MAP FOR SECOND LINE GROUP.
5321                               ;-
5322 022674 005703              TST   R3          ;HAVE BOTH LINE GROUPS BEEN TESTED?.
5323 022676 001417              BEQ   60$          ;YES, THEN EXIT THIS TEST.
5324 022700 005003              CLR   R3          ;NO, CLEAR THE LOOP CONTROL FLAG,
5325 022702 012705 177777      MOV   @MAPLNS,R5 ;PASS THE BIT MAP OF ALL AVAILABLE LINE.
5326 022706 004767 173256      JSR   PC, TXENBL ;RE-ENABLE TRANSMISSION ON ALL LINES.
5327 022712 016705 157312      MOV   ACTLNS,R5 ;GET THE ACTIVE LINE BIT MAP.
5328 022716 046705 157344      BIC   LGRP1M,R5 ;REMOVE ALL ACTIVE LINES IN GROUP 1.
5329 022722 000167 177372      JMP   2$          ;ONCE MORE AROUND AND WE ARE DONE.
5330                               ;
5331 022726 000000      40$:   .WORD 0      ;STORAGE FOR CURRENT LINE NUMBER.
5332 022730 000000      45$:   .WORD 0      ;STORAGE FOR CURRENT ACTIVE LINE BIT MAP.
5333 022732 004767 173002      50$:   JSR   PC, TSABRT ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
5334 022736 005067 157316      60$:   CLR   CTRLCF ;INDICATE THAT WE ARE NOT WITHIN A TEST.
5335                               ;
5336 022742              ENDTST
022742
022742 104401

```

L10027: TRAP C#ETST

```

5338
5339
5340
5341
5342
5343
5344
5345
5346
5347
5348
5349
5350
5351
5352
5353
5354
5355
5356 022744
      022744
5357 022744
      022744 012700 000240
      022750 104441
5358      000006
5359 022752 012767 000006 157302
5360 022760 012767 177777 157272
5361 022766 012767 000001 161064
5362 022774 012767 011755 161060
5363 023002 012767 006336 161054
5364 023010 012767 012634 161050
5365
5366
5367
5368
5369
5370 023016 004767 170552
5371 023022 103156
5372
5373
5374
5375
5376
5377
5378 023024 004767 171142
5379
5380
5381
5382
5383 023030 016705 157174
5384 023034 012700 000204
5385 023040 004767 173604
5386 023044 012700 177670
5387 023050 004767 173624
5388 023054 012704 000012
5389 023060 004767 170646
5390
5391

```

```

.SBTTL HARDWARE TEST - IAUTOI -
;*****
;
; 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.
;*****
BGNTST
T6::
SETPRI #PRIOS ;ALLOW LTC INTERRUPTS.
MOV #PRIOS,R0
TRAP C$PRI
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (51)
MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #5101,ERRNBR ;SET ERROR NUMBER TO 5101.
MOV #EM5101,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
MOV #ER9101,ERRBK ;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 <<<<.
;
JSR PC,CLRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
BCC 600 ;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.
;
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.
;
MOV ACTLNS,R5 ;PASS THE ACTIVE LINE BIT MAP.
MOV #204,R0 ;PASS INT'L LOOPBACK, ENABLE RX, DISABLE IAUTO.
JSR PC,WTMLNC ;INITIALISE THE LINE CONTROL REGISTER.
MOV #177670,R0 ;PASS THE LPR CONTENTS.
JSR PC,WTMLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
MOV #10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
JSR PC,DELAY ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
;

```

```

5392 ; SET UP LOOP FOR ALL ACTIVE LINES.
5393 ; TEST THE STATE OF THE IAUTO BIT PRIOR TO TRANSMITTING THE DATA PATTERN.
5394 ; IF THE BIT IS SET, THEN REPORT THE ERROR AND SKIP TRANSMITTING
5395 ; THE DATA PATTERN ON THE SELECTED LINE.
5396 ; TRANSMIT A 256 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
5397 ; EMPTY THE FIFO, AND VERIFY NO XOFF OR XON CHARS WERE FOUND.
5398 ;-
5399 023064 005001          CLR     R1           ;CLEAR THE LINE NUMBER COUNTER.
5400 023066 005067 000264 CLR     55#         ;CLEAR STORAGE FOR LINE NUMBER.
5401 023072 012767 011756 160762 2# : MOV     #5102.,ERRNBR ;SET THE ERROR NUMBER TO 5102.
5402 023100 004767 171656 JSR     PC,PUFIFO    ;PURGE THE FIFO.
5403 023104 103121          BCC     50#         ;GO REPORT ERROR IF FIFO DID NOT PURGE.
5404 023106 000241          CLC                     ;CLEAR CARRY PRIOR TO ROTATING BIT MAP.
5405 023110 006005          ROR     R5           ;ROTATE THE BIT MAP INTO THE CARRY BIT.
5406 023112 103107          BCC     12#         ;BRANCH IF LINE IS INACTIVE.
5407
5408 ;+
5409 ; TEST THE IAUTO BIT ON THE SELECTED ACTIVE LINE.
5410 ; REPORT ERROR IF IT IS SET.
5411 ; DC NOT TRANSMIT THE DATA PATTERN ON THE SELECTED LINE.
5412 023114 005267 160742          INC     ERRNDR       ;SET ERROR NUMBER TO 5103.
5413 023120 010177 157112          MOV     R1,CSRA     ;SELECT LINE TO TEST.
5414 023124 032777 000002 157114 BIT     #BIT1,SLNCTRA ;TEST THE STATE OF THE IAUTO BIT ON THIS LINE.
5415 023132 001410          BEQ     4#           ;SKIP ERROR IF IAUTO BIT CLEAR.
5416 023134 012702 006373          MOV     #EM5102,R2 ;PASS THE CORRECT ERROR MESSAGE.
5417 023140          ERROR          ; >>>> ERROR <<<<.
5418          104460          TRAP     C#ERROR
5419
5420 ;+
5421 ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5422 023142 032767 000100 157050 BIT     #BIT06.OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5423 023150 001503          BEQ     60#         ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5424
5425 ;DURING THE SOFTWARE QUESTIONS.
5426 023152 000467          BR      12#         ;SKIP TRANSMITTING DATA PATTERN.
5427
5428 ;+
5429 ; TRANSMIT DATA PATTERN OF 256 CHARS.
5430 ;-
5431 023154 005267 160702 2# : INC     ERRNBR       ;SET ERROR NUMBER TO 5104.
5432 023160 012702 002660          MOV     #BUFBA5,R2 ;PASS THE START OF THE DATA PATTERN TO TX.
5433 023164 012703 000400          MOV     #256.,R3   ;PASS THE LENGTH OF THE DATA PATTERN.
5434 023170 004767 170576          JSR     PC,DODMA    ;TRANSMIT THE DATA PATTERN.
5435 023174 103065          BCC     50#         ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
5436
5437 ;+
5438 ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER PLUS XOFF
5439 ; TO ARRIVE IN THE FIFO.
5440 ;-
5441 023176 005267 160660          INC     ERRNBR       ;SET ERROR NUMBER TO 5105.
5442 023202 012701 170536          MOV     #170536,R1 ;PASS TIME-OUT VALUE OF 350 MILLI SECS.
5443 023206 016702 157024          MOV     CSRA,R2    ;PASS THE ADDRESS OF THE CSR.
5444 023212 004767 173316          JSR     PC,WAIBIS   ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
5445 023216 103054          BCC     50#         ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
5446 023220 012704 000012          MOV     #10.,R4    ;PASS DELAY OF 10 MILLI SECS.
5447 023224 004767 170502          JSR     PC,DELAY    ;WAIT FOR LAST CI AR TO ARRIVE IN THE FIFO.

```



```

5502 .SBTTL  HARDWARE TEST          - IAUTOA -
5503 ;*****
5504 ;                                     - IAUTO BIT ACTIVE TEST -
5505 ;
5506 ; THIS TEST VERIFIES THAT THE DUT'S IAUTO FUNCTION BEHAVES CORRECTLY
5507 ; WHEN ACTIVE, IE IAUTO ASSERTED HIGH.
5508 ; ALL ACTIVE LINES ARE TESTED INDIVIDUALLY BY FILLING THE FIFO, AND
5509 ; CHECKING FOR THE PRESENCE OF AT LEAST ONE XOFF(ASCII DC3) CHARACTER
5510 ; AND ONE XON (ASCII DC1) CHARACTER.
5511 ; ANY BMP CODES THAT ARE FOUND WILL BE PLACED ON THE BMP CODE QUEUE,
5512 ; TO BE REPORTED LATER.
5513 ; THE CHARACTERS ARE TRANSMITTED ON ALL ACTIVE LINES, IN INTERNAL
5514 ; LOOPBACK MODE.
5515 ;
5516 ;-----
5517 ;
5518 023366          BGNTST
5519 023366          SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.          T7::
5520 023366          012700 000240          ;
5521 023372          104441          ;
5522 000007          TNUM == TNUM + 1          ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
5523 023374          012767 000007 156660  MOV  #TNUM,TSTNUM          ;SET UP THE TEST NUMBER.          (52)
5524 023402          012767 177777 156650  MOV  #-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
5525 023410          012767 000001 160442  MOV  #1,ERRTYP          ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
5526 023416          012767 012121 160436  MOV  #5201,ERRNBR          ;SET ERROR NUMBER TO 5201.
5527 023424          012767 006461 160432  MOV  #EM5201,ERRMSG          ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
5528 023432          012767 012634 160426  MOV  #ER9101,ERRBLK          ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
5529 ;
5530 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
5531 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
5532 ; THIS SUBROUTINE REPORTS ERROR >>>> 5201 <<<<<.
5533 023440          004767 170130          JSR  PC,CLRST          ;RESET THE DMU-11. REPORT ANY ERRORS FOUND.
5534 023444          103402          BCS  .+6
5535 023446          000167 000400          JMP  608          ;EXIT TEST IF FATAL ERROR FOUND.
5536 ;
5537 ; INITIALIZE THE 256 BYTE DATA PATTERN.
5538 ; ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
5539 ; NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
5540 ;
5541 023452          004767 170514          JSR  PC,INDTPX          ;INITIALISE DATA PATTERN.
5542 ;
5543 ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
5544 ; SET LPR TO 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
5545 ;
5546 023456          016705 156546          MOV  ACTLNS,R5          ;PASS THE ACTIVE LINE BIT MAP.
5547 023462          012700 000206          MOV  #206,R0          ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
5548 023466          004767 173156          JSR  PC,WTWLNLC          ;INITIALISE THE LINE CONTROL REGISTER.
5549 023472          012700 177670          MOV  #177670,R0          ;PASS THE LPR CONTENTS.
5550 023476          004767 173176          JSR  PC,WTWLPRL          ;SET THE LPR CONTENTS TO 38.4K BAUD.
5551 023502          012704 000012          MOV  #10.,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
5552 023506          004767 170220          JSR  PC,DELAY          ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
5553 ;
5554 ;
5555 ; SET UP LOOP FOR ALL ACTIVE LINES.
    
```

```

5556 ; TEST THE STATE OF THE OAUTO BIT PRIOR TO TRANSMITTING THE DATA PATTERN.
5557 ; IF THE BIT IS CLEAR, THEN REPORT THE ERROR AND SKIP TRANSMITTING
5558 ; THE DATA PATTERN ON THE SELECTED LINE.
5559 ; TRANSMIT A 224 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
5560 ; EMPTY THE FIFO, AND COUNT THE XOFF AND AN XON CHARS FOUND.
5561 ;-
5562 023512 005001          CLR    R1          ;CLEAR THE LINE NUMBER COUNTER.
5563 023514 005067 000330  CLR    55$        ;CLEAR STORAGE FOR LINE NUMBER.
5564 023520 012767 012122 160334 2$: MOV    #5202,ERRNBR ;SET THE ERROR NUMBER TO 5202.
5565 023526 004767 171230      JSR    PC,PUFIFO   ;PURGE THE FIFO.
5566 023532 103143          BCC    50$        ;GO REPORT ERROR IF FIFO DID NOT PURGE.
5567 023534 000241          CLC                    ;CLEAR CARRY PRIOR TO ROTATING BIT MAP.
5568 023536 006005          ROR    R5          ;ROTATE THE BIT MAP INTO THE CARRY BIT.
5569 023540 103131          BCC    16$        ;BRANCH IF LINE IS INACTIVE.
5570
5571 ;+
5572 ; TEST THE IAUTO BIT ON THE SELECTED ACTIVE LINE.
5573 ; REPORT ERROR IF IT IS CLEAR.
5574 ; DO NOT TRANSMIT THE DATA PATTERN ON THE SELECTED LINE.
5575 023542 005267 160314      INC    ERRNBR      ;SET ERROR NUMBER TO 5203.
5576 023546 010177 156464      MOV    R1,BCSRA   ;SELECT LINE TO TEST.
5577 023552 032777 000002 156466  BIT    #BIT1,BLNCTRA ;TEST THE STATE OF THE IAUTO BIT ON THIS LINE.
5578 023560 001010          BNE    4$         ;SKIP ERROR IF IAUTO BIT SET.
5579 023562 012702 006514      MOV    #EM5202,R2 ;PASS THE CORRECT ERROR MESSAGE.
5580 ; "IAUTO BIT FOUND CLEAR ON LINE NN"
5581 023566          ERROR          ; >>>>> ERROR <<<<<<.
5582 023566 104460          TRAP    C#ERROR
5583
5584 ;+
5585 ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5586 023570 032767 000100 156422  BIT    #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5587 023576 001525          BEQ    60$        ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5588 ; DURING THE SOFTWARE QUESTIONS.
5589
5590 023600 000511          BR     16$        ;SKIP TRANSMITTING DATA PATTERN.
5591
5592 ;+
5593 ; TRANSMIT DATA PATTERN TO FILL THE FIFO, 223 CHARS + 32 XOFF'S + XON.
5594 ;-
5595 023602 005267 160254 4$: INC    ERRNBR      ;SET ERROR NUMBER TO 5204.
5596 023606 012702 002660      MOV    #BUFBAS,R2 ;PASS THE START OF THE DATA PATTERN TO TX.
5597 023612 012703 000337      MOV    #223,,R3   ;PASS THE LENGTH OF THE DATA PATTERN.
5598 023616 004767 170150      JSR    PC,DODMA   ;TRANSMIT THE DATA PATTERN.
5599 023622 103107          BCC    50$        ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
5600
5601 ;+
5602 ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER PLUS XOFF
5603 ; TO ARRIVE IN THE FIFO.
5604 ;-
5605 023624 005267 160232      INC    ERRNBR      ;SET ERROR NUMBER TO 5205.
5606 023630 012701 170454      MOV    #170454,R1 ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
5607 023634 016702 156376      MOV    CSRA,R2    ;PASS THE ADDRESS OF THE CSR.
5608 023640 004767 172670      JSR    PC,WAIBIS  ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
5609 023644 103076          BCC    50$        ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
5610 023646 012704 000012      MOV    #10,,R4   ;PASS DELAY OF 10 MILLI SECS.
5611 023652 004767 170054      JSR    PC,DELAY   ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.

```

```

5612
5613
5614
5615
5616 023656 005003
5617 023660 005004
5618 023662 005267 160174
5619 023666 012701 000400
5620 023672 017702 156342
5621 023676 100061
5622
5623
5624
5625 023700 012700 170301
5626 023704 040200
5627 023706 001002
5628 023710 004767 171566
5629
5630
5631
5632 023714 120227 000023
5633 023720 001001
5634 023722 005203
5635 023724 120227 000021
5636 023730 001001
5637 023732 005204
5638 023734 005301
5639 023736 001412
5640
5641
5642
5643
5644 023740 020127 000176
5645 023744 001352
5646 023746 010400
5647
5648 023750 012704 000001
5649 023754 004767 167752
5650 023760 010004
5651 023762 000743
5652
5653
5654
5655
5656 023764 005703
5657 023766 001403
5658 023770 020427 000001
5659 023774 001413
5660 023776 005267 160060
5661 024002 016701 000042
5662 024006 012702 006431
5663
5664 024012
5664 024012 104460
5665
5666
5667

;+
; READ 256 CHARS FROM THE FIFO, COUNT ANY XOFF OR XON CHARS FOUND.
;-
CLR R3 ;CLEAR XOFF COUNTER.
CLR R4 ;CLEAR XON COUNTER.
INC ERRNBR ;INCREMENT ERROR NUMBER TO 5206.
MOV #256,R1 ;INITIALISE THE READ COUNTER.
6#: MOV BRBUFA,R2 ;READ CHAR FROM THE FIFO.
BPL 50# ;GO REPORT ERROR IF FIFO EMPTY.

;+
; CHECK FOR BMP CODE IN THE FIFO. SAVE ANY FOUND ON THE QUEUE.
;-
MOV #170301,R0 ;SET UP BMP BIT MASK.
BIC R2,R0 ;TRY TO CLEAR ALL THE BMP BITS.
BNE 8# ;SKIP BMP SAV IF NOT A BMP CODE.
JSR PC,SAVBMP ;SAVE THE BMP CODE ON THE QUEUE.

;+
; CHECK FOR XOFF AND XON CHARACTERS.
;-
8#: CMPB R2,#23 ;IS IT AN XOFF CHARACTER?.
BNE 10# ;NO, BRANCH TO SEE IF IT IS AN XON.
INC R3 ;COUNT THE XOFF CHAR.
10#: CMPB R2,#21 ;IS IT AN XON CHARACTER?.
BNE 12# ;NO, SKIP THE NEXT INSTRUCTION.
INC R4 ;COUNT THE XON.
12#: DEC R1 ;DECREMENT THE READ COUNT.
BEQ 13# ;BRANCH IF ALL CHARACTERS READ.

;+
; CHECK IF THE FIFO HAS BEEN EMPTIED BELOW THE HALF LEVEL, IF IT
; HAS DELAY FOR 1MS TO ALLOW THE XON TO BE GENERATED.
;-
CMP R1,#126. ;IS THE FIFO LEVEL = 126 ?
BNE 6# ;LOOP TO READ THE NEXT CHARACTER IF NOT.
MOV R4,R0 ;SAVE THE XON COUNT, ALTHOUGH THERE SHOULDN'T
;BE ANY.
MOV #1,R4 ;SET THE DELAY TO 1MS.
JSR PC,DELAY ;PERFORM THE DELAY.
MOV R0,R4 ;RESTORE THE XON COUNT.
BR 6# ;LOOP TO READ THE NEXT CHAR.

;+
; VERIFY THAN AT LEAST 1 XOFF AND 1 XON WAS FOUND IN THE FIFO.
; REPORT ERROR IF NONE WERE FOUND.
;-
13#: TST R3 ;CHECK XOFF COUNT.
BEQ 14# ;GO REPORT ERROR IF NONE FOUND.
CMP R4,#1 ;CHECK XON COUNT = 1.
BEQ 16# ;SKIP THE ERROR REPORT IF ONE XON WAS FOUND.
14#: INC ERRNBR ;SET ERROR NUMBER TO 5207.
MOV 55#,R1 ;PASS THE LINE NUMBER TO BE REPORTED.
MOV #EM5103,R2 ;PASS THE ERROR MESSAGE TO BE REPORTED.
; "IAUTO BIT BAD ON LINE NN".
; >>>> ERROR <<<<<.
; ERROR TRAP C#ERROR

;+
; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED

```


5688
 5689
 5690
 5691
 5692
 5693
 5694
 5695
 5696
 5697
 5698
 5699
 5700
 5701
 5702
 5703
 5704 024060
 024060
 5705 024060
 024060 012700 000240
 024064 104441
 5706 000010
 5707 024066 012767 000010 156166
 5708 024074 012767 177777 156156
 5709 024102 012767 000001 157750
 5710 024110 012767 012265 157744
 5711 024116 012767 006552 157740
 5712
 5713
 5714
 5715
 5716
 5717 024124 004767 167444
 5718 024130 103113
 5719
 5720
 5721
 5722
 5723 024132 004767 167724
 5724 024136 103110
 5725 024140 004767 167776
 5726
 5727
 5728
 5729
 5730
 5731
 5732
 5733
 5734 024144 012700 000204
 5735 024150 004767 172474
 5736 024154 012700 177670
 5737 024160 004767 172514
 5738 024164 012704 000012
 5739 024170 004767 167536
 5740 024174 012702 002660
 5741 024200 012703 000400

```
.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.
;*****
BGNTST
;*****
; T8::
SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
;*****
; MOV #PRI05,R0
; TRAP C$SPRI
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (53)
MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #5301,ERRNBR ;SET ERROR NUMBER TO 5301.
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 <<<<.
;*****
JSR PC,CLNRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
BCC 60# ;EXIT TEST IF FATAL ERROR FOUND.
;*****
; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
; INITIALISE 256 BYTE DATA PATTERN.
;*****
JSR PC,FINACT ;FIND AN ACTIVE LINE.
BCC 60# ;EXIT IF NO ACTIVE LINES FOUND.
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.
;*****
MOV #204,R0 ;PASS PARAMETER FOR INTERNAL LOOPBACK,ENABLE RX.
JSR PC,WTLNC ;INITIALISE THE LINE CONTROL REGISTER.
MOV #177670,R0 ;PASS THE LPR CONTENTS.
JSR PC,WTLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
MOV #10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
JSR PC,DELAY ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
MOV #BUFBAS,R2 ;PASS THE START OF THE DATA PATTERN TO TX.
MOV #BUFMID-BUFBAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN
```

```

5742 024204 005267 157652          INC   ERRNBR      ;SET ERROR NUMBER TO 5302.
5743 024210 004767 167556          JSR   PC,DODMA   ;TRANSMIT THE DATA PATTERN.
5744 024214 103057                   BCC   50$        ;ABORT TEST IF ERROR FOUND DURING DMA TX.
5745                                     ;*
5746                                     ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
5747                                     ; THE FIFO.
5748                                     ;-
5749 024216 005267 157640          INC   ERRNBR      ;SET ERROR NUMBER TO 5303.
5750 024222 010103                   MOV   R1,R3      ;SAVE THE NUMBER OF THE SELECTED ACTIVE LINE.
5751 024224 012701 170536          MOV   #170536,R1 ;PASS TIME-OUT VALUE OF 350 MILLI SECS.
5752 024230 016702 156002          MOV   CSRA,R2   ;PASS THE ADDRESS OF THE CSR.
5753 024234 004767 172274          JSR   PC,WAIBIS  ;WAIT FOR DMA TO COMPLETE, TX ACTION SET.
5754 024240 103045                   BCC   50$        ;BRANCH IF FIFO EMPTY, ABORT THE TEST.
5755 024242 012704 000005          MOV   #5,R4     ;PASS DELAY OF 5 MILLI SECS.
5756 024246 004767 167460          JSR   PC,DELAY   ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
5757                                     ;*
5758                                     ; READ THE FIFO CHECKING FOR DATA CORRUPTION, REPORT ANY ERRORS FOUND.
5759                                     ; ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
5760                                     ;-
5761 024252 006303                   ASL   R3         ;MULTIPLY BY 2.
5762 024254 005004                   CLR   R4         ;INITIALISE THE EXPECTED DATA.
5763 024256 016705 155756          MOV   RBUFA,R5  ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
5764 024262 012767 012270 157572 2$: MOV   #5304,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
5765 024270 011502                   MOV   (R5),R2   ;GET THE ACTUAL DATA FROM THE FIFO.
5766 024272 100030                   BPL   50$        ;ABORT THE TEST IF THE FIFO IS EMPTY.
5767                                     ;*
5768                                     ; CHECK IF THE READ CHARACTER IS A BMP CODE.
5769                                     ; IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
5770                                     ; ABORT THE TEST.
5771                                     ;-
5772 024274 005267 157562          INC   ERRNBR      ;SET ERROR NUMBER TO 5305.
5773 024300 004767 167170          JSR   PC,CHKBMP ;CHECK IF CHARACTER IS A BMP CODE.
5774 024304 103002                   BCC   4$        ;BRANCH IF NOT A BMP CODE.
5775 024306 104460                   ERROR >>>> ERROR 5305 <<<<<.
5776                                     TRAP C$ERROR
5777 024310 000423                   BR    60$       ;ABORT THIS TEST.
5778                                     ;-
5779 024312 005267 157544          4$: INC   ERRNBR      ;SET ERROR NUMBER TO 5306.
5780 024316 120402                   CMPB  R4,R2     ;COMPARE THE EXPECTED WITH THE ACTUAL DATA.
5781 024320 001412                   BEQ   8$        ;SKIP ERROR REPORT IF DATA IS OK.
5782 024322 012767 012456 157536  MOV   #ER9002,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
5783 024330 012701 006606          MOV   #EM5302,R1 ;PASS THE MESSAGE TO BE REPORTED.
5784                                     ;REPORT THE ERROR "FIFO BAD, DATA FIELD CORRUPTED"
5785 024334 104460          6$: ERROR >>>> ERROR 5306 <<<<<.
5786                                     TRAP C$ERROR
5787                                     ;*
5788                                     ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
5789                                     ;-
5790 024336 032767 000100 155654  BIT   #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
5791 024344 001405                   BEQ   60$       ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
5792                                     ;DURING THE SOFTWARE QUESTIONS.
5793                                     ;-
5794 024346 105204          8$: INCB  R4        ;INCREMENT THE EXPECTED DATA.
5795 024350 001344                   BNE  2$        ;LOOP IF NOT DONE.
5796 024352 000402                   BR   60$       ;EXIT

```

5797
5798 024354 004767 171360
5799 024360 005067 155674
5800
5801 024364
024364
024364 104401

50\$: JSR PC,TSABRT
60\$: CLR CTRLCF

ENDTST

;ABORT THE TEST, REASON SHOWN BY ERROR NUMBER.
;INDICATE THAT WE ARE NOT WITHIN A TEST.

L10032: TRAP C#ETST

5803
 5804
 5805
 5806
 5807
 5808
 5809
 5810
 5811
 5812
 5813
 5814
 5815
 5816
 5817
 5818
 5819 024366
 024366
 5820 024366 012700 000240
 024366 104441
 024372 000011
 5821
 5822 024374 012767 000011 155660
 5823 024402 012767 177777 155650
 5824 024410 012767 000001 157442
 5825 024416 012767 012431 157436
 5826 024424 012767 006737 157432
 5827 024432 012767 011604 157426
 5828
 5829
 5830
 5831
 5832
 5833 024440 004767 167130
 5834 024444 103111
 5835
 5836
 5837
 5838 024446 004767 167410
 5839 024452 103106
 5840
 5841
 5842
 5843
 5844
 5845
 5846 024454 004767 167512
 5847
 5848
 5849
 5850
 5851
 5852
 5853
 5854
 5855 024460 012700 000206
 5856 024464 004767 172160

```

.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
SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.          T9::
MOV     #PRI05,R0      MOV     #PRI05,R0
TRAP   C#SPRI         TRAP   C#SPRI

TNUM == TNUM + 1      ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV     #TNUM,TSTNUM  ;SET UP THE TEST NUMBER.          (54)
MOV     #-1,CTRLCF    ;INDICATE THAT WE ARE IN A TEST.
MOV     #1,ERRTYP     ;SET FATAL ERROR TYPE IN ERROR TABLE.
MOV     #5401.,ERRNBR ;SET ERROR NUMBER TO 5401.
MOV     #EMS401,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
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 <<<<.
;*
;*   JSR     PC,CLNRST    ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
;*   BCC     60#         ;EXIT TEST IF FATAL ERROR FOUND.
;*
;*   ;*
;*   ;*   FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
;*   ;*
;*   ;*   JSR     PC,FINACT  ;FIND THE NUMBER OF THE FIRST ACTIVE LINE.
;*   ;*   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.
;*   ;*
;*   ;*   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.
;*   ;*
;*   ;*   MOV     #206,R0    ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
;*   ;*   JSR     PC,WTLNCR ;INITILAISE THE LINE CONTROL REGISTER.

```

```

5857 024470 012700 177670      MOV    #177670,R0      ;PASS THE LPR CONTENTS.
5858 024474 004767 172200      JSR    PC,WTWLPB      ;SET THE LPR CONTENTS TO 38.4K BAUD.
5859 024500 012704 000012      MOV    #10.,R4        ;PASS DELAY TIME OF 10 MILLI SECONDS.
5860 024504 004767 167222      JSR    PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
5861 024510 012702 002660      MOV    #BUFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
5862 024514 012703 000277      MOV    #191.,R3       ;PASS THE LENGTH OF THE DATA PATTERN.
5863 024520 004767 167246      JSR    PC,DODMA       ;TRANSMIT THE DATA PATTERN.
5864 024524 103057              BCC    50#            ;IF ERROR FOUND DURING DMA THEN ABORT TEST.
5865
5866
5867
5868
5869
5870 024526 005267 157330      ;*
; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
; THE FIFO.
5871 024532 012701 170454      ;-
      INC    ERRNBR      ;SET ERROR NUMBER TO 5402.
5872 024536 016702 155474      MOV    #170454,R1     ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
5873 024542 004767 171766      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
5874 024546 103046              JSR    PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
5875 024550 012704 000005      BCC    50#            ;IF FIFO EMPTY, REPORT ERROR, ABORT THE TEST.
5876 024554 004767 167152      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
5877
5878
5879
5880
5881
5882
5883
5884
5885 024560 005004              ;*
; READ THE CONTENTS OF THE FIFO. IF ANY OF THE FOLLOWING CONDITIONS OCCUR
5886 024562 016705 155452      ; REPORT THE ERROR AND ABORT THE TEST;
5887 024566 012767 012433 157266 2# ; FIFO EMPTY TOO SOON.
5888 024574 011502              ; BMP CODE FOUND.
5889 024576 100032              ; XOFF CODE FOUND.
5890 024600 005204              ; EXTRA (192) CHARACTER FOUND IN FIFO.
5891
5892
5893
5894
5895
5896 024602 005267 157254      ;-
      CLR    R4          ;CLEAR THE CHARACTER COUNT.
5897 024606 004767 166662      MOV    RBUFA,R5       ;GET THE ADDRESS OF THE RECEIVER BUFFER REG.
5898 024612 103001              MOV    #5403.,ERRNBR ;SET ERROR NUMBER TO 5403.
5899
5900 024614 000421              MOV    (R5),R2        ;GET THE ACTUAL DATA FROM THE FIFO.
5901
5902
5903
5904
5905 024616 005267 157240      BPL    50#            ;FIFO EMPTY, ABORT TEST.
5906 024622 122702 000023      INC    R4             ;COUNT THE CHARACTER.
5907 024626 001003              ;*
; CHECK IF THE READ CHARACTER IS A BMP CODE.
5908 024630 012701 007005      ; IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
; ABORT THE TEST.
5909
5910 024634 000411              ;-
      INC    ERRNBR      ;SET ERROR NUMBER TO 5404.
5911
5912 024636 005267 157220      JSR    PC,CHKBMP      ;CHECK IF CHARACTER IS A BMP CODE.
5913 024642 020427 000277      BCC    4#            ;BRANCH IF NOT A BMP CODE.
;REPORT ERROR "BMP CODE FOUND IN FIFO, TEST INVALIDATED".
;REPORT THE ERROR AND ABORT THE TEST.
5905 024616 005267 157240      4# : INC    ERRNBR      ;SET ERROR NUMBER TO 5405.
5906 024622 122702 000023      CMPB   #23,R2         ;CHECK IF THE READ DATA IS AN XOFF
5907 024626 001003              BNE    6#            ;BRANCH IF NOT AN XOFF.
5908 024630 012701 007005      MOV    #EM5402,R1     ;PASS THE MESSAGE TO BE REPORTED.
5909
5910 024634 000411              ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
5911
5912 024636 005267 157220      BR     8#            ;GO REPORT THE ERROR AND ABORT THE TEST.
5913 024642 020427 000277      6# : INC    ERRNBR      ;SET ERROR NUMBER TO 5406.
      CMP    R4,#191.   ;CHECK IF WE HAVE READ ALL THE CHARACTERS.

```

```

5914 024646 001347      BNE      2$      ;LOOP BACK TO GET THE NEXT CHARACTER.
5915 024650 011502      MOV      (R5),R2 ;TRY TO READ AN EXTRA CHARACTER FROM THE FIFO.
5916 024652 100006      BPL      60$     ;EXIT IF NON FOUND.
5917 024654 012701 007005  MOV      @EM5402,R1 ;PASS THE MESSAGE TO BE REPORTED.
5918                                ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
5919
5920 024660      8$:      ERROR      ;          >>>> ERRORS 5304 THRU 5306 <<<<<.
      024660 104460      BR          TRAP      C#ERROR
5921 024662 000402      BR          60$     ;EXIT THE TEST.
5922
5923                                ;          >>>> ERRORS 5402 AND 5403 <<<<<.
5924 024664 004767 171050 50$:      JSR      PC,TSABRT ;REPORT TEST ABORTED. NON-TEST RELATED ERROR.
5925 024670 005067 155364 60$:      CLR      CTRLCF ;INDICATE THAT WE ARE NOT WITHIN A TEST.
5926
5927 024674      ENDTST
      024674
      024674 104401      L10033:
                                TRAP      C#ETST

```

```

5929 .SBTTL HARDWARE TEST - FI3QLA -
5930 ;*****
5931 ;
5932 ;
5933 ;
5934 ; THIS TEST VERIFIES THAT THE DUT'S FIFO 3/4 LEVEL ALARM SYSTEM
5935 ; BECOMES ACTIVE WHEN THE FIFO CONTAINS > 192 CHARACTERS.
5936 ; THE TEST COMPARES THE ACTUAL NUMBER OF XOFF (ASCII DC3)
5937 ; CHARACTERS THAT ARE FOUND IN THE FIFO WITH THE EXPECTED NUMBER.
5938 ; AN ERROR WILL BE REPORTED, IF THE COUNTS ARE FOUND TO DIFFER.
5939 ; ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
5940 ; HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
5941 ; REPORTED LATER.
5942 ; THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
5943 ; INTERNAL LOOPBACK MODE.
5944 ;
5945 ;*****
5946 BGNTST
5947 SETPRI #PRI05 ;ALLOW LTC INTERRUPTS. T10::
5948 ;
5949 ;
5950 ;
5951 ;
5952 ;
5953 ;
5954 ;
5955 ;
5956 ;
5957 ;
5958 ;
5959 ;
5960 ;
5961 ;
5962 ;
5963 ;
5964 ;
5965 ;
5966 ;
5967 ;
5968 ;
5969 ;
5970 ;
5971 ;
5972 ;
5973 ;
5974 ;
5975 ;
5976 ;
5977 ;
5978 ;
5979 ;
5980 ;
5981 ;
5982 ;

```

Address	OpCode	Op1	Op2	Op3	Op4	Comment
5946	BGNTST					
5947	SETPRI	#PRI05				;ALLOW LTC INTERRUPTS. T10::
5948						
5949	MOV	#TNUM,TSTNUM				;INCREMENT THE ASSEMBLY TIME TEST COUNTER. (55)
5950	MOV	#-1,CTRLCF				;SET UP THE TEST NUMBER.
5951	MOV	#1,ERRTYP				;INDICATE THAT WE ARE IN A TEST.
5952	MOV	#5501,ERRNBR				;SET ERROR TYPE AS FATAL IN ERROR TABLE.
5953	MOV	#EM5501,ERRMSG				;SET ERROR NUMBER TO 5501.
5954						;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
5955						; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
5956						; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
5957						; THIS SUBROUTINE REPORTS ERROR >>>> 5501 <<<<<.
5958						
5959	JSR	PC,CLNRST				;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
5960	BCS	.+6				;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
5961	JMP	60+				;EXIT TEST FATAL ERROR FOUND.
5962						
5963						; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
5964						
5965	JSR	PC,FINACT				;FIND AN ACTIVE LINE.
5966	BCS	.+6				;SKIP EXIT OF TEST IF ACTIVE LINE FOUND.
5967	JMP	60+				;EXIT TEST.
5968						
5969						; INITIALIZE THE 256 BYTE DATA PATTERN.
5970						; ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
5971						; NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
5972						
5973	JSR	PC,INDTPX				;INITIALISE DATA PATTERN.
5974						
5975						; TRANSMIT A 191 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
5976						; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
5977						
5978						
5979						; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
5980						; TRANSMIT THE FIRST 191 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
5981						
5982	INC	ERRNBR				;SET ERROR NUMBER TO 5502.

```

5983 024776 012700 000206      MOV      #206,R0          ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
5984 025002 004767 171642      JSR      PC,WTWLNCR      ;INITIALISE THE LINE CONTROL REGISTER.
5985 025006 012700 177670      MOV      #177670,R0     ;PASS THE LPR CONTENTS.
5986 025012 004767 171662      JSR      PC,WTWLPR      ;SET THE LPR CONTENTS TO 38.4K BAUD.
5987 025016 012704 000012      MOV      #10.,R4        ;PASS DELAY TIME OF 10 MILLI SECONDS.
5988 025022 004767 166704      JSR      PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
5989 025026 010105              MOV      R1,R5          ;COPY THE LINE NUMBER.
5990 025030 012702 002660      MOV      #8UFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
5991 025034 012703 000277      MOV      #191.,R3       ;PASS THE LENGTH OF THE DATA PATTERN.
5992 025040 004767 166726      JSR      PC,DODMA       ;TRANSMIT THE DATA PATTERN.
5993 025044 103147              BCC      50$           ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
5994
5995
5996                          ;+
5996                          ; WAIT FOR DMA TO COMPLETE. THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
5997                          ; THE FIFO.
5997                          ;-
5998
5999 025046 005267 157010      INC      ERRNBR         ;SET ERROR NUMBER TO 5503.
6000 025052 012701 170454      MOV      #170454,R1     ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
6001 025056 016702 155154      MOV      CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
6002 025062 004767 171446      JSR      PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6003 025066 103136              BCC      50$           ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
6004 025070 012704 000005      MOV      #5,R4          ;PASS DELAY OF 5 MILLI SECS.
6005 025074 004767 166632      JSR      PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
6006
6007                          ;+
6007                          ; TRANSMIT A NULL CHARACTER WHICH WILL CAUSE AN XOFF TO BE GENERATED.
6007                          ;-
6008
6009 025100 005267 156756      INC      ERRNBR         ;SET ERROR NUMBER TO 5504.
6010 025104 010501              MOV      R5,R1          ;PASS THE LINE NUMBER.
6011 025106 012702 002660      MOV      #8UFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
6012 025112 012703 000001      MOV      #1,R3          ;PASS THE NUMBER OF CHARACTERS TO TX.
6013 025116 004767 166650      JSR      PC,DODMA       ;TX A NULL CHARACTER TO CAUSE AN XOFF.
6014 025122 103120              BCC      50$           ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
6015
6016                          ;+
6016                          ; WAIT FOR THE DMA TO COMPLETE AND THE LAST CHAR TO ARRIVE IN THE FIFO
6016                          ;-
6017
6018 025124 005267 156732      INC      ERRNBR         ;SET ERROR NUMBER TO 5505.
6019 025130 012701 170012      MOV      #170012,R1     ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
6020 025134 016702 155076      MOV      CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
6021 025140 004767 171370      JSR      PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6022 025144 103107              BCC      50$           ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
6023 025146 012704 000005      MOV      #5,R4          ;PASS DELAY OF 5 MILLI SECS.
6024 025152 004767 166554      JSR      PC,DELAY       ;WAIT FOR XOFF TO GET INTO THE FIFO.
6025
6026                          ;+
6026                          ; INITIALISE THE 256 BYTE DATA PATTERN TO ALL NULLS.
6026                          ;-
6027
6028 025156 012702 002660      MOV      #8UFBAS,R2     ;INITIALIZE THE DATA PATTERN TO BE
6029 025162 105022              CLR      (R2)+          ; ALL NULLS.
6030 025164 020227 003260      CMP      R2,#8UFBAS    ;
6031 025170 103774              BLO      4$            ;
6032
6033                          ;+
6033                          ; TRANSMIT A FURTHER 31 NULL CHARACTERS WHICH WILL CAUSE 31 XOFF'S TO BE
6034                          ; GENERATED.
6034                          ;-
6035
6036
6037 025172 005267 156664      INC      ERRNBR         ;SET ERROR NUMBER TO 5506.
6038 025176 010501              MOV      R5,R1          ;PASS THE LINE NUMBER.
6039 025200 012702 002660      MOV      #8UFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
    
```


6096 025362 000402
6097
6098 025364 004767 170350
6099 025370 005067 154664
6100
6101 025374
025374
025374 104401

BR 60#
50#: JSR PC,TSABRT
60#: CLR CTRLCF
ENDTST

;ABORT THE TEST.
;REPORT TEST ABORTED. ERROR # SHOWS REASON.
;INDICATE THAT WE ARE NOT WITHIN A TEST.

L10034: TRAP C#ETST

```

6103 .SBTTL  HARDWARE TEST          - FI3QAI -
6104 ;*****
6105 ; - FIFO 3/4 ALARM LEVEL ACTIVE/INACTIVE TEST -
6106 ;*
6107 ;* THIS TEST VERIFIES THAT THE DUT'S FIFO 3/4 LEVEL ALARM SYSTEM
6108 ;* BECOMES ACTIVE AND INACTIVE AT THE CORRECT LEVELS.
6109 ;* ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
6110 ;* HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
6111 ;* REPORTED LATER.
6112 ;* THE CHARACTERS ARE TRANSMITTED ON THE FIRST AVAILABLE ACTIVE LINE, IN
6113 ;* INTERNAL LOOPBACK MODE.
6114 ;*
6115 ;-----*****
6116
6117 025376          BGNTST
        025376
6118 025376          SETPRI  #PRI05          ;ALLOW LTC INTERRUPTS.          T11::
        025376 012700 000240
        025402 104441
        000013
6119          TNUM == TNUM + 1          ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
6120 025404 012767 000013 154650          MOV  #TNUM,TSTNUM          ;SET UP THE TEST NUMBER.          (56)
6121 025412 012767 177777 154640          MOV  #-1,CTRLCF          ;INDICATE THAT WE ARE IN A TEST.
6122 025420 012767 000001 156432          MOV  #1,ERRTYP          ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
6123 025426 012767 012741 156426          MOV  #5601,ERRNBR          ;SET ERROR NUMBER TO 5601.
6124 025434 012767 007112 156422          MOV  #EM5601,ERRMSG          ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
6125
6126 ;*
6127 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
6128 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
6129 ; THIS SUBROUTINE REPORTS ERROR >>>> 5601 <<<<<.
6130 025442 004767 166126
6131 025446 103402
6132 025450 000167 000412
6133 025454
20:
6134 ;*
6135 ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
6136 ;*
6137 025454 004767 166402
6138 025460 103402
6139 025462 000167 000400
6140
6141 ;*
6142 ; INITIALIZE THE 256 BYTE DATA PATTERN.
6143 ; ENSURE THE DATA PATTERN IS FREE FROM XON'S OR XOFF'S TO PREVENT ERRORS.
6144 ; NOTE: THE FIRST TWO CHARACTERS AND THE LAST TWO CHARACTERS WILL BE THE SAME.
6145 025466 004767 166500
6146 ;*
6147 ; TRANSMIT A 256 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
6148 ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
6149 ;*
6150 ;*
6151 ; SET INTERNAL LOOPBACK, ENABLE IAUTO AND RECEIVER ON THE SELECTED LINE.
6152 ; TRANSMIT THE FIRST 191 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
6153 ;*
6154 025472 005267 156364          INC  ERRNBR          ;SET ERROR NUMBER TO 5602.
6155 025476 012700 000206          MOV  #206,RO          ;PASS INTERNAL LOPBCK, ENABLE RX AND IAUTO.
6156 025502 004767 171142          JSR  PC,WTWLNCR          ;INITIALISE THE LINE CONTROL REGISTER.
    
```

```

6157 025506 012700 177670      MOV    #177670,R0      ;PASS THE LPR CONTENTS.
6158 025512 004767 171162      JSR    PC,WTMLPR      ;SET THE LPR CONTENTS TO 38.4K BAUD.
6159 025516 012704 000012      MOV    #10.,R4        ;PASS DELAY TIME OF 10 MILLI SECONDS.
6160 025522 004767 166204      JSR    PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
6161 025526 010105              MOV    R1,R5          ;COPY THE LINE NUMBER.
6162 025530 012702 002660      MOV    #BUFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
6163 025534 012703 000277      MOV    #191.,R3       ;PASS THE LENGTH OF THE DATA PATTERN.
6164 025540 004767 166226      JSR    PC,DODMA       ;TRANSMIT THE DATA PATTERN.
6165 025544 103146              BCC    50#            ;EXIT IF ERROR FOUND DURING DMA TX.
6166
6167      ;*
6168      ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
6169      ; THE FIFO.
6170 025546 005267 156310      INC    ERRNBR         ;SET ERROR NUMBER TO 5603.
6171 025552 012701 170454      MOV    #170454,R1     ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
6172 025556 016702 154454      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
6173 025562 004767 170746      JSR    PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6174 025566 103135              BCC    50#            ;BRANCH IF FIFO EMPTY, ABORT THE TEST.
6175 025570 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
6176 025574 004767 166132      JSR    PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
6177
6178
6179      ;*
6180      ; TRANSMIT A NULL CHARACTER WHICH WILL CAUSE AN XOFF TO BE GENERATED.
6181 025600 005267 156256      INC    ERRNBR         ;SET ERROR NUMBER TO 5604.
6182 025604 010501              MOV    R5,R1          ;PASS THE LINE NUMBER.
6183 025606 012702 002660      MOV    #BUFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
6184 025612 012703 000001      MOV    #1,R3          ;PASS THE NUMBER OF
6185 025616 004767 166150      JSR    PC,DODMA       ;TX A NULL CHARACTER TO CAUSE AN XOFF.
6186 025622 103117              BCC    50#            ;ABORT THE TEST IF ERROR FOUND DURING DMA TX.
6187
6188      ;*
6189      ; WAIT FOR THE XOFF TO BE RECEIVED BEFORE CONTINUING THE TEST.
6190 025624 005267 156232      INC    ERRNBR         ;SET ERROR NUMBER TO 5605.
6191 025630 012701 170012      MOV    #170012,R1     ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
6192 025634 016702 154376      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
6193 025640 004767 170670      JSR    PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6194 025644 103106              BCC    50#            ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
6195 025646 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
6196 025652 004767 166054      JSR    PC,DELAY       ;WAIT FOR XOFF TO GET INTO THE FIFO.
6197
6198 025656 010577 154354      MOV    R5,@CSRA       ;SELECT THE LINE READY FOR TRANSMISSION.
6199
6200      ;*
6201      ; READ THREE CHARACTERS, TRANSMIT ONE CHARACTER UNTIL THE FIRST 192 CHARACTERS
6202      ; HAVE BEEN READ FROM THE FIFO, IE UNTIL THE HALF LEVEL IS REACHED.
6203      ; THEN READ THE FIFO UNTIL EMPTY.
6204      ; COUNT ALL XOFF'S THAT ARE DETECTED.
6205 025662 005005              CLR    R5             ;CLEAR THE TX FLAG.
6206 025664 005004              CLR    R4             ;CLEAR THE CHARACTER COUNTER.
6207 025666 012703 000300      MOV    #192.,R3       ;SET UP READ COUNTER FOR THE FIRST 192 CHARS.
6208
6209 025672 012700 000003      4#:   MOV    #3,R0        ;SET READ COUNTER.
6210 025676 012701 170005      6#:   MOV    #170005,R1    ;INDICATE TO TEST DATA.VALID BIT, TIME-OUT SMS.
6211 025702 016702 154332      MOV    RBUFA,R2       ;INDICATE TO CHECK RECEIVE BUFFER REGISTER.
6212 025706 004767 170622      JSR    PC,WAIBIS      ;WAIT FOR RECEIVED CHAR OR TIME-OUT.
6213 025712 103046              BCC    14#            ;EXIT LOOP IF TIME-OUT, FIFO EMPTY.

```

```

6214 025714 005300          DEC    R0          ;DECREMENT READ COUNTER.
6215 025716 005303          DEC    R3          ;DECPEMENT CHAR COUNTER.
6216 025720 003002          BGT    8#         ;SKIP DISBL'G TX IF FIRST 192 CHARS NOT READ.
6217 025722 052705 100000  BIS    #BIT15,R5  ;DISABLE ANY FURTHER TRANSMISSIONS.
6218
6219          ;*
6220          ; CHECK IF THE READ CHARACTER IS A BMP CODE.
6221          ; IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
6222          ; ABORT THE TEST.
6223 025726 012767 012746 156126 8#:  MOV    #5606.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
6224 025734 004767 165534          JSR    PC,CHKBMP  ;CHECK IF CHARACTER IS A BMP CODE.
6225 025740 103446          BCS    16#        ;GO REPORT ERROR AND ABORT TEST IF BMP FOUND.
6226
6227          ;*
6228          ; CHECK FOR XOFF CHARACTER. IF ONE IS FOUND, COUNT IT.
6229          ; TRANSMIT A NULL CHARACTER UNTIL THE FIRST 192 CHARS HAVE BEEN READ.
6230 025742 122702 000023 10#:  CMPB   #23,R2      ;CHECK IF THE RECEIVED CHARACTER WAS AN XOFF.
6231 025746 001001          BNE    12#        ;BRANCH IF CHARACTER WAS NOT AN XOFF.
6232 025750 005204          INC    R4         ;INCREMENT THE XOFF CHAR FOUND COUNTER.
6233
6234 025752 005700          12#:  TST    R0         ;CHECK READ COUNT, TO SEE IF A CHAR CAN BE TX.
6235 025754 001350          BNE    6#         ;BRANCH IF 3 CHARS HAVE NOT YET BEEN READ.
6236 025756 005705          TST    R5         ;CHECK THE TRANSMISSION ENABLED FLAG.
6237 025760 100744          BMI    4#         ;SKIP TRANSMITTING A CHARACTER IF TX DISABLED.
6238 025762 112777 000000 154254  MOVB   #0,8FDATA ;TX A NULL CHARACTER.
6239 025770 010446          MOV    R4,-(SP)   ;SAVE THE XOFF COUNT ON THE STACK.
6240
6241          ;*
6242          ; WAIT FOR THE CHARACTER TO BE RECEIVED BEFORE CONTINUING THE TEST.
6243          ;-
6243 025772 005267 156064          INC    ERRNBR     ;SET ERROR NUMBER TO 5607.
6244 025776 012701 170012          MOV    #170012,R1 ;PASS TIME-OUT VALUE OF 10 MILLI SECS.
6245 026002 016702 154230          MOV    CSRA,R2    ;PASS THE ADDRESS OF THE CSR.
6246 026006 004767 170522          JSR    PC,WAIBIS  ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6247 026012 103023          BCC    50#        ;IF NO TX_ACTION WAS RECEIVED, ABORT THE TEST.
6248 026014 012704 000005          MOV    #5,R4     ;PASS DELAY OF 5 MILLI SECS.
6249 026020 004767 165706          JSR    PC,DELAY   ;WAIT FOR XOFF TO GET INTO THE FIFO.
6250 026024 012604          MOV    (SP)+,R4  ;RESTORE THE XOFF COUNT.
6251 026026 000721          BR    4#         ;GO RESET THE READ COUNT AND GET NEXT CHAR.
6252
6253          ;*
6254          ; CHECK IF THE CORRECT NUMBER OF XOFF'S WERE FOUND IN THE FIFO
6255          ; REPORT ERROR IF COUNT IS INCORRECT.
6256          ;-
6257 026030 012767 012750 156024 14#:  MOV    #5608.,ERRNBR ;SET ERROR NUMBER TO 5608.
6258 026036 020427 000077          CMP    R4,#63.   ;COMPARE THE EXPECTED AND ACTUAL XOFF COUNTS.
6259 026042 001411          BEQ    60#        ;EXIT TEST IF SUCCESS.
6260 026044 012767 011604 156014  MOV    #ER0503,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
6261 026052 012701 007005          MOV    #EM5402,R1 ;PASS THE MESSAGE TO BE REPORTED.
6262          ;REPORT THE ERROR "FIFO BAD, ALARM SIGNAL DEFECTIVE".
6263 026056          16#:  ERROR          ;
6264 026060 104460          TRAP   C#ERROR
6265 000402          BR    60#        ;EXIT THIS TEST.
6266 026062 004767 167652 50#:  JSR    PC,TSABRT  ;REPORT TEST ABORTED. ERROR # INDICATES FAULT.
6267 026066 005067 154166 60#:  CLR    CTRLCF    ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6268
6269 026072          ENDTST

```

C13

026072
026072 104401

L10035: TRAP C#ETST

6271
6272
6273
6274
6275
6276
6277
6278
6279
6280
6281
6282
6283
6284
6285 026074
026074
6286 026074
026074 012700 000240
026100 104441
6287 000014
6288 026102 012767 000014 154152
6289 026110 012767 177777 154142
6290 026116 012767 000001 155734
6291 026124 012767 013105 155730
6292 026132 012767 007167 155724
6293 026140 012767 011604 155720
6294
6295
6296
6297
6298
6299 026146 004767 165422
6300 026152 103402
6301 026154 000167 000364
6302 026160
6303
6304
6305
6306 026160 004767 165676
6307 026164 103167
6308
6309
6310
6311
6312
6313 026166 004767 166000
6314
6315
6316
6317
6318
6319
6320
6321
6322
6323 026172 005267 155664
6324 026176 004767 167412

```
.SBTTL  HARDWARE TEST      - FIHAVL -
;*****
;          - FIFO HALF LEVEL ACTIVE/INACTIVE TEST -
;
; THIS TEST CHECKS THAT THE DUT'S FIFO HALF 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
                                T12::
SETPRI  #PRI05                  ;ALLOW LTC INTERRUPTS.

                                MOV  #PRI05,R0
                                TRAP C#SPRI

TNUM == TNUM + 1                ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV  #TNUM,TSTNUM              ;SET UP THE TEST NUMBER. (57)
MOV  #-1,CTRLCF                ;INDICATE THAT WE ARE IN A TEST.
MOV  #1,ERRTYP                 ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV  #5701,ERRNBR              ;SET ERROR NUMBER TO 5701.
MOV  #EM5701,ERRMSG            ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
MOV  #ER0503,ERRBLK           ;SELECT THE 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 >>>> 5701 <<<<.
;
;-
JSR  PC,CLRST                 ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
BCS  2#                         ;SKIP EXITING TEST A SUCCESSFUL RESET.
JMP  60#                         ;EXIT THIS TEST.

2#:
;
; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
;
;-
JSR  PC,FINACT                ;FIND AN ACTIVE LINE.
BCC  60#                         ;EXIT IF NO ACTIVE LINES 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.
;
;-
JSR  PC,INDTPX                ;INITIALISE THE DATA PATTERN.

;
; FILL THE FIFO AND THE UART'S 3 CHAR BUFFER BY TRANSMITTING 225 CHARS
; (IE 225 + 34 XOFF'S). TRANSMIT 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 225 CHARACTERS ON THE FIRST AVAILABLE ACTIVE LINE.
;
;-
INC  ERRNBR                     ;SET ERROR NUMBER TO 5702.
JSR  PC,SETPAR                ;SET UP PARAMETERS FOR TRANSMISSION.
```

6325	026202	012700	000341		MOV	#225.,R0	;	PASS LENGTH OF DATA PATTERN.
6326	026206	004767	167640		JSR	PC,TXDATP	;	TRANSMIT DATA PATTERN.
6327	026212	103152			BCC	50#	;	EXIT IF ERROR FOUND DURING TX.
6328	026214	010105			MOV	R1,R5	;	COPY THE LINE NUMBER.
6329								
6330					;		;	WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
6331					;		;	THE FIFO.
6332					;			
6333	026216	005267	155640		INC	ERRNBR	;	SET ERROR NUMBER TO 5703.
6334	026222	004767	170362		JSR	PC,WAITTX	;	WAIT FOR TRANSMISSION TO COMPLETE.
6335	026226	103144			BCC	50#	;	GO REPORT ERROR IF TX FAILED TO COMPLETE.
6336								
6337					;		;	READ THE FIRST 130 CHARACTERS FROM THE FIFO, IF ANY XON'S ARE FOUND
6338					;		;	REPORT THE ERROR, IF ANY BMP CODES ARE FOUND THEN SAVE THEM ON THE QUEUE
6339					;		;	AND ABORT THE TEST.
6340					;			
6341	026230	005267	155626		INC	ERRNBR	;	SET ERROR NUMBER TO 5704.
6342	026234	012700	000202		MOV	#130.,R0	;	PASS THE NUMBER OF CHARS TO READ.
6343	026240	004767	167002		JSR	PC,READBX	;	READ THE FIRST 130 CHARS FROM THE FIFO.
6344	026244	103135			BCC	50#	;	GO REPORT ERROR IF BMP CODE FOUND.
6345	026246	005267	155610		INC	ERRNBR	;	SET ERROR NUMBER TO 5705.
6346	026252	005701			TST	R1	;	CHECK IF AN XON WAS FOUND.
6347	026254	001125			BNE	30#	;	GO REPORT ERROR IF AN XON WAS FOUND.
6348								
6349					;		;	TRANSMIT A NULL CHARACTER (WHICH CAUSES AN XOFF TO BE GENERATED).
6350					;			
6351					;			
6352	026256	010577	153754		MOV	R5,BCSRA	;	SELECT THE LINE READY FOR TRANSMISSION.
6353	026262	112777	000000	153754	MOV	#0,BFDATA	;	TRANSMIT A NULL CHARACTER.
6354	026270	005267	155566		INC	ERRNBR	;	SET ERROR NUMBER TO 5706.
6355	026274	004767	170310		JSR	PC,WAITTX	;	WAIT FOR TX TO COMPLETE.
6356	026300	103117			BCC	50#	;	GO REPORT ERROR IF TX DID NOT COMPLETE.
6357								
6358					;		;	READ THREE CHARACTERS, TO CAUSE THE XON TO BE GENERATED.
6359					;			
6360	026302	005267	155554		INC	ERRNBR	;	SET ERROR NUMBER TO 5707.
6361	026306	012700	000003		MOV	#3,R0	;	SET THE READ COUNT TO 3.
6362	026312	004767	166730		JSR	PC,READBX	;	READ 3 CHARACTERS FROM THE FIFO.
6363	026316	103110			BCC	50#	;	GO REPORT ERROR IF FIFO EMPTY.
6364	026320	005267	155536		INC	ERRNBR	;	SET ERROR NUMBER TO 5708.
6365	026324	005701			TST	R1	;	CHECK IF AN XON WAS FOUND.
6366	026326	001102			BNE	40#	;	GO REPORT ERROR IF AN XON WAS FOUND.
6367								
6368					;		;	TRANSMIT 62 CHARACTERS TO BRACKET THE XON AND FILL THE FIFO WITH 191 CHARS.
6369					;			
6370	026330	012700	000076		MOV	#62.,R0	;	PASS LENGTH OF DATA PATTERN.
6371	026334	010501			MOV	R5,R1	;	PASS THE LINE NUMBER.
6372	026336	005267	155526		INC	ERRNBR	;	SET ERROR NUMBER TO 5709.
6373	026342	004767	167504		JSR	PC,TXDATP	;	TRANSMIT DATA PATTERN.
6374	026346	103074			BCC	50#	;	EXIT IF ERROR FOUND DURING TX.
6375								
6376					;		;	WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
6377					;		;	THE FIFO.
6378					;			
6379					;			
6380	026350	005267	155506		INC	ERRNBR	;	SET ERROR NUMBER TO 5710.
6381	026354	004767	170230		JSR	PC,WAITTX	;	WAIT FOR TX TO COMPLETE.

```

6382 026360 103067          BCC      50#           ;GO REPORT ERROR IF TX FAILED TO COMPLETE.
6383
6384                          ;*
6385                          ; READ THE FIRST 126 CHARACTERS.
6386                          ; READ THE NEXT 4 CHARACTERS AND CHECK IF THEY ARE IN THE FOLLOWING ORDER
6387                          ; NULL, XOFF, XON, NULL.
6388 026362 005267 155474      INC      ERRNBR           ;SET ERROR NUMBER TO 5711.
6389 026366 012700 000176      MOV      #126.,R0        ;SET UP READ COUNTER.
6390 026372 004767 166650      JSR      PC,READBX      ;READ THE FIRST 126 CHARS.
6391 026376 103060          BCC      50#           ;GO REPORT THE ERROR IF FIFO EMPTY.
6392 026400 005267 155456      INC      ERRNBR           ;SET ERROR NUMBER TO 5712.
6393 026404 005701          TST      R1             ;CHECK IF AN XON WAS FOUND.
6394 026406 001052          BNE      40#           ;GO REPORT ERROR IF AN XON WAS FOUND.
6395 026410 005267 155446      INC      ERRNBR           ;SET ERROR NUMBER TO 5713.
6396 026414 012701 007005      MOV      #EM5402,R1     ;PASS THE MESSAGE TO BE REPORTED.
6397 026420 016703 153614      MOV      RBUFA,R3       ;GET THE RECEIVER BUFFER ADDRESS.
6398 026424 011302          MOV      (R3),R2        ;READ THE NULL CHARACTER FROM THE FIFO.
6399 026426 120227 000000      CMPB    R2,#000        ;CHECK IF IT IS A NULL CHARACTER.
6400 026432 001040          BNE      40#           ;GO REPORT THE ERROR IF NOT THE SAME.
6401 026434 005267 155422      INC      ERRNBR           ;SET ERROR NUMBER TO 5714.
6402 026440 011302          MOV      (R3),R2        ;READ THE XOFF FROM THE FIFO.
6403 026442 120227 000023      CMPB    R2,#23         ;CHECK IF THE READ CHAR IS AN XOFF.
6404 026446 001032          BNE      40#           ;GO REPORT THE ERROR IF NOT THE SAME.
6405 026450 011302          MOV      (R3),R2        ;READ THE XON FROM THE FIFO.
6406 026452 005267 155404      INC      ERRNBR           ;SET ERROR NUMBER TO 5715.
6407 026456 120227 000021      CMPB    R2,#21         ;CHECK IF THE READ CHARACTER IS AN XON.
6408 026462 001024          BNE      40#           ;GO REPORT THE ERROR IF NOT THE SAME.
6409 026464 005267 155372      INC      ERRNBR           ;SET ERROR NUMBER TO 5716.
6410 026470 011302          MOV      (R3),R2        ;READ THE NULL CHARACTER FROM THE FIFO.
6411 026472 120227 000000      CMPB    R2,#000        ;CHECK IF IT IS A NULL CHARACTER.
6412 026476 001016          BNE      40#           ;GO REPORT THE ERROR IF NOT THE SAME.
6413
6414
6415                          ;*
6416                          ; READ THE REMAINING CHARACTERS FROM THE FIFO.
6417 026500 012700 000075      6# :    MOV      #61.,R0        ;SET UP READ COUNTER.
6418 026504 005267 155352      INC      ERRNBR           ;SET ERROR NUMBER TO 5717.
6419 026510 004767 166532      JSR      PC,READBX      ;READ THE FIRST 61 CHARS.
6420 026514 103011          BCC      50#           ;GO REPORT THE ERROR IF FIFO EMPTY.
6421 026516 005267 155340      INC      ERRNBR           ;SET ERROR NUMBER TO 5718.
6422 026522 005701          TST      R1             ;CHECK IF AN XON WAS FOUND.
6423 026524 001003          BNE      40#           ;GO REPORT ERROR IF AN XON WAS FOUND.
6424 026526 000406          BR       60#           ;EXIT THE TEST.
6425 026530 012701 007005      30# :    MOV      #EM5402,R1     ;SET UP THE MESSAGE
6426
6427 026534          40# :    ERROR                               ; "FIFO ALARM SIGNAL DEFECTIVE".
6428 026536 000402          BR       60#           ;EXIT THE TEST.
6429
6430 026540 004767 167174      50# :    JSR      PC,TSABRT      ;REPORT TEST ABORTED. ERROR # INDICATES FAULT.
6431 026544 005067 153510      60# :    CLR      CTRLCF        ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6432
6433          ENDTST
                                L10036:
                                TRAP      C#ETST

```

```

6435 .SBTTL HARDWARE TEST - BREAKB -
6436 ;*****
6437 ;* - BREAK GENERATION TEST -
6438 ;* THIS TEST VERIFIES THAT ALL SERIAL TRANSMIT LINES CAN GENERATE A BREAK
6439 ;* BY SETTING THE BRK BIT IN THE ASSOCIATED LNCTRL REGISTER.
6440 ;* USE OF THE INTERNAL LOOPBACK FEATURE OF THE DUARTS IS MADE TO MINIMISE
6441 ;* ANY EXTERNAL EFFECTS CAUSED ON THE SERIAL LINES BY THIS TEST.
6442 ;* FRAMING ERROR DETECTION IS USED TO INDICATE THE PRESENCE OF A BREAK,
6443 ;* BY SETTING THE APPROPRIATE BIT IN THE RBUF REGISTER.
6444 ;*****
6445
6446 026552          BGNTST
        026552
6447
6448 026552 012767 177777 153500      MOV    #1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
6449          000015                    TNUM -- TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
6450 026560 012767 000015 153474      MOV    #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (64)
6451 026566 012767 000001 155264      MOV    #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
6452 026574 012767 014401 155260      MOV    #6401,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
6453 026602 012767 007244 155254      MOV    #EM6401,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
6454
6455 ;*
6456 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
6457 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
6458 ; THIS SUBROUTINE REPORTS ERROR >>>> 6401 <<<<.
6459 026610 004767 164760      JSR    PC,CLRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
6460 026614 103165              BCC    60# ;EXIT TEST IF FATAL ERROR FOUND.
6461
6462 ;*
6463 ; SET UP DEVICE UNDER TEST (DUT) TO:
6464 ; DISABLE TRANSMISSION AND RECEPTION INTERRUPTS.
6465 ; DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME TO CLEAR ANY BREAKS.
6466 026616 012705 177777      MOV    #MAPLNS,R5 ;PASS ACTIVE LINE BIT MAP.
6467 026622 012700 000200      MOV    #200,R0 ;PASS INTERNAL LOOPBACK MODE.
6468 026626 004767 170016      JSR    PC,WTMLNC ;SELECT INTERNAL LOOPBACK,DISABLE DMA.
6469 026632 012704 000012      MOV    #10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
6470 026636 004767 165070      JSR    PC,DELAY ;DELAY TO ALLOW ANY BREAKS TO BE CLEARED.
6471
6472 ;*
6473 ; SET UP TRANSMISSION AN RECEPTION PARAMETERS FOR ALL LINES.
6474 ; 9600 BAUD,8 CHAR,1 STOPBIT,NO PARITY.
6475 026642 012700 156430      MOV    #156430,R0 ;SET UP BAUD RATE,ETC.
6476 026646 004767 170026      JSR    PC,WTMLPR ;SET COMMUNICATION PARAMETERS ON ALL LINES.
6477
6478 ;*
6479 ; ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
6480 026652 016705 153352      MOV    ACTLNS,R5 ;PASS ACTIVE LINE BIT MAP.
6481 026656 004767 167306      JSR    PC,TXENBL ;ENABLE TRANSMISSIONS ON ALL LINES.
6482
6483 ;*
6484 ; PURGE THE FIFO OF ANY UNWANTED CHARACTERS.
6485 ; THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6402 THRU 6404 <<<<.
6486 026662 005267 155174      INC    ERRNBR ;SET ERROR NUMBER TO 6402.
6487 026666 004767 166152      JSR    PC,PUFIFR ;PURGE FIFO.
6488 026672 103136              BCC    60# ;ABORT TEST IF FIFO WILL NOT CLEAR.
6489
6490 ;*
        ; VERIFY BREAK GENERATION ON INDIVIDUAL LINES.

```

```

6491 ; CLEAR BREAKS ON ALL LINES.
6492 ; DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME FOR ANY BREAKS TO BE CLEARED.
6493 ; SELECT LINE,SET BREAK BIT IN LNCTRL REGISTER.
6494 ; TEST FOR A CHARACTER IN THE FIFO WITH FRAME ERROR.
6495 ;
6496 026674 005002 2# : CLR R2 ;CLEAR LINE COUNTER.
6497 026676 012703 000001 MOV #1,R3 ;SET UP ACTIVE LINE BIT MASK.
6498 026702 030367 153322 4# : BIT R3,ACTLNS ;CHECK IF THIS LINE IS ACTIVE.
6499 026706 001440 BEQ #1 ;GO SELECT NEXT LINE IF THIS ONE IS INACTIVE.
6500 026710 012700 000200 MOV #200,R0 ;SET UP PARAMETER TO CLEAR BREAK BITS.
6501 026714 004767 167730 JSR PC,WTMLNC ;CLEAR BREAK BIT,RESELECT INTERNAL LOOPBACK.
6502 026720 012704 000012 MOV #10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
6503 026724 004767 165002 JSR PC,DELAY ;DELAY TO ALLOW BREAKS TO BE CLEARED.
6504 ;
6505 ; SET BREAK BIT ON SELECTED LINE.
6506 ; SET UP PARAMETERS TO TEST FOR THE FRAME ERROR BIT SET IN RBUF.
6507 ; TIME-OUT = 5 MILLI SECONDS.
6508 ; CALL ROUTINE TO CHECK FOR CONDITION FOUND.
6509 ;
6510 026730 010305 6# : MOV R3,R5 ;COPY ACTIVE LINE BIT MASK.
6511 026732 012700 000214 MOV #214,R0 ;SET BREAK,RESELECT LOOPBACK,ENABLE RECEPTION.
6512 026736 004767 167706 JSR PC,WTMLNC ;SET BREAK ON SELECTED LINE.
6513 ;
6514 ; DELAY FOR 5 MS TO ALLOW TIME FOR BREAK TO BE GENERATED AND RECEIVED.
6515 ; VERIFY RECEPTION OF A CHARACTER WITH FRAME ERROR BIT SET.
6516 ;
6517 026742 012704 000005 MOV #5,R4 ;SET DELAY VALUE TO 5 MILLI SECS.
6518 026746 004767 164760 JSR PC,DELAY ;ALLOW TIME FOR CHARACTER RECEPTION.
6519 026752 017700 153262 MOV BRBUFA,R0 ;GET CHARACTER FROM RBUF REGISTER.
6520 026756 032700 020000 BIT #BIT13,R0 ;CHECK FOR FRAME ERROR BIT.
6521 026762 001012 BNE #1 ;SKIP ERROR REPORT IF SET.
6522 026764 012701 007301 MOV #EM6402,R1 ;SELECT MESSAGE TO BE PRINTED.
6523 ;REPORT ERROR"BREAK NOT RECEIVED ON LINE #NN"
6524 026770 ERRDF 6405,EM6401,ER6401 ; >>>>> ERROR #6405 <<<<<.
        026770 104455 TRAP C#ERDF
        026772 014405 .WORD 6405
        026774 007244 .WORD EM6401
        026776 011734 .WORD ER6401
6525 ;
6526 ;
6527 ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
6528 ;
6529 027000 032767 000100 153212 BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
6530 027006 001470 BEQ #60# ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
6531 ; DURING THE SOFTWARE QUESTIONS.
6532 ;
6533 027010 006303 8# : ASL R3 ;SHIFT BIT MASK FOR NEXT LINE.
6534 027012 005202 INC R2 ;NEXT LINE
6535 027014 020227 000020 CMP R2,#NUMLNS ;CHECK FOR MAX LINE COUNT.
6536 027020 001330 BNE #1 ;IF <>,LOOP TO CHECK NEXT LINE
6537 ;
6538 ; VERIFY BREAK GENERATION ON ALL LINES SIMULTANEOUSLY.
6539 ; CLEAR BREAKS ON ALL LINES.
6540 ; DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME FOR ANY BREAKS TO BE CLEARED.
6541 ; PURGE THE FIFO.
6542 ; SET BREAK BIT IN LNCTRL REGISTERS ON ALL ACTIVE LINES.
6543 ; TEST FOR CHARACTERS IN THE FIFO WITH FRAME ERROR.
    
```

```

6544
6545 027022 012705 177777      ;
6546 027026 012700 000200      MOV    #MAPLNS,R5      ;SET UP LINE TO CLEAR BREAKS ON.
6547 027032 004767 167612      MOV    #200,R0         ;SET UP PARAMETER TO CLEAR BREAK BITS.
6548 027036 012704 000012      JSR    PC,WTWLNLC      ;CLEAR BREAK BIT,RESELECT INTERNAL LOOPBACK.
6549 027042 004767 164664      MOV    #10.,R4         ;PASS DELAY TIME OF 10 MILLI SECONDS.
6550                               JSR    PC,DELAY         ;DELAY TO ALLOW BREAKS TO BE CLEARED.
6551                               ;
6552                               ;*
6553 027046 004767 165710      ; PURGE THE FIFO OF UNWANTED CHARACTERS.
6554 027052 103044              ;
6555                               ;
6556                               ;*
6557                               ; SET UP PARAMETERS FOR SETTING THE BREAK BIT ON ALL ACTIVE LINES.
6558                               ; THEN CALL ROUTINE TO DO IT.
6559 027054 016705 153150      10#: MOV    ACTLNS,R5     ;SET UP ACTIVE LINE BIT MASK.
6560 027060 012700 000214      MOV    #214,R0         ;SET BREAK,RESELECT LOOPBACK,ENABLE RECEPTION.
6561 027064 004767 167560      JSR    PC,WTWLNLC      ;SET BREAK ON SELECTED LINES.
6562                               ;
6563                               ;*
6564                               ; DELAY FOR 10 MILLI SECONDS,TO ALLOW TIME FOR RECEPTION.
6565                               ; TEST FOR CHARACTERS IN FIFO WITH FRAME ERROR BIT SET.
6566 027070 012704 000012      ;
6567 027074 004767 164632      MOV    #10.,R4         ;SET DELAY VALUE TO 10 MILLI SECS.
6568 027100 010502              JSR    PC,DELAY         ;ALLOW TIME FOR CHARACTER RECEPTION.
6569 027102 004767 165162      MOV    R5,R2           ;COPY ACTIVE LINE BIT MAP.
6570 027106 017701 153126      JSR    PC,MAPCNT       ;COUNT THE NUMBER OF LINES AVAILABLE.
6571 027112 100011              12#: MOV    BRBUFA,R1    ;GET CHARACTER FROM RBUF REGISTER.
6572 027114 032701 020000      BPL    14#             ;BRANCH IF DATA_VALID NOT SET.
6573 027120 001406              BIT    #BIT'13,R1     ;CHECK FOR FRAME ERROR BIT.
6574 027122 000301              BEQ    14#             ;DO NOT CLR FLG FOR THIS LINE IF FRAME BIT CLR.
6575 027124 042701 177400      SWAB   R1              ;GET LINE NUMBER IN LOW BYTE.
6576 027130 004767 165106      BIC    #177400,R1     ;CLEAR EVERYTHING BUT THE LINE NUMBER.
6577 027134 040005              JSR    PC,LINBIT      ;CALC BIT MASK FROM LINE NUMBER.
6578 027136 005302              BIC    R0,R5           ;CLEAR LINE FLAG.
6579 027140 001362              14#: DEC    R2         ;DECREMENT THE LINE NUMBER COUNTER.
6580 027142 005705              BNE    12#             ;LOOP TO GET THE NEXT CHARACTER.
6581 027144 001411              TST    R5              ;CHECK IF ANY BREAKS NOT RECEIVED.
6582 027146 012701 007301      BEQ    60#             ;EXIT TEST IF ALL CLEAR.
6583                               MOV    #EM6402,R1      ;SELECT MESSAGE TO BE PRINTED.
6584 027152 104455              ;REPORT ERROR"BREAK NOT RECEIVED ON LINE #NN".
6585 027154 014406              ERRDF 6406,EM6401,ER6401; >>>> ERROR #6407 <<<<<.
6586 027156 007244              TRAP  C#ERDF
6587 027160 011734              .WORD 6406
6588 027162 000402              .WORD EM6401
6589 027174 104401              .WORD ER6401
6585 027162 000402              BR     60#             ;EXIT THE TEST.
6586
6587 027164 004767 166550      50#: JSR    PC,TSABRT   ;ABORT THE TEST.
6588 027170 005067 153064      60#: CLR    CTRLCF      ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6589 027174
6589 027174 104401              L10037: TRAP  C#ETST

```

```

6591 .SBTTL HARDWARE TEST - NORERR -
6592 ;*****
6593 ;
6594 ; NO OVERRUN ERROR TEST -
6595 ;
6596 ; THIS TEST VERIFIES THAT THE DUT WILL NOT REPORT DATA OVERRUN
6597 ; ERRORS WHEN THEY DO NOT OCCUR.
6598 ; THIS TEST PUTS 256 CHARACTERS IN THE DUT FIFO PLUS 4 IN EACH ACTIVE
6599 ; UART AND VERIFIES THAT NO OVERRUN ERRORS ARE REPORTED.
6600 ; ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
6601 ; HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
6602 ; REPORTED LATER.
6603 ;
6604 ;-----*****
6605 027176 BGNTST
6606 027176 SETPRI #PRI05 ;ALLOW LTC INTERRUPTS. T14::
        027176 012700 000240 ;
        027202 104441 ;
6607 000016 ; TRAP #PRI05,R0
6608 027204 012767 000016 153050 ; TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
6609 027212 012767 177777 153040 ; MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (66)
6610 027220 012767 000001 154632 ; MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
6611 027226 012767 014711 154626 ; MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
6612 027234 012767 007342 154622 ; MOV #6601,ERRNBR ;SET ERROR NUMBER TO 6601.
        ; MOV #EM6601,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
6613 ;
6614 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
6615 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
6616 ; THIS SUBROUTINE REPORTS ERROR >>>> 6601 <<<<<.
6617 ;
6618 027242 004767 164326 ;
6619 027246 103402 ; JSR PC,CLNRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
6620 027250 000167 000432 ; BCS .+6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
        ; JMP 601 ;EXIT THE TEST, FATAL ERROR WAS FOUND.
6621 ;
6622 ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
6623 ; INITIALIZE THE 256 BYTE DATA PATTERN.
6624 ;
6625 027254 004767 164602 ;
6626 027260 103402 ; JSR PC,FINACT ;FIND AN ACTIVE LINE.
6627 027262 000167 000420 ; BCS .+6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
6628 027266 004767 164650 ; JMP 601 ;EXIT THE TEST, FATAL ERROR WAS FOUND.
        ; JSR PC,INDATP ;INITIALISE DATA PATTERN.
6629 ;
6630 ; TRANSMIT A 265 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
6631 ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
6632 ;
6633 ;
6634 ; SET INTERNAL LOOPBACK ON THE SELECTED LINE.
6635 ; TRANSMIT THE DATA PATTERN ON THE FIRST AVAILABLE ACTIVE LINE.
6636 ;
6637 027272 005267 154564 ;
6638 027276 012700 000204 ; INC ERRNBR ;SET THE ERROR REPORT NUMBER TO 6602.
6639 027302 004767 167342 ; MOV #204,R0 ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
        ; JSR PC,WTLNC ;INITILAISE THE LINE CONTROL REGISTER.
6640 027306 012700 177670 ; MOV #177670,R0 ;PASS THE LPR CONTENTS.
6641 027312 004767 167362 ; JSR PC,WMLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
6642 027316 012704 000012 ; MOV #10.,R4 ;PASS DELAY TIME OF 10 HILLI SECONDS.
6643 027322 004767 164404 ; JSR PC,DELAY ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
6644 027326 012702 002660 ; MOV #BUFBAS,R2 ;PASS THE START OF THE DATA PATTERN TO TX.
    
```

```

6645 027332 012703 000400      MOV      #BUF MID-BUF BAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN.
6646 027336 004767 164430      JSR      PC,DODMA           ;TRANSMIT THE DATA PATTERN.
6647 027342 103157                BCC      50#                ;EXIT IF ERROR FOUND DURING DMA TX.
6648
6649                               ;*
6650                               ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
6651                               ; THE FIFO.
6652 027344 005267 154512      INC      ERRNBR             ;SET ERROR NUMBER TO 6603.
6653 027350 012701 170536      MOV      #170536,R1         ;PASS TIME-OUT VALUE OF 350 MILLI SECS.
6654 027354 016702 152656      MOV      CSRA,R2           ;PASS THE ADDRESS OF THE CSR.
6655 027360 004767 167150      JSR      PC,WAIBIS         ;WAIT FOR DMA TO COMPLETE, TX_ACTION SET.
6656 027364 103146                BCC      50#                ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
6657 027366 012704 000005      MOV      #5,R4             ;PASS DELAY OF 5 MILLI SECS.
6658 027372 004767 164334      JSR      PC,DELAY          ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
6659
6660                               ;*
6661                               ; TRANSMIT 4 CHARACTERS ON EACH ACTIVE LINE.
6662                               ;-
6663 027376 016705 152626      MOV      ACTLNS,R5         ;ALTER PARAMETERS FOR ALL ACTIVE LINES.
6664 027402 012700 000204      MOV      #204,R0           ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
6665 027406 004767 167236      JSR      PC,WTWLNCR        ;INITIALIZE THE LINE CONTROL REGISTER.
6666 027412 012700 177670      MOV      #177670,R0        ;PASS THE LPR CONTENTS.
6667 027416 004767 167256      JSR      PC,WTWLPRL        ;SET THE LPR CONTENTS TO 38.4K BAUD.
6668 027422 012704 000012      MOV      #10,R4           ;PASS DELAY TIME OF 10 MILLI SECONDS.
6669 027426 004767 164300      JSR      PC,DELAY          ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
6670
6671 027432 012702 002660      MOV      #BUF BAS,R2       ;PASS THE START OF THE DATA PATTERN TO TX.
6672 027436 012703 000004      MOV      #4,R3            ;PASS THE LENGTH OF THE DATA PATTERN.
6673 027442 005001                CLR      R1                ;CLEAR THE LINE COUNTER.
6674 027444 005267 154412      INC      ERRNBR           ;SET ERROR NUMBER TO 6604.
6675 027450 010100                2# : MOV      R1,R0
6676 027452 006300                ASI      R0                ;CALCULATE THE LINE OFFSET FROM THE LINE #.
6677 027454 036067 002344 152546  BIT      BITBL(R0),ACTLNS ;TEST FOR THIS LINE BEING ACTIVE.
6678 027462 001403                BEQ      4#                ;SKIP THE TX ON THIS LINE IF IT IS NOT ACTIVE.
6679 027464 004767 164302      JSR      PC,DODMA         ;TRANSMIT THE 5 CHAR DATA PATTERN.
6680 027470 103104                BCC      50#                ;ABORT IF ERROR FOUND DURING DMA TX.
6681 027472 005201                4# : INC      R1            ;INCREMENT THE LINE COUNTER.
6682 027474 020127 000020      CMP      R1,#NUMLNS        ;TEST FOR ALL POSSIBLE LINES HANDLED
6683 027500 002763                BLT      2#                ;LOOP IF NOT ALL LINES HANDLED.
6684
6685 027502 005267 154354      INC      ERRNBR           ;SET ERROR NUMBER TO 6605.
6686 027506 012701 170040      MOV      #170040,R1        ;PASS TIME-OUT VALUE OF 32 MILLI SECS.
6687 027512 016702 152520      MOV      CSRA,R2           ;PASS THE ADDRESS OF THE CSR.
6688 027516 004767 167012      JSR      PC,WAIBIS         ;WAIT FOR A DMA TO COMPLETE, TX_ACTION SET.
6689 027522 103067                BCC      50#                ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
6690 027524 012704 000005      MOV      #5,R4             ;PASS DELAY OF 5 MILLI SECS.
6691 027530 004767 164176      JSR      PC,DELAY          ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
6692
6693                               ;*
6694                               ; READ THE FIFO CHECKING FOR OVERRUN ERRORS. REPORT ERRORS IF FOUND.
6695                               ; ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
6696 027534 016702 152470      MOV      ACTLNS,R2
6697 027540 004767 164524      JSR      PC,MAPCNT         ;GET THE NUMBER OF ACTIVE LINES.
6698 027544 006302                ASL      R2
6699 027546 006302                ASL      R2                ;MULTIPLY NUMBER OF ACTIVE LINES BY 4.
6700 027550 012705 000400      MOV      #256,R5
6701 027554 060205                ADD      R2,R5             ;CALCULATE NUMBER OF CHARACTERS TO RX.

```

```

6702 027556 005004          CLR    R4          ;CLEAR THE CHARACTER COUNTER.
6703 027560 012767 014716 154274 6:  MOV    #6606.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
6704 027566 017702 152446          MOV    @RBUFA,R2    ;READ A CHARACTER FROM THE FIFO.
6705 027572 100036          BPL    10:         ;EXIT THE READ LOOP IF THE FIFO IS EMPTY.
6706
6707          ;+
6708          ; CHECK IF THE READ CHARACTER IS A BMP CODE.
6709          ; IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
6710          ; ABORT THE TEST.
6711 027574 004767 163674          JSR    PC,CHKBMP    ;CHECK IF CHARACTER IS A BMP CODE.
6712 027600 103002          BCC    8:         ;BRANCH IF NOT A BMP CODE.
6713 027602          ERROR          ;
6714 027602 104460          >>>> ERROR #6606 <<<<<.
6715 027604 000440          BR     60:        ;EXIT THIS TEST.
6716 027606 005267 154250          8:    INC    ERRNBR    ;SET ERROR NUMBER TO 6607.
6717 027612 005204          INC    R4          ;COUNT THIS CHARACTER.
6718 027614 020405          CMP    R4,R5       ;COMPARE # OF CHARS WITH MAX # OF CHARS.
6719 027616 003031          BGT    50:        ;ABORT TEST IF TOO MANY VALID CHARS READ.
6720 027620 032702 040000          BIT    #BIT14,R2   ;TEST THE OVERRUN BIT OF THE READ CHAR.
6721 027624 001755          BEQ    6:         ;LOOP TO READ THE NEXT CHAR IF NO ERROR.
6722 027626 005267 154230          INC    ERRNBR     ;SET ERROR NUMBER TO 6608.
6723 027632 012767 012042 154226          MOV    #ER7801,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
6724 027640 012701 007377          MOV    #EM6602,R1  ;PASS THE MESSAGE TO BE REPORTED.
6725 027644 010203          MOV    R2,R3
6726 027646 000303          SWAB  R3
6727 027650 042703 177760          BIC    #177760,R3  ;GET FAILING LINE NUMBER.
6728          ;REPORT "OVERRUN ERROR REPORTED WHEN NONE FORCED, ON LINE NN ..."
6729 027654          ERROR          ;
6730 027654 104460          >>>> ERROR #6608 <<<<<.
6731          TRAP    C#ERROR
6732
6733          ;+
6734          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
6735          ;-
6736          BIT    #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
6737          BEQ    60:        ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
6738          ;DURING THE SOFTWARE QUESTIONS.
6739          BR     6:         ;LOOP TO READ THE NEXT CHAR.
6740 027670 012767 014721 154164 10:  MOV    #6609.,ERRNBR ;SET ERROR NUMBER TO 6609.
6741 027676 020405          CMP    R4,R5       ;COMPARE NUMBER OF CHARS READ WITH EXPECTED.
6742 027700 001402          BEQ    60:        ;EXIT TEST WITHOUT ABORT IF CORRECT # OF CHARS.
6743
6744 027702 004767 166032          50:   JSR    PC,TSABRT    ;ABORT THE TEST, NON-RELATED TEST ERROR FOUND.
6745 027706 005067 152346          60:   CLR    CTRLCF      ;INDICATE THAT WE ARE NOT WITHIN A TEST.
6746 027712          ENDTST
        L10040:
        TRAP    C#ETST
    
```

```

6748 .SBTTL HARDWARE TEST - ORERR -
6749 ;*****
6750 ;
6751 ; - OVERRUN ERROR TEST -
6752 ;
6753 ; THIS TEST VERIFIES THAT THE DUT WILL REPORT DATA OVERRUN ERRORS WHEN
6754 ; THEY OCCUR.
6755 ; THIS TEST PUTS 256 CHARACTERS IN THE DUT FIFO PLUS 5 IN EACH ACTIVE
6756 ; UART AND VERIFIES THAT OVERRUN ERRORS ARE REPORTED ON ALL ACTIVE LINES.
6757 ; ANY BMP CODE FOUND WILL INVALIDATE THE TEST AND CAUSE IT TO BE ABORTED.
6758 ; HOWEVER THE BMP CODE WILL BE PLACED ON THE BMP CODE QUEUE, TO BE
6759 ; REPORTED LATER.
6760 ;
6761 ;*****
6762 BGN1ST
6763 027714 SETPRI @PRI05 ;ALLOW LTC INTERRUPTS. T15::
        027714 012700 000240
        027720 104441
        000017
6764 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
6765 027722 012767 000017 152332 MOV @TNUM,TSTNUM ;SET UP THE TEST NUMBER. (67)
6766 027730 012767 177777 152322 MOV @-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
6767 027736 012767 000001 154114 MOV @1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
6768 027744 012767 015055 154110 MOV @6701,ERRNBR ;SET ERROR NUMBER TO 6701.
6769 027752 012767 007451 154104 MOV @EM6701,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
6770
6771 ;
6772 ; RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
6773 ; CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
6774 ; THIS SUBROUTINE REPORTS ERROR >>>> 6701 <<<<.
6775 027760 004767 163610 JSR PC,CLNRST ;RESET THE DHU-11, REPORT ANY ERRORS FOUND.
6776 027764 103402 BCS .+6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
6777 027766 000167 000660 JMP 60$ ;EXIT THE TEST, FATAL ERROR WAS FOUND.
6778
6779 ;
6780 ; FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
6781 ; INITIALIZE THE 256 BYTE DATA PATTERN.
6782 027772 004767 164064 JSR PC,FINACT ;FIND AN ACTIVE LINE.
6783 027776 103402 BCS .+6 ;IF ACTIVE LINE IS FOUND, DON'T ABORT TEST.
6784 030000 000167 000646 JMP 60$ ;ABORT THE TEST, NO ACTIVE LINES WERE FOUND.
6785 030004 004767 164132 JSR PC,INDATP ;INITIALISE DATA PATTERN.
6786
6787 ;
6788 ; TRANSMIT A 265 CHARACTER DATA PATTERN USING DMA, ON A SINGLE CHANNEL
6789 ; AT 38.4K BAUD, 8 BITS PER CHARACTER, ODD PARITY, 2 STOP BITS.
6790
6791 ;
6792 ; SET INTERNAL LOOPBACK ON THE SELECTED LINE.
6793 ; TRANSMIT THE DATA PATTERN ON THE FIRST AVAILABLE ACTIVE LINE.
6794 030010 005267 154046 INC ERRNBR ;SET ERROR NUMBER TO 6702.
6795 030014 012700 000204 MOV @204,R0 ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
6796 030020 004767 166624 JSR PC,WTWLNCR ;INITILAISE THE LINE CONTROL REGISTER.
6797 030024 012700 177670 MOV @177670,R0 ;PASS THE LPR CONTENTS.
6798 030030 004767 166644 JSR PC,WTWLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
6799 030034 012704 000012 MOV @10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
6800 030040 004767 163666 JSR PC,DELAY ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
6801 030044 012702 002660 MOV @BUFBRAS,R2 ;PASS THE START OF THE DATA PATTERN TO TX.

```

```

6802 030050 012703 000400      MOV    #BUF MID-BUF BAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN.
6803 030054 004767 163712      JSR    PC,DODMA             ;TRANSMIT THE DATA PATTERN.
6804 030060 103402                BCS    .+6                 ;IF NO ERROR FOUND DURING DMA TX, DON'T ABORT.
6805 030062 000167 000560      JMP    50#                 ;ABORT TEST, ERROR FOUND DURING DMA TX.
6806
6807
6808
6809
6810 030066 005267 153770      ;+
        ; WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
        ; THE FIFO.
        ;-
6811 030072 012701 170536      INC    ERRNBR              ;SET ERROR NUMBER TO 6703.
6812 030076 016702 152134      MOV    #170536,R1         ;PASS TIME-OUT VALUE OF 350 MILLI SECS.
6813 030102 004767 166426      MOV    CSRA,R2            ;PASS THE ADDRESS OF THE CSR.
6814 030106 103402                JSR    PC,WAIBIS          ;WAIT FOR DMA TO COMPLETE, TX ACTION SET.
6815 030110 000167 000532      BCS    .+6                 ;IF NO TIME-OUT ON DMA COMPLETION, DON'T ABORT.
6816 030114 012704 000005      JMP    50#                 ;ABORT TEST, TIME-OUT ON DMA COMPLETION.
6817 030120 004767 163606      MOV    #5,R4              ;PASS DELAY OF 5 MILLI SECS.
6818
6819
6820
        ;+
        ; TRANSMIT 5 CHARACTERS ON EACH ACTIVE LINE.
        ;-
6821 030124 016705 152100      MOV    ACTLNS,R5          ;ALTER PARAMETERS FOR ALL ACTIVE LINES.
6822 030130 012700 000204      MOV    #204,R0            ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
6823 030134 004767 166510      JSR    PC,WTW LNC         ;INITIALISE THE LINE CONTROL REGISTER.
6824 030140 012700 177670      MOV    #177670,R0         ;PASS THE LPR CONTENTS.
6825 030144 004767 166530      JSR    PC,WTW LPR         ;SET THE LPR CONTENTS TO 38.4K BAUD.
6826 030150 012704 000012      MOV    #10.,R4           ;PASS DELAY TIME OF 10 MILLI SECONDS.
6827 030154 004767 163552      JSR    PC,DELAY           ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
6828
6829 030160 012702 002660      MOV    #BUF BAS,R2        ;PASS THE START OF THE DATA PATTERN TO TX.
6830 030164 012703 000005      MOV    #5,R3              ;PASS THE LENGTH OF THE DATA PATTERN.
6831 030170 005001                CLR    R1                  ;CLEAR THE LINE COUNTER.
6832 030172 005267 153664      INC    ERRNBR              ;SET ERROR NUMBER TO 6704.
6833 030176 010100                2# : MOV    R1,R0
6834 030200 006300                ASL    R0                  ;CALCULATE LINE OFFSET FROM THE LINE #.
6835 030202 036067 002344 152020 BIT    BITTBL(R0),ACTLNS ;TEST FOR THIS LINE BEING ACTIVE.
6836 030210 001405                BEQ    4#                  ;SKIP THE TX ON THIS LINE IF IT IS NOT ACTIVE.
6837 030212 004767 163554      JSR    PC,DODMA           ;TRANSMIT THE 5 CHAR DATA PATTERN.
6838 030216 103402                BCS    .+6                 ;IF NO TIME-OUT ON DMA COMPLETION, DON'T ABORT.
6839 030220 000167 000422      JMP    50#                 ;ABORT TEST, TIME-OUT ON DMA COMPLETION.
6840 030224 005201                4# : INC    R1              ;INCREMENT THE LINE NUMBER COUNTER.
6841 030226 020127 000020      CMP    R1,#NUM LNS        ;TEST FOR ALL POSSIBLE LINES HANDLED
6842 030232 002761                BLT    2#                  ;LOOP IF NOT ALL LINES HANDLED.
6843
6844 030234 005267 153622      INC    ERRNBR              ;SET ERROR NUMBER TO 6705.
6845 030240 012701 170040      MOV    #170040,R1         ;PASS TIME-OUT VALUE OF 32 MILLI SECS.
6846 030244 016702 151766      MOV    CSRA,R2            ;PASS THE ADDRESS OF THE CSR.
6847 030250 004767 166260      JSR    PC,WAIBIS          ;WAIT FOR A DMA TO COMPLETE, TX ACTION SET.
6848 030254 103174                BCC    50#                 ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
6849 030256 012704 000005      MOV    #5,R4              ;PASS DELAY OF 5 MILLI SECS.
6850 030262 004767 163444      JSR    PC,DELAY           ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
6851
6852
6853
        ;+
        ; READ 256 CHARS FROM THE FIFO CHECKING FOR BMP CODES.
        ; ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
        ;-
6854
6855 030266 012704 000400      MOV    #256.,R4           ;SET UP THE CHARACTER COUNTER.
6856 030272 012767 015062 153562 6# : MOV    #6706.,ERRNBR      ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
6857 030300 017702 151734      MOV    BRBUFA,R2          ;READ A CHARACTER FROM THE FIFO.
6858 030304 100160                BPL    50#                 ;ABORT THE TEST IF DATA.VALID IS CLEAR.

```

```

6859 030306 005267 153550          INC    ERRNBR          ;SET ERROR NUMBER TO 6707.
6860 030312 004767 163156          JSR    PC,CHKBMP      ;CHECK IF CHARACTER IS A BMP CODE.
6861 030316 103551                   BCS    24$            ;REPORT ERROR AND ABORT TEST IF A BMP CODE.
6862 030320 005304                   DEC    R4              ;COUNT THIS CHARACTER.
6863 030322 001363                   BNE    6$              ;LOOP IF NOT 256 CHARS READ FROM FIFO.
6864
6865          ;*
6866          ; READ THE REMAINING AND VERIFY 1 OVERRUN PLUS 1 CHAR FROM EACH LINE.
6867          ;-
6867 030324 005004                   CLR    R4              ;CLEAR THE OVERRUN ERROR FLAGS.
6868 030326 012700 003720          MOV    @RXCNTB,R0
6869 030332 004767 163260          JSR    PC,CLR16W      ;CLEAR RX CHAR COUNT TABLE.
6870 030336 012767 015064 153516 8$:  MOV    #6708.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
6871 030344 017702 151670          MOV    @RBUFA,R2      ;READ A CHARACTER FROM THE FIFO.
6872 030350 100047                   BPL    14$            ;GO ANALYZE THE RESULTS IF ALL CHARS READ.
6873 030352 004767 163116          JSR    PC,CHKBMP      ;CHECK IF CHAR IS A BMP CODE.
6874 030356 103531                   BCS    24$            ;REPORT ERROR AND ABORT TEST IF A BMP CODE.
6875 030360 005267 153476          INC    ERRNBR          ;SET ERROR NUMBER TO 6709.
6876 030364 010200                   MOV    R2,R0
6877 030366 000300                   SWAB   R0
6878 030370 042700 177760          BIC    #177760,R0     ;CALCULATE THE LINE NUMBER OF THE CHAR.
6879 030374 006300                   ASL    R0              ;FORM WORD TABLE OFFSET FOR TABLE ACCESS.
6880 030376 042702 007400          BIC    #7400,R2       ;REMOVE LINE NUMBER FROM THE READ CHAR.
6881 030402 036067 002344 151620    BIT    BITTBL(R0),ACTLNS ;TEST FOR ACTIVE LINE.
6882 030410 001516                   BEQ    50$            ;ABORT TEST IF FOR INACTIVE LINE.
6883 030412 005267 153444          INC    ERRNBR          ;SET ERROR NUMBER TO 6710.
6884 030416 005760 003720          TST   RXCNTB(R0)      ;CHECK THE RX CHAR COUNTER FOR THIS LINE.
6885 030422 001006                   BNE    10$           ;IS THIS FIRST CHAR ON LINE?
6886 030424 020227 140000          CMP    R2,#140000     ;YES, TEST FOR NULL CHAR WITH OVERRUN.
6887 030430 001414                   BEQ    12$           ;IS CHAR A NULL?
6888 030432 056004 002344          BIS    BITTBL(R0),R4  ;NO, SET THE OVERRUN BIT ERROR FLAG FOR LINE.
6889 030436 000411                   BR     12$           ;GO COUNT THE CHAR AND CONTINUE.
6890 030440 026027 003720 000004 10$:  CMP    RXCNTB(R0),#4
6891 030446 002077                   BGE    50$           ;5TH CHAR ON THIS LINE? YES, ABORT.
6892 030450 032702 040000          BIT    #BIT14,R2      ;NO, CHECK OVERRUN BIT.
6893 030454 001402                   BEQ    12$           ;IS OVERRUN BIT CLEAR? YES, GO COUNT CHAR.
6894 030456 056004 002344          BIS    BITTBL(R0),R4  ;NO, SET THE OVERRUN BIT ERROR FLAG FOR LINE.
6895 030462 005260 003720 12$:  INC    RXCNTB(R0)      ;COUNT THIS CHARACTER.
6896 030466 000723                   BR     8$              ;LOOP UNTIL ALL CHARS ARE READ FROM FIFO.
6897
6898          ;*
6899          ; TEST FOR ABORT CONDITIONS. ONLY NONE ABORT CONDITIONS ARE:
6900          ; 1) 2 CHARS RXED ON A LINE AND NO OVERRUN ERROR BIT FAILURE DETECTED.
6901          ; 2) 2 TO 4 CHARS RXED ON A LINE AND AN OVERRUN BIT FAILURE DETECTED.
6902 030470 005001                   ;-
6903 030472 012767 015067 153362 14$:  CLR    R1              ;INITIALIZE LINE LOOP, CLEAR LINE OFFSET.
6904 030500 036167 002344 151522 16$:  MOV    #6711.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
6905 030506 001415                   BIT    BITTBL(R1),ACTLNS
6906 030510 026127 003720 000002    BEQ    18$            ;LINE ACTIVE? NO, NEXT LINE.
6907 030516 002453                   CMP    RXCNTB(R1),#2  ;YES.
6908 030520 036104 002344          BLT    50$           ;FEWER THAN 2 CHARS RXED? YES, ABORT.
6909 030524 001006                   BIT    BITTBL(R1),R4  ;NO.
6910 030526 005267 153330          BNE    18$           ;OVERRUN BIT ERROR FLAG SET? YES, NEXT LINE.
6911 030532 026127 003720 000002    INC    ERRNBR          ;SET LINE NUMBER TO 6712.
6912 030540 001042                   CMP    RXCNTB(R1),#2
6913 030542 062701 000002 18$:  BNE    50$           ;NOT 2 CHARS RXED? YES, ABORT. NO, NEXT LINE.
6914 030546 020127 000040          ADD    #2,R1          ;SET LINE OFFSET TO THE NEXT LINE.
6915 030552 002747                   CMP    R1,#NUMLNS+2
6915 030552 002747                   BLT    16$           ;ALL LINES DONE? NO, LOOP. YES, CONTINUE.
    
```

```

6916
6917
6918
6919 030554 012767 015071 153300
6920 030562 005001
6921 030564 010102
6922 030566 036104 002344
6923 030572 001415
6924 030574 010103
6925 030576 006203
6926 030600 012767 012042 153260
6927 030606 012701 007503
6928
6929 030612
030612 104460
6930
6931
6932
6933
6934 030614 032767 000100 151376
6935 030622 001413
6936
6937
6938 030624 010201
6939 030626 046104 002344
6940 030632 001407
6941 030634 062701 000002
6942 030640 000751
6943
6944 030642
6945 030642
030642 104460
6946 030644 000402
6947
6948 030646 004767 165066
6949 030652 005067 151402
6950
6951 030656
030656
030656 104401

;
; CHECK FOR OVERRUN ERROR BIT FAILURES, PRINT ERROR MESSAGE IF FOUND.
;
;
MOV #6713,,ERRNBR ;SET UP ERROR NUMBER.
CLR R1 ;INITIALIZE LOOP, CLEAR LINE OFFSET.
20: MOV R1,R2 ;COPY THE LINE OFFSET.
BIT BITTBL(R1),R4 ;OVERRUN BIT FAILURE FLAGS ARE IN R4.
BEQ 22: ;ERROR FLAG CLEAR? YES, NEXT LINE.
MOV R1,R3
ASR R3 ;CALCULATE LINE NUMBER FROM LINE OFFSET.
MOV #ER7801,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
MOV #EM6702,R1 ;PASS THE MESSAGE TO BE REPORTED.
;REPORT "OVERRUN ERROR NOT REPORTED CORRECTLY WHEN FORCED, ON LINE NN ..."
ERROR ; >>>> ERROR #6713 <<<<<.
; TRAP C#ERROR

;
; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;
;
BIT #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
BEQ 60: ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
;DURING THE SOFTWARE QUESTIONS.

22: MOV R2,R1 ;RESTORE THE LINE OFFSET THAT WAS DESTROYED.
BIC BITTBL(R1),R4 ;CLEAR THE LINE ERROR FLAG WE JUST HANDLED.
BEQ 60: ;ALL FAILURE BITS HANDLED? YES, EXIT TEST.
ADD #2,R1 ;NO, INCREMENT THE LINE OFFSET.
BR 20: ;LOOP TO HANDLE THE NEXT LINE.

24: ;REPORT "BMP CODE FOUND IN FIFO, TEST INVAILDATED."
ERROR ; >>>> ERROR <<<<<.
; TRAP C#ERROR

BR 60: ;EXIT THIS TEST.

50: JSR PC,TSABRT ;ABORT THE TEST. ERROR # INDICATES FAULT TYPE.
60: CLR CTRLCF ;INDICATE THAT WE ARE NOT WITHIN A TEST.

ENDTST

L1C041:
TRAP C#ETST
    
```

```

6953
6954
6955
6956
6957
6958
6959
6960
6961
6962
6963
6964
6965 030660
      030660
6966
6967
6968
6969 030660 032767 000002 151344
6970 030666 001002
6971 030670 000167 000504
6972 030674
      030674 012700 000240
      030700 104441
6973      000020
6974 030702 012767 000020 151352
6975 030710 012767 177777 151342
6976 030716 012767 000001 153134
6977 030724 012767 017171 153130
6978 030732 012767 007560 153124
6979
6980
6981
6982
6983
6984 030740 004767 162630
6985 030744 103402
6986 030746 000167 000426
6987
6988
6989
6990 030752 004767 162162
6991
6992
6993
6994
6995
6996
6997 030756 005003
6998 030760 010300
6999 030762 006300
7000 030764 036067 002344 151236
7001 030772 001471
7002
7003
7004
7005 030774 005000
7006 030776 012705 177777

```

```

.SBTTL  HARDWARE TEST          - DTRMCS -
;*****
;*          - DATA TERMINAL READY MODEM CONTROL SIGNAL TEST -
;*
;*      THIS TEST VERIFIES THAT THE DTR MODEM CONTROL SIGNAL IS WORKING
;*      CORRECTLY.  IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
;*      LOOPBACK IS SPECIFIED.  THIS TEST USES THE LOOPED BACK SIGNALS RI
;*      AND DSR TO TEST THE DTR SIGNAL.  THIS TEST IS PERFORMED ON ALL
;*      ACTIVE LINES.
;*****
      BGNTST
      T16::
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
;
;      BIT      #BIT1,LOPBACK      ;CHECK TYPE OF LOOPBACK MODE SELECTED.
;      BNE      2#
;      JMP      60#                ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
2#:   SETPRI   #PRI05              ;ALLOW LTC INTERRUPTS.
;
;      MOV      #PRI05,R0
;      TRAP    C#SPRI
;      TNUM == TNUM + 1          ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
;      MOV     #TNUM,TSTNUM      ;SET UP THE TEST NUMBER. (78)
;      MOV     #-1,CTRLCF        ;INDICATE THAT WE ARE IN A TEST.
;      MOV     #1,ERRTYP         ;SET ERROR TYPE IN ERROR TABLE.
;      MOV     #7801,ERRNBR      ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
;      MOV     #EM7801,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 >>>> 7801 <<<<<.
;
;      JSR     PC,CLRST          ;RESET THE DUT.
;      BCS     4#
;      JMP     60#              ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;
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 AND DSR SIGNALS.
; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
;
;      CLR     R3                ;CLEAR THE LINE COUNTER.
6#:   MOV     R3,R0
;      ASL     R0
;      BIT     BITBL(R0),ACTLNS
;      BEQ    12#              ;DON'T TEST IF NOT ACTIVE LINE.
;
; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
;
;      CLR     R0
;      MOV     #MAPLNS,R5      ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
;                               ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.

```

```

7007 031002 004767 165642          JSR    PC,WTWLNLC      ;CLEAR ALL THE DUT DTR BITS.
7008 031006 012704 000074          MOV    #60.,R4
7009 031012 004767 162714          JSR    PC,DELAY        ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
7010
7011          ;+
7012          ; CHECK THAT AT LEAST ONE OF ASSOCIATED DSR OR RI IS CLEAR AND RECORD STATES.
7013 031016 116304 004020          ;-
7014 031022 010477 151210          MOV    TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
7015 031026 017705 151212          MOV    R4,BCSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7016 031032 012700 120000          MOV    #FSLSA,R5      ;GET THE STATE OF THE ASSOCIATED DSR, RI BITS.
7017 031036 040500                    MOV    #BIT15!BIT13,R0
7018 031040 001431                    BIC    R5,R0           ;CHECK FOR BOTH DSR AND RI SET.
7019          BEQ    10#    ;GO REPORT DTR IS BAD IF BOTH ARE SET.
7020          ;+
7021          ; SET THE DTR FOR THE SELECTED LINE AND WAIT FOR EITHER DSR OR RI TO SET.
7022 031042 010377 151170          ;-
7023 031046 052777 001000 151172          MOV    R3,BCSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
7024 031054 012701 150074          BIS    #BIT9,ALNCTRA   ;SET THE SELECTED LINE DTR.
7025 031060 032705 100000          MOV    #150074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR RI TO SET.
7026 031064 001002                    BIT    #BIT15,R5       ;CHECK PREVIOUS STATE OF DSR BIT.
7027 031066 012701 170074          BNE    8#             ;GO USE RI IF DSR BIT WAS NOT CLEAR.
7028 031072 016702 151146          MOV    #170074,R1      ;SPECIFY TO WAIT UP TO 60 MS FOR DSR SET.
7029 031076 010477 151134          8#:    MOV    FLSA,R2     ;SPECIFY TO LOOK IN STAT REG FOR BIT TO SET.
7030 031102 004767 165426          MOV    R4,BCSRA        ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7031 031106 103423                    JSR    PC,WAIBIS       ;WAIT UP TO 60 MS FOR SIGNAL TO GO SET.
7032 031110 017700 151130          BCS    12#           ;SELECT NEXT LINE AND LOOP IF SIGNAL IS SET.
7033 031114 042700 057777          MOV    #FSLSA,R0       ;GET THE STATUS REGISTER CONTENTS.
7034 031120 040500                    BIC    #57777,R0       ;REMOVE ALL BUT THE DSR AND RI BITS.
7035 031122 001015                    BIC    R5,R0           ;TEST FOR SIGNAL ONCE CLEAR, BUT NOW SET.
7036 031124                    BNE    12#           ;GO LOOP IF SIGNAL HAS GONE FROM CLR TO SET.
7037 031124 012767 017172 152730          10#:  ;REPORT DTR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7038 031132 012767 012042 152726          MOV    #7802.,ERRNBR   ;SELECT THE ERROR NUMBER.
7039 031140 012701 007622          MOV    #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
7040 031144                    MOV    #EM7802,R1      ;SELECT THE ERROR MESSAGE.
7041          ERROR
7042          TRAP    C#ERROR
7043          ;+
7044          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7045 031146 032767 000100 151044          ;-
7046 031154 001511                    BIT    #BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
7047          BEQ    60#    ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7048          ;DURING THE SOFTWARE QUESTIONS.
7049 031156 005203                    12#:  INC    R3           ;SELECT THE NEXT LINE NUMBER.
7050 031160 020327 000020                    CMP    R3,#NULNLS     ;TEST FOR ALL LINES DONE.
7051          BLT    6#    ;LOOP IF NOT ALL LINES DONE.
7052          ;+
7053          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7054          ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7055          ; A RESPONSE ON THE ASSOCIATED RI AND DSR SIGNALS.
7056          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7057 031166 005003                    ;-
7058 031170 010300                    CLR    R3             ;CLEAR THE LINE COUNTER.
7059 031172 006300                    14#:  MOV    R3,R0
7060 031174 036067 002344 151026          ASL    R0
7061 031202 001472                    BIT    BITTBL(R0),ACTLNS
7062          BEQ    20#    ;DON'T TEST IF NOT ACTIVE LINE.

```

```

7063      ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
7064      ;-
7065 031204 012700 001000      MOV    #BIT9,R0      ;SPECIFY THAT DTR BITS ARE TO BE SET.
7066 031210 012705 177777      MOV    #MAPLNS,R5   ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
7067 031214 004767 165430      JSR    PC,WTWLNLC   ;SET ALL THE DUT DTR BITS.
7068 031220 012704 000074      MOV    #60.,R4
7069 031224 004767 162502      JSR    PC,DELAY     ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
7070
7071      ;*
7072      ; CHECK THAT AT LEAST ONE OF ASSOCIATED DSR OR RI IS SET AND RECORD STATES.
7073      ;-
7073 031230 116304 004020      MOVB   TXRLNB(R3),R4 ;GET THE ASSOCIATED LINE NUMBER.
7074 031234 010477 150776      MOV    R4,BCSRA    ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7075 031240 017705 151000      MOV    #FSLSA,R5   ;GET THE STATE OF THE ASSOCIATED DSR, RI BITS.
7076 031244 010500
7077 031246 042700 057777      MOV    R5,R0
7078 031252 001431      BIC    #57777,R0   ;CHECK FOR BOTH DSR AND RI CLEAR.
7079      BEQ    18#     ;GO REPORT DTR IS BAD IF BOTH ARE CLEAR.
7080      ;*
7081      ; CLEAR THE DTR FOR THE SELECTED LINE AND WAIT FOR EITHER DSR OR RI TO CLEAR.
7082      ;-
7082 031254 010377 150756      MOV    R3,BCSRA    ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
7083 031260 042777 001000 150760 BIC    #BIT9,BLNCTRA ;CLEAR THE SELECTED LINE DTR.
7084 031266 012701 150074      MOV    #150074,R1  ;SPECIFY TO WAIT UP TO 60 MS FOR RI TO CLEAR.
7085 031272 032705 100000      BIT    #BIT15,R5   ;CHECK PREVIOUS STATE OF DSR BIT.
7086 031276 001402      BEQ    16#     ;GO USE RI IF DSR BIT WAS NOT SET.
7087 031300 012701 170074      MOV    #170074,R1  ;SPECIFY TO WAIT UP TO 60 MS FOR DSR CLEAR.
7088 031304 016702 150734 16# :    MOV    FLSA,R2    ;SPECIFY TO LOOK IN STAT REG FOR BIT TO CLR.
7089 031310 010477 150722      MOV    R4,BCSRA    ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7090 031314 004767 165140      JSR    PC,WAIBIC   ;WAIT UP TO 60 MS FOR SIGNAL TO GO CLEAR.
7091 031320 103423      BCS    20#     ;SELECT NEXT LINE AND LOOP IF SIGNAL IS CLEAR.
7092 031322 017700 150716      MOV    #FSLSA,R0   ;GET THE STATUS REGISTER CONTENTS.
7093 031326 042705 057777      BIC    #57777,R5
7094 031332 040005      BIC    R0,R5
7095 031334 001015      BNE    20#     ;TEST FOR SIGNAL ONCE SET, BUT NOW CLEAR.
7096 031336      ;GO LOOP IF SIGNAL HAS GONE FROM SET TO CLR.
7097 031336 012767 017173 152516 18# : ;REPORT DTR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7098 031344 012767 012042 152514      MOV    #7803.,ERRNBR ;SELECT THE ERROR NUMBER.
7099 031352 012701 007622      MOV    #ER7801,ERRBLK ;SELECT THE ERROR PRINT ROUTINE.
7100 031356      MOV    #EM7802,R1  ;SELECT THE ERROR MESSAGE.
7101      TRAP    C#ERROR
7102
7103      ;*
7104      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7105      ;-
7104 031360 032767 000100 150632      BIT    #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
7105 031366 001404      BEQ    60#     ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7106      ;DURING THE SOFTWARE QUESTIONS.
7107
7108 031370 005203 20# :    INC    R3        ;SELECT THE NEXT LINE NUMBER.
7109 031372 020327 000020      CMP    R3,#NUMLNS  ;TEST FOR ALL LINES DONE.
7110 031376 002674      BLT    14#     ;LOOP IF NOT ALL LINES DONE.
7111
7112 031400 005067 150654 60# :    CLR    CTRLCF    ;INDICATE THAT WE ARE NOT WITHIN A TEST.
7113 031404      SETPRI #PRI07   ;DISABLE ALL INTERRUPTS.
7114      MOV    #PRI07,R0
7115 031412      TRAP    C#SPRI
7115 031412      ENDTST
    
```

L10042:

G14

031412 104401

TRAP C#ETST

```

7117 .SBTTL HARDWARE TEST - RTSMCS -
7118 ;*****
7119 ;* - REQUEST TO SEND MODEM CONTROL SIGNAL TEST -
7120 ;*
7121 ;* THIS TEST VERIFIES THAT THE RTS MODEM CONTROL SIGNAL IS WORKING
7122 ;* CORRECTLY. IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
7123 ;* LOOPBACK IS SPECIFIED. THIS TEST USES THE LOOPED BACK SIGNALS CTS
7124 ;* AND DCD TO TEST THE RTS SIGNAL. THIS TEST IS PERFORMED ON ALL
7125 ;* ACTIVE LINES.
7126 ;*
7127 ;*****
7128 ;-----
7129 031414 BGNTST
031414
7130 T17::
7131 ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
7132 ;-
7133 031414 032767 000002 150610 BIT #BIT1,LOPBCK ;CHECK TYPE OF LOOPBACK MODE SELECTED.
7134 031422 001002 BNE 1#
7135 031424 000167 000504 JMP 60# ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
7136 031430 1# SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
031430 012700 000240 MOV #PRI05,R0
031434 104441 TRAP C#SPRI
000021
7137 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
7138 031436 012767 000021 150616 MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (79)
7139 031444 012767 177777 150606 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
7140 031452 012767 000001 152400 MOV #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
7141 031460 012767 017335 152374 MOV #7901.,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
7142 031466 012767 007653 152370 MOV #EM7901,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
7143 ;
7144 ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
7145 ; CLEAR TX AND RX INTERRUPT ENABLE BITS.
7146 ; THIS SUBROUTINE REPORTS ERROR >>>> 7901 <<<<.
7147 ;-
7148 031474 004767 162074 JSR PC,CLNRST ;RESET THE DUT.
7149 031500 103402 BCS 3#
7150 031502 000167 000426 JMP 60# ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
7151 ;
7152 ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
7153 ;-
7154 031506 004767 161426 JSR PC,ASLNTL ;SET UP THE ASSOCIATED LINE TABLES.
7155 ;
7156 ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7157 ; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
7158 ; A RESPONSE ON THE ASSOCIATED CTS AND DCD SIGNALS.
7159 ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7160 ;-
7161 031512 005003 CLR R3 ;CLEAR THE LINE COUNTER.
7162 031514 010300 MOV R3,R0
7163 031516 006300 ASL R0
7164 031520 036067 002344 150502 BIT BITTBL(R0),ACTLNS
7165 031526 001471 BEQ 8# ;DON'T TEST IF NOT ACTIVE LINE.
7166 ;
7167 ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
7168 ;-
7169 031530 005000 CLR R0 ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
7170 031532 012705 177777 MOV #MAPLNS,R5 ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.

```

```

7171 031536 004767 165106          JSR   PC,WTLNLC      ;CLEAR ALL THE DUT RTS BITS.
7172 031542 012704 000074          MOV   #60.,R4
7173 031546 004767 162160          JSR   PC,DELAY      ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
7174
7175      ; CHECK THAT AT LEAST ONE OF ASSOCIATED DCD OR CTS IS CLEAR AND RECORD STATES.
7176      ;-
7177 031552 116304 004020          MOVB  TXRLNB(R3),R4  ;GET THE ASSOCIATED LINE NUMBER.
7178 031556 010477 150454          MOV   R4,BCSRA     ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7179 031562 017705 150456          MOV   #FSLSA,R5    ;GET THE STATE OF THE ASSOCIATED DCD, CTS BITS.
7180 031566 012700 014000          MOV   #BIT12!BIT11,R0
7181 031572 040500                    BIC   R5,R0        ;CHECK FOR BOTH DCD AND CTS SET.
7182 031574 001431                    BEQ   6#          ;GO REPORT RTS IS BAD IF BOTH ARE SET.
7183
7184      ; SET THE RTS FOR THE SELECTED LINE AND WAIT FOR EITHER DCD OR CTS TO SET.
7185      ;-
7186 031576 010377 150434          MOV   R3,BCSRA     ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
7187 031602 052777 010000 150436  BIS   #BIT12,ALNCTRA ;SET THE SELECTED LINE RTS.
7188 031610 012701 130074          MOV   #130074,R1   ;SPECIFY TO WAIT UP TO 60 MS FOR CTS TO SET.
7189 031614 032705 010000          BIT   #BIT12,R5    ;CHECK PREVIOUS STATE OF DCD BIT.
7190 031620 001002                    BNE   4#          ;GO USE CTS IF DCD BIT WAS NOT CLEAR.
7191 031622 012701 140074          MOV   #140074,R1   ;SPECIFY TO WAIT UP TO 60 MS FOR DCD SET.
7192 031626 016702 150412 4# :   MOV   FLSA,R2     ;SPECIFY TO LOOK IN STAT REG FOR BIT TO SET.
7193 031632 010477 150400          MOV   R4,BCSRA     ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
7194 031636 004767 164672          JSR   PC,WAIBIS    ;WAIT UP TO 60 MS FOR SIGNAL TO GO SET.
7195 031642 103423                    BCS   8#          ;SELECT NEXT LINE AND LOOP IF SIGNAL IS SET.
7196 031644 017700 150374          MOV   #FSLSA,R0    ;GET THE STATUS REGISTER CONTENTS.
7197 031650 042700 163777          BIC   #163777,R0   ;REMOVE ALL BUT THE DCD AND CTS BITS.
7198 031654 040500                    BIC   R5,R0        ;TEST FOR SIGNAL ONCE CLEAR, BUT NOW SET.
7199 031656 001015                    BNE   8#          ;GO LOOP IF SIGNAL HAS GONE FROM CLR TO SET.
7200 031660          6# : ;REPORT RTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7201 031660 012767 017336 152174    MOV   #7902.,ERRNBR ;SELECT THE ERROR NUMBER.
7202 031666 012767 012042 152172    MOV   #ER7801,ERRBLK ;SELECT THE ERROR PRINT ROUTINE.
7203 031674 012701 007715          MOV   #EM7902,R1   ;SELECT THE ERROR MESSAGE.
7204 031700          ERROR                               ;
          >>>>> ERROR <<<<<<.
7205          TRAP      C#ERROR
7206
7207      ;*
7208      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7209      ;-
7209 031702 032767 000100 150310    BIT   #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
7210 031710 001511                    BEQ   60#         ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7211
7212
7213 031712 005203          8# :   INC   R3          ;SELECT THE NEXT LINE NUMBER.
7214 031714 020327 000020          CMP   R3,#NUMLNS   ;TEST FOR ALL LINES DONE.
7215 031720 002675          BLT   2#          ;LOOP IF NOT ALL LINES DONE.
7216
7217      ;*
7218      ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7219      ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7220      ; A RESPONSE ON THE ASSOCIATED CTS AND DCD SIGNALS.
7221      ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7221      ;-
7222 031722 005003          CLR   R3          ;CLEAR THE LINE COUNTER.
7223 031724 010300 10# :   MOV   R3,R0
7224 031726 006300          ASL   R0
7225 031730 036067 002344 150272    BIT   BITTBL(R0),ACTLNS
7226 031736 001472          BEQ   16#         ;DON'T TEST IF NOT ACTIVE LINE.

```

```

7227
7228
7229
7230 031740 012700 010000
7231 031744 012705 177777
7232 031750 004767 164674
7233 031754 012704 000074
7234 031760 004767 161746
7235
7236
7237
7238 031764 116304 004020
7239 031770 010477 150242
7240 031774 017705 150244
7241 032000 010500
7242 032002 042700 163777
7243 032006 001431
7244
7245
7246
7247 032010 010377 150222
7248 032014 042777 010000 150224
7249 032022 012701 130074
7250 032026 032705 010000
7251 032032 001402
7252 032034 012701 140074
7253 032040 016702 150200 120:
7254 032044 010477 150166
7255 032050 004767 164404
7256 032054 103423
7257 032056 017700 150162
7258 032062 042705 163777
7259 032066 040005
7260 032070 001015
7261 032072
7262 032072 012767 017337 151762
7263 032100 012767 012042 151760
7264 032106 012701 007715
7265 032112
       032112 104460
7266
7267
7268
7269
7270 032114 032767 000100 150076
7271 032122 001404
7272
7273
7274 032124 005203 160:
7275 032126 020327 000020
7276 032132 002674
7277
7278 032134 005067 150120 600:
7279 032140
       032140 012700 000340
       032144 104441
7280

```

```

;*
; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
;-
    MOV    #BIT12,R0      ;SPECIFY THAT RTS BITS ARE TO BE SET.
    MOV    #MAPLNS,R5     ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
    JSR    PC,WTWLCNC     ;SET ALL THE DUT RTS BITS.
    MOV    #60.,R4
    JSR    PC,DELAY       ;DELAY FOR 60 MS TO ALLOW SIGNALS TO SETTLE.
;*
; CHECK THAT AT LEAST ONE OF ASSOCIATED DCD OR CTS IS SET AND RECORD STATES.
;-
    MOV    TXRLNB(R3),R4  ;GET THE ASSOCIATED LINE NUMBER.
    MOV    R4,@CSRA       ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
    MOV    @FSLSA,R5      ;GET THE STATE OF THE ASSOCIATED DCD, CTS BITS.
    MOV    R5,R0
    BIC    #163777,R0     ;CHECK FOR BOTH DCD AND CTS CLEAR.
    BEQ    140            ;GO REPORT RTS IS BAD IF BOTH ARE CLEAR.
;*
; CLEAR THE RTS FOR THE SELECTED LINE AND WAIT FOR EITHER DCD OR CTS TO CLEAR.
;-
    MOV    R3,@CSRA       ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
    BIC    #BIT12,@LNCTRA ;CLEAR THE SELECTED LINE RTS.
    MOV    #130074,R1     ;SPECIFY TO WAIT UP TO 60 MS FOR CTS TO CLEAR.
    BIT    #BIT12,R5      ;CHECK PREVIOUS STATE OF DCD BIT.
    BEQ    120            ;GO USE CTS IF DCD BIT WAS NOT SET.
    MOV    #140074,R1     ;SPECIFY TO WAIT UP TO 60 MS FOR DCD CLEAR.
120:   MOV    FSLSA,R2      ;SPECIFY TO LOOK IN STAT REG FOR BIT TO CLR.
    MOV    R4,@CSRA       ;SELECT ASSOCIATED LINE IND.ADR.REG FIELD.
    JSR    PC,WAIBIC      ;WAIT UP TO 60 MS FOR SIGNAL TO GO CLEAR.
    BCS    160            ;SELECT NEXT LINE AND LOOP IF SIGNAL IS CLEAR.
    MOV    @FSLSA,R0      ;GET THE STATUS REGISTER CONTENTS.
    BIC    #163777,R5
    BIC    R0,R5
    BNE    160            ;TEST FOR SIGNAL ONCE SET, BUT NOW CLEAR.
140:   ;REPORT RTS MODEM CONTROL ;GO LOOP IF SIGNAL HAS GONE FROM SET TO CLR.
       ;SIGNAL DEFECTIVE ON LINE NN.
    MOV    #7903.,ERRNBR  ;SELECT THE ERROR NUMBER.
    MOV    #ERR7801,ERRBLK ;SELECT THE ERROR PRINT ROUTINE.
    MOV    #EM7902,R1     ;SELECT THE ERROR MESSAGE.
    ERROR                                ;
                                     >>>>> ERROR <<<<<<
                                     TRAP    C#ERROR

```

```

;*
; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
;-
    BIT    #BIT06,CPTION  ;EXIT WITH TEST FAILURE MESSAGE IF
    BEQ    600            ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
                                     ;DURING THE SOFTWARE QUESTIONS.
160:   INC    R3           ;SELECT THE NEXT LINE NUMBER.
    CMP    R3,@NUMLNS     ;TEST FOR ALL LINES DONE.
    BLT    100            ;LOOP IF NOT ALL LINES DONE.
600:   CLR    CTRLCF      ;INDICATE THAT WE ARE NOT WITHIN A TEST.
    SETPRI #PRI07         ;DISABLE ALL INTERRUPTS.
                                     MOV    #PRI07,R0
                                     TRAP    C#SPRI

```

K14

7281 032146
032146
032146 104401

ENDTST

L10043: TRAP C#ETST

```

7283 .SBTTL HARDWARE TEST - DSRMS -
7284 ;*****
7285 ; - DATA SET READY MODEM SIGNAL TEST -
7286 ;
7287 ; THIS TEST VERIFIES THAT THE DSR MODEM STATUS SIGNAL IS WORKING
7288 ; CORRECTLY. IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED
7289 ; LOOPBACK IS SPECIFIED. THIS TEST USES THE LOOPED BACK DTR SIGNALS
7290 ; TO TEST THE DSR SIGNAL. THIS TEST IS PERFORMED ON ALL THE ACTIVE
7291 ; LINES.
7292 ;
7293 ;-----*****
7294
7295 032150          BGNTST
       032150
7296
7297 ;
7298 ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
7299 032150 032767 000002 150054
7300 032156 001002
7301 032160 000167 000420
7302 032164
       032164 012700 000240
       032170 104441
7303 000022
7304 032172 012767 000022 150062
7305 032200 012767 177777 150052
7306 032206 012767 000001 151644
7307 032214 012767 017501 151640
7308 032222 012767 007746 151634
7309
7310 ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
7311 ; CLEAR TX AND RX INTERRUPT ENABLE BITS.
7312 ; THIS SUBROUTINE REPORTS ERROR >>>> 8001 <<<<<.
7313 ;
7314 032230 004767 161340
7315 032234 103402
7316 032236 000167 000342
7317
7318 ;
7319 ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
7320 032242 004767 160672
7321
7322 ;
7323 ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7324 ; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
7325 ; A RESPONSE ON THE ASSOCIATED DSR SIGNAL.
7326 ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7327 032246 005003
7328 032250 010300
7329 032252 006300
7330 032254 036067 002344 147746
7331 032262 001454
7332
7333 ;
7334 ; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
7335 032264 005000
7336 032266 012705 177777

```

```

;*****
; - DATA SET READY MODEM SIGNAL TEST -
;
; THIS TEST VERIFIES THAT THE DSR 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 DSR SIGNAL. THIS TEST IS PERFORMED ON ALL THE ACTIVE
; LINES.
;-----*****

          BGNTST
                                  T18::
;
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
;
          BIT    #BIT1,LOPBCK      ;CHECK TYPE OF LOOPBACK MODE SELECTED.
          BNE    2$
          JMP    60$                ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
2$:        SETPRI #PRI05           ;ALLOW LTC INTERRUPTS.
                                  MOV    #PRI05,R0
                                  TRAP   C$SPRI
          TNUM == TNUM + 1        ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
          MOV    #TNUM,TSTNUM     ;SET UP THE TEST NUMBER. (80)
          MOV    #-1,CTRLCF       ;INDICATE THAT WE ARE IN A TEST.
          MOV    #1,ERRTYP        ;SET ERROR TYPE IN ERROR TABLE.
          MOV    #8001,ERRNBR     ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
          MOV    #EM8001,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 >>>> 8001 <<<<<.
;
          JSR    PC,CLNRST        ;RESET THE DUT.
          BCS    4$
          JMP    60$                ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;
4$:        JSR    PC,ASLNTL       ;SET UP THE ASSOCIATED LINE TABLES.
;
; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
; THIS LOOP CLEARS ALL THE DTRS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
; A RESPONSE ON THE ASSOCIATED DSR SIGNAL.
; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
;
          CLR    R3                ;CLEAR THE LINE COUNTER.
6$:        MOV    R3,R0
          ASL    R0
          BIT    BITTBL(R0),ACTLNS
          BEQ    10$                ;DON'T TEST IF NOT ACTIVE LINE.
;
; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
;
          CLR    R0
          MOV    #MAPLNS,R5       ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
                                  ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.

```

```

7337 032272 004767 164352      JSR    PC,WTWLNCR      ;CLEAR ALL THE DUT DTR BITS.
7338 032276 012704 000050      MOV    #40.,R4
7339 032302 004767 161424      JSR    PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
7340
7341      ;+
7342      ; CHECK THAT THE SPECIFIED DSR IS CLEAR.
7343 032306 010377 147724      MOV    R3,@CSRA       ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7344 032312 032777 100000 147724 BIT    @BIT15,@FSLSA
7345 032320 001020      BNE    #0             ;GO REPORT DSR IS BAD IF BIT IS NOT CLEAR.
7346      ;+
7347      ; SET THE DTR FOR THE ASSOCIATED LINE.
7348      ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
7349      ; IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
7350      ;-
7351 032322 116304 004020      MOVB   TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
7352 032326 010477 147704      MOV    R4,@CSRA       ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
7353 032332 052777 001000 147706 BIS    @BIT9,@LNCTRA   ;SET THE ASSOCIATED LINE DTR.
7354      ;+
7355      ; CHECK THAT THE SELECTED LINE DSR IS ACTIVE.
7356      ;-
7357 032340 010377 147672      MOV    R3,@CSRA       ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7358 032344 012701 170050      MOV    #170050,R1     ;PASS TIMEOUT OF 40 MILLI-SEC. AND BIT TO TEST.
7359 032350 016702 147670      MOV    FSLSA,R2       ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7360 032354 004767 164154      JSR    PC,WAIBIS      ;WAIT FOR DSR TO BECOME SET OR TIMEOUT.
7361 032360 103415      BCS    10#           ;SKIP ERROR REPORT IF SELECTED DSR IS SET.
7362
7363
7364 032362      8#:      ;REPORT DSR MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7365 032362 012767 017502 151472 MOV    #8002.,ERRNBR   ;SELECT THE ERROR NUMBER.
7366 032370 012767 012042 151470 MOV    #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
7367 032376 012701 010012      MOV    #EM8002,R1     ;SELECT THE ERROR MESSAGE.
7368 032402      ERROR
7369      TRAP    C#ERROR
7370
7371      ;+
7372      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7373 032404 032767 000100 147606 BIT    @BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
7374 032412 001474      BEQ    60#           ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7375      ; DURING THE SOFTWARE QUESTIONS.
7376
7377 032414 005203      10#:     INC    R3           ;SELECT THE NEXT LINE NUMBER.
7378 032416 020327 000020      CMP    R3,#NUMLNS    ;TEST FOR ALL LINES DONE.
7379 032422 002712      BLT    6#           ;LOOP IF NOT ALL LINES DONE.
7380      ;+
7381      ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7382      ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7383      ; A RESPONSE ON THE SELECTED DSR SIGNAL.
7384      ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7385      ;-
7386 032424 005003      12#:     CLR    R3           ;CLEAR THE LINE COUNTER.
7387 032426 010300      MOV    R3,R0
7388 032430 006300      ASL    R0
7389 032432 036067 002344 147570 BIT    BITTBL(R0),ACTLNS
7390 032440 001455      BEQ    16#           ;DON'T TEST IF NOT ACTIVE LINE.
7391      ;+
7392      ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.

```


7444
7445
7446
7447
7448
7449
7450
7451
7452
7453
7454
7455
7456 032620
032620
7457
7458
7459
7460 032620 032767 000002 147404
7461 032626 001002
7462 032630 000167 000420
7463 032634
032634 012700 000240
032640 104441
7464 000023
7465 032642 012767 000023 147412
7466 032650 012767 177777 147402
7467 032656 012767 000001 151174
7468 032664 012767 017645 151170
7469 032672 012767 010056 151164
7470
7471
7472
7473
7474
7475 032700 004767 160670
7476 032704 103402
7477 032706 000167 000342
7478
7479
7480
7481 032712 004767 160222
7482
7483
7484
7485
7486
7487
7488 032716 005003
7489 032720 010300
7490 032722 006300
7491 032724 036067 002344 147276
7492 032732 001454
7493
7494
7495
7496 032734 005000
7497 032736 012705 177777

```
.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.
;*****
          BGNTST
          T19::
;
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
;
;          BIT    #BIT1,LOPBACK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
;          BNE    2#
;          JMP    60#
;          SETPRI #PRI05          ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
;                                ;ALLOW LTC INTERRUPTS.
;
;                                MOV    #PRI05,RO
;                                TRAP   C#SPRI
;          TNUM == TNUM + 1      ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
;          MOV    #TNUM,TSTNUM   ;SET UP THE TEST NUMBER. (81)
;          MOV    #-1,CTRLCF     ;INDICATE THAT WE ARE IN A TEST.
;          MOV    #1,ERRTYP      ;SET ERROR TYPE IN ERROR TABLE.
;          MOV    #8101,ERRNBR   ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
;          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 <<<<<.
;
;          JSR    PC,CLNRST      ;RESET THE DUT.
;          BCS    4#
;          JMP    60#          ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;
;          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.
;
;          CLR    R3              ;CLEAR THE LINE COUNTER.
;          MOV    R3,RO
;          ASL    RO
;          BIT    BITBL(RO),ACTLNS
;          BEQ    10#          ;DON'T TEST IF NOT ACTIVE LINE.
;
; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
;
;          CLR    RO
;          MOV    #MAPLNS,R5    ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
;                                ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
```

```

7498 032742 004767 163702      JSR    PC,WTWLNLC      ;CLEAR ALL THE DUT DTR BITS.
7499 032746 012704 000050      MOV    #40,,R4
7500 032752 004767 160754      JSR    PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
7501
7502      ;*
7503      ; CHECK THAT THE SPECIFIED RI IS CLEAR.
7504 032756 010377 147254      MOV    R3,BCSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7505 032762 032777 020000 147254  BIT    #BIT13,#FSLSA
7506 032770 001020          BNE    #0              ;GO REPORT RI IS BAD IF BIT IS NOT CLEAR.
7507
7508      ;*
7509      ; SET THE DTR FOR THE ASSOCIATED LINE.
7510      ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
7511      ; IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
7512 032772 116304 004020      MOV    TXRLNB(R3),R4    ;GET THE ASSOCIATED LINE NUMBER.
7513 032776 010477 147234      MOV    R4,BCSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
7514 033002 052777 001000 147236  BIS    #BIT9,#LNCTRA    ;SET THE ASSOCIATED LINE DTR.
7515
7516      ;*
7517      ; CHECK THAT THE SELECTED LINE RI IS ACTIVE.
7518 033010 010377 147222      MOV    R3,BCSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7519 033014 012701 150050      MOV    #150050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
7520 033020 016702 147220      MOV    FLSA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7521 033024 004767 163504      JSR    PC,WAIBIS       ;WAIT FOR RI TO BECOME SET OR TIMEOUT.
7522 033030 103415          BCS    100            ;SKIP ERROR REPORT IF SELECTED RI IS SET.
7523
7524
7525 033032          #0: ;REPORT RI MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7526 033032 012767 017646 151022  MOV    #8102,,ERRNBR   ;SELECT THE ERROR NUMBER.
7527 033040 012767 012042 151020  MOV    #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
7528 033046 012701 010121          MOV    #EM8102,R1      ;SELECT THE ERROR MESSAGE.
7529 033052          ERROR
7530          TRAP    C#ERROR
7531
7532      ;*
7533      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7534 033054 032767 000100 147136  BIT    #BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
7535 033062 001474          BEQ    600            ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7536          ;DURING THE SOFTWARE QUESTIONS.
7537
7538 033064 005203          100: INC    R3          ;SELECT THE NEXT LINE NUMBER.
7539 033066 020327 000020      CMP    R3,#NUMLNS     ;TEST FOR ALL LINES DONE.
7540 033072 002712          BLT    60            ;LOOP IF NOT ALL LINES DONE.
7541
7542      ;*
7543      ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7544      ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7545      ; A RESPONSE ON THE SELECTED RI SIGNAL.
7546      ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7547 033074 005003          ;-
7548 033076 010300          120: CLR    R3          ;CLEAR THE LINE COUNTER.
7549 033100 006300      MOV    R3,R0
7550 033102 036067 002344 147120  ASL    R0
7551 033110 001455      BIT    BITBL(R0),ACTLNS
7552          BEQ    160            ;DON'T TEST IF NOT ACTIVE LINE.
7553      ;*
7553      ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.

```

```

7554
7555 033112 012700 001000
7556 033116 012705 177777
7557 033122 004767 163522
7558 033126 012704 000050
7559 033132 004767 160574
7560
7561
7562
7563 033136 010377 147074
7564 033142 032777 020000 147074
7565 033150 001420
7566
7567
7568
7569
7570
7571 033152 116304 004020
7572 033156 010477 147054
7573 033162 042777 001000 147056
7574
7575
7576
7577 033170 010377 147042
7578 033174 012701 150050
7579 033200 016702 147040
7580 033204 004767 163250
7581 033210 103415
7582
7583 033212
7584 033212 012767 017647 150642
7585 033220 012767 012042 150640
7586 033226 012701 010121
7587 033232
    033232 104460
7588
7589
7590
7591
7592 033234 032767 000100 146756
7593 033242 001404
7594
7595
7596 033244 005203
7597 033246 020327 000020
7598 033252 002711
7599
7600 033254 005067 147000
7601 033260
    033260 012700 000340
    033264 104441
7602
7603 033266
    033266
    033266 104401

; -
      MOV  #BIT9,R0           ;SPECIFY THAT DTR BITS ARE TO BE SET.
      MOV  #MAPLNS,R5        ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
      JSR  PC,WTWLNLC        ;SET ALL THE DUT DTR BITS.
      MOV  #40.,R4           ;
      JSR  PC,DELAY          ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
; *
; CHECK THAT THE SPECIFIED RI IS SET.
; -
      MOV  R3,#CSRA          ;SET IND.ADR.REG FIELD TO SELECTED LINE.
      BIT  #BIT13,#FSLSA    ;
      BEQ  14#              ;GO REPORT RI IS BAD IF BIT IS NOT SET.
; *
; CLEAR THE DTR FOR THE ASSOCIATED LINE.
; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, DTR WILL NOT HAVE BEEN TESTED
; IN THE DTR TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
; -
      MOVB TXRLNB(R3),R4     ;GET THE ASSOCIATED LINE NUMBER.
      MOV  R4,#CSRA          ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
      BIC  #BIT9,#LNCTRA    ;CLEAR THE ASSOCIATED LINE DTR.
; *
; CHECK THAT THE SELECTED LINE RI IS CLEAR.
; -
      MOV  R3,#CSRA          ;SET IND.ADR.REG FIELD TO SELECTED LINE.
      MOV  #150050,R1        ;PASS TIMEOUT OF 40 MILLI-SEC. AND BIT TO TEST.
      MOV  FSLSA,R2          ;PASS THE ADDRESS OF THE REGISTER TO TEST.
      JSR  PC,WAIBIC         ;WAIT FOR RI TO BECOME CLEAR OR TIMEOUT.
      BCS  16#              ;SKIP ERROR REPORT IF SELECTED RI IS CLEAR.
14#:  ;REPORT RI MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
      MOV  #8103.,ERRNBR     ;SELECT THE ERROR NUMBER.
      MOV  #ER7801,ERRBLK    ;SELECT THE ERROR PRINT ROUTINE.
      MOV  #EM8102,M1        ;SELECT THE ERROR MESSAGE.
      ERROR
                                 TRAP  C#ERROR
; *
; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
; -
      BIT  #BIT06,OPTION     ;EXIT WITH TEST FAILURE MESSAGE IF
      BEQ  60#              ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
                                 ;DURING THE SOFTWARE QUESTIONS.
16#:  INC  R3                ;SELECT THE NEXT LINE NUMBER.
      CMP  R3,#NUMPLNS      ;TEST FOR ALL LINES DONE.
      BLT  12#              ;LOOP IF NOT ALL LINES DONE.
60#:  CLR  CTRLCF           ;INDICATE THAT WE ARE NOT WITHIN A TEST.
      SETPRI #PRI07         ;DISABLE ALL INTERRUPTS.
                                 MOV  #PRI07,R0
                                 TRAP  C#SPRI
      ENDTST
L10045: TRAP  C#ETST

```

```

7605
7606
7607
7608
7609
7610
7611
7612
7613
7614
7615
7616
7617 033270
      033270
7618
7619
7620
7621 033270 032767 000002 146734
7622 033276 001002
7623 033300 000167 000420
7624 033304
      033304 012700 000240
      033310 104441
7625      000024
7626 033312 012767 000024 146742
7627 033320 012767 177777 146732
7628 033326 012767 000001 150524
7629 033334 012767 020011 150520
7630 033342 012767 010164 150514
7631
7632
7633
7634
7635
7636 033350 004767 160220
7637 033354 103402
7638 033356 000167 000342
7639
7640
7641
7642 033362 004767 157552
7643
7644
7645
7646
7647
7648
7649 033366 005003
7650 033370 010300
7651 033372 006300
7652 033374 036067 002344 146626
7653 033402 001454
7654
7655
7656
7657 033404 005000
7658 033406 012705 177777

```

```

.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
      T20::
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
;-
      BIT      #BIT1,LOPBCK ;CHECK TYPE OF LOOPBACK MODE SELECTED.
      BNE     2#
      JMP     60# ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
2# : SETPRI   #PRI05 ;ALLOW LTC INTERRUPTS.
                                 MOV     #PRI05,R0
                                 TRAP   C#SPRI
      TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
      MOV     #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (82)
      MOV     #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
      MOV     #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
      MOV     #8201,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
      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 <<<<<.
;-
      JSR     PC,CLNRST ;RESET THE DUT.
      BCS     4#
      JMP     60# ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;-
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.
;-
      CLR     R3 ;CLEAR THE LINE COUNTER.
6# : MOV     R3,R0
      ASL     R0
      BIT     BITTBL(R0),ACTLNS
      BEQ     10# ;DON'T TEST IF NOT ACTIVE LINE.
;
; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
;-
      CLR     R0 ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
      MOV     #MAPLNS,R5 ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.

```

```

7659 033412 004767 163232      JSR    PC,WTWLNCR      ;CLEAR ALL THE DUT RTS BITS.
7660 033416 012704 000050      MOV    #40.,R4
7661 033422 004767 160304      JSR    PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
7662                                ;+
7663                                ; CHECK THAT THE SPECIFIED CTS IS CLEAR.
7664                                ;-
7665 033426 010377 146604      MOV    R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7666 033432 032777 004000 146604  BIT    @BIT11,@FSLSA
7667 033440 001020                BNE    @#             ;GO REPORT CTS IS BAD IF BIT IS NOT CLEAR.
7668                                ;+
7669                                ; SET THE RTS FOR THE ASSOCIATED LINE.
7670                                ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
7671                                ;       IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
7672                                ;-
7673 033442 116304 004020      MOVB   TXRLNB(R3),R4   ;GET THE ASSOCIATED LINE NUMBER.
7674 033446 010477 146564      MOV    R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
7675 033452 052777 010000 146566  BIS    @BIT12,@LNCTRA ;SET THE ASSOCIATED LINE RTS.
7676                                ;+
7677                                ; CHECK THAT THE SELECTED LINE CTS IS ACTIVE.
7678                                ;-
7679 033460 010377 146552      MOV    R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7680 033464 012701 130050      MOV    #130050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC, AND BIT TO TEST.
7681 033470 016702 146550      MOV    FLSA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7682 033474 004767 163034      JSR    PC,WAIBIS       ;WAIT FOR CTS TO BECOME SET OR TIMEOUT.
7683 033500 103415                BCS    10#            ;SKIP ERROR REPORT IF SELECTED CTS IS SET.
7684
7685
7686 033502                @#:    ;REPORT CTS MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7687 033502 012767 020012 150352  MOV    @#202.,ERRNBR   ;SELECT THE ERROR NUMBER.
7688 033510 012767 012042 150350  MOV    @#ER7801,ERRBLK ;SELECT THE ERROR PRINT ROUTINE.
7689 033516 012701 010230      MOV    @#EM8202,R1     ;SELECT THE ERROR MESSAGE.
7690 033522                ERROR
7691                                TRAP    C#ERROR
7692
7693                                ;+
7694                                ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7695                                ;-
7695 033524 032767 000100 146466  BIT    @BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
7696 033532 001474                BEQ    60#             ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7697                                ;DURING THE SOFTWARE QUESTIONS.
7698
7699 033534 005203 10#:    INC    R3             ;SELECT THE NEXT LINE NUMBER.
7700 033536 020327 000020      CMP    R3,#NUNLNS     ;TEST FOR ALL LINES DONE.
7701 033542 002712                BLT    6#             ;LOOP IF NOT ALL LINES DONE.
7702                                ;+
7703                                ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7704                                ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7705                                ;       A RESPONSE ON THE SELECTED CTS SIGNAL.
7706                                ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7707                                ;-
7708 033544 005003 12#:    CLR    R3             ;CLEAR THE LINE COUNTER.
7709 033546 010300      MOV    R3,R0
7710 033550 006300      ASL    R0
7711 033552 036067 002344 146450  BIT    BITTBL(R0),ACTLNS
7712 033560 001455                BEQ    16#            ;DON'T TEST IF NOT ACTIVE LINE.
7713                                ;+
7714                                ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.

```

```

7715
7716 033562 012700 010000      ;
7717 033566 012705 177777      ;
7718 033572 004767 163052      ;
7719 033576 012704 000050      ;
7720 033602 004767 160124      ;
7721
7722      ;+
7723      ; CHECK THAT THE SPECIFIED CTS IS SET.
7724 033606 010377 146424      ;
7725 033612 032777 004000 146424      ;
7726 033620 001420      ;
7727      ;+
7728      ; CLEAR THE RTS FOR THE ASSOCIATED LINE.
7729      ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
7730      ; IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
7731
7732 033622 116304 004020      ;
7733 033626 010477 146404      ;
7734 033632 042777 010000 146406      ;
7735      ;+
7736      ; CHECK THAT THE SELECTED LINE CTS IS CLEAR.
7737
7738 033640 010377 146372      ;
7739 033644 012701 130050      ;
7740 033650 016702 146370      ;
7741 033654 004767 162600      ;
7742 033660 103415      ;
7743
7744 033662      ;
7745 033662 012767 020013 150172      ;
7746 033670 012767 012042 150170      ;
7747 033676 012701 010230      ;
7748 033702      ;
7749 033702 104460      ;
7750
7751      ;+
7752      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7753 033704 032767 000100 146306      ;
7754 033712 001404      ;
7755      ;+
7756      ;
7757 033714 005203      ;
7758 033716 020327 000020      ;
7759 033722 002711      ;
7760
7761 033724 005067 146330      ;
7762 033730      ;
7763 033730 012700 000340      ;
7764 033734 104441      ;
7765
7766      ;
7767      ;
7768      ;
7769      ;
7770      ;
7771      ;
7772      ;
7773      ;
7774      ;
7775      ;
7776      ;
7777      ;
7778      ;
7779      ;
7780      ;
7781      ;
7782      ;
7783      ;
7784      ;
7785      ;
7786      ;
7787      ;
7788      ;
7789      ;
7790      ;
7791      ;
7792      ;
7793      ;
7794      ;
7795      ;
7796      ;
7797      ;
7798      ;
7799      ;
7800      ;
7801      ;
7802      ;
7803      ;
7804      ;
7805      ;
7806      ;
7807      ;
7808      ;
7809      ;
7810      ;
7811      ;
7812      ;
7813      ;
7814      ;
7815      ;
7816      ;
7817      ;
7818      ;
7819      ;
7820      ;
7821      ;
7822      ;
7823      ;
7824      ;
7825      ;
7826      ;
7827      ;
7828      ;
7829      ;
7830      ;
7831      ;
7832      ;
7833      ;
7834      ;
7835      ;
7836      ;
7837      ;
7838      ;
7839      ;
7840      ;
7841      ;
7842      ;
7843      ;
7844      ;
7845      ;
7846      ;
7847      ;
7848      ;
7849      ;
7850      ;
7851      ;
7852      ;
7853      ;
7854      ;
7855      ;
7856      ;
7857      ;
7858      ;
7859      ;
7860      ;
7861      ;
7862      ;
7863      ;
7864      ;
7865      ;
7866      ;
7867      ;
7868      ;
7869      ;
7870      ;
7871      ;
7872      ;
7873      ;
7874      ;
7875      ;
7876      ;
7877      ;
7878      ;
7879      ;
7880      ;
7881      ;
7882      ;
7883      ;
7884      ;
7885      ;
7886      ;
7887      ;
7888      ;
7889      ;
7890      ;
7891      ;
7892      ;
7893      ;
7894      ;
7895      ;
7896      ;
7897      ;
7898      ;
7899      ;
7900      ;
7901      ;
7902      ;
7903      ;
7904      ;
7905      ;
7906      ;
7907      ;
7908      ;
7909      ;
7910      ;
7911      ;
7912      ;
7913      ;
7914      ;
7915      ;
7916      ;
7917      ;
7918      ;
7919      ;
7920      ;
7921      ;
7922      ;
7923      ;
7924      ;
7925      ;
7926      ;
7927      ;
7928      ;
7929      ;
7930      ;
7931      ;
7932      ;
7933      ;
7934      ;
7935      ;
7936      ;
7937      ;
7938      ;
7939      ;
7940      ;
7941      ;
7942      ;
7943      ;
7944      ;
7945      ;
7946      ;
7947      ;
7948      ;
7949      ;
7950      ;
7951      ;
7952      ;
7953      ;
7954      ;
7955      ;
7956      ;
7957      ;
7958      ;
7959      ;
7960      ;
7961      ;
7962      ;
7963      ;
7964      ;
7965      ;
7966      ;
7967      ;
7968      ;
7969      ;
7970      ;
7971      ;
7972      ;
7973      ;
7974      ;
7975      ;
7976      ;
7977      ;
7978      ;
7979      ;
7980      ;
7981      ;
7982      ;
7983      ;
7984      ;
7985      ;
7986      ;
7987      ;
7988      ;
7989      ;
7990      ;
7991      ;
7992      ;
7993      ;
7994      ;
7995      ;
7996      ;
7997      ;
7998      ;
7999      ;
8000      ;

```

```

7766
7767
7768
7769
7770
7771
7772
7773
7774
7775
7776
7777
7778 033740
      033740
7779
7780
7781
7782 033740 032767 000002 146264
7783 033746 001002
7784 033750 000167 000420
7785 033754
      033754 012700 000240
      033760 104441
7786      000025
7787 033762 012767 000025 146272
7788 033770 012767 177777 146262
7789 033776 012767 000001 150054
7790 034004 012767 020155 150050
7791 034012 012767 010274 150044
7792
7793
7794
7795
7796
7797 034020 004767 157550
7798 034024 103402
7799 034026 000167 000342
7800
7801
7802
7803 034032 004767 157102
7804
7805
7806
7807
7808
7809
7810 034036 005003
7811 034040 010300
7812 034042 006300
7813 034044 036067 002344 146156
7814 034052 001454
7815
7816
7817
7818 034054 005000
7819 034056 012705 177777

```

```

.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.
;*****
      BGNTST
      T21::
;+
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
;-
      BIT   #BIT1,LOPBCK      ;CHECK TYPE OF LOOPBACK MODE SELECTED.
      BNE   2$
      JMP   60$
2$:     SETPRI #PRI05        ;ALLOW LTC INTERRUPTS.
      MOV   #PRI05,R0
      TRAP C$SPRI
      TNUM == TNUM + 1      ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
      MOV   #TNUM,TSTNUM    ;SET UP THE TEST NUMBER. (83)
      MOV   #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
      MOV   #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
      MOV   #8301,ERRNBR    ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
      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 <<<<<.
;-
      JSR   PC,CLNRST      ;RESET THE DUT.
      BCS   4$
      JMP   60$           ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;+
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;-
4$:     JSR   PC,ASLNTL    ;SET UP THE ASSOCIATED LINE TABLES.
;+
; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND WAITS FOR
; A RESPONSE ON THE ASSOCIATED DCD SIGNAL.
; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
;-
6$:     CLR   R3           ;CLEAR THE LINE COUNTER.
      MOV   R3,R0
      ASL   R0
      BIT   BITTBL(R0),ACTLNS
      BEQ   10$          ;DON'T TEST IF NOT ACTIVE LINE.
;+
; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
;-
      CLR   R0
      MOV   #MAPLNS,R5   ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
                          ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.

```

```

7820 034062 004767 162562          JSR    PC,WTWLNLC      ;CLEAR ALL THE DUT RTS BITS.
7821 034066 012704 000050          MOV    #40.,R4
7822 034072 004767 157634          JSR    PC,DELAY        ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
7823
7824          ;+
7825          ; CHECK THAT THE SPECIFIED DCD IS CLEAR.
7826 034076 010377 146134          MOV    R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7827 034102 032777 010000 146134  BIT    @BIT12,@FSLSA
7828 034110 001020                    BNE    #0                ;GO REPORT DCD IS BAD IF BIT IS NOT CLEAR.
7829
7830          ;+
7831          ; SET THE RTS FOR THE ASSOCIATED LINE.
7832          ; NOTE: IF THE ASSOCIATED LINE IS NOT SELECTED, RTS WILL NOT HAVE BEEN TESTED
7833          ; IN THE RTS TEST (ONLY AN ISSUE IN STAGGERED LOOPBACK).
7834 034112 116304 004020          MOV    TXRLNB(R3),R4    ;GET THE ASSOCIATED LINE NUMBER.
7835 034116 010477 146114          MOV    R4,@CSRA        ;SET IND.ADR.REG FIELD TO ASSOCIATED LINE.
7836 034122 052777 010000 146116  BIS    @BIT12,@LNCTRA   ;SET THE ASSOCIATED LINE RTS.
7837
7838          ;+
7839          ; CHECK THAT THE SELECTED LINE DCD IS ACTIVE.
7840 034130 010377 146102          MOV    R3,@CSRA        ;SET IND.ADR.REG FIELD TO SELECTED LINE.
7841 034134 012701 140050          MOV    #140050,R1      ;PASS TIMEOUT OF 40 MILLI-SEC. AND BIT TO TEST.
7842 034140 016702 146100          MOV    FLSA,R2         ;PASS THE ADDRESS OF THE REGISTER TO TEST.
7843 034144 004767 162364          JSR    PC,WAIBIS       ;WAIT FOR DCD TO BECOME SET OR TIMEOUT.
7844 034150 103415                    BCS    10#              ;SKIP ERROR REPORT IF SELECTED DCD IS SET.
7845
7846
7847 034152                    8# : ;REPORT DCD MODEM CONTROL SIGNAL DEFECTIVE ON LINE NN.
7848 034152 012767 020156 147702  MOV    #8302.,ERRNBR   ;SELECT THE ERROR NUMBER.
7849 034160 012767 012042 147700  MOV    #ER7801,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
7850 034166 012701 010340          MOV    #EM8302,R1      ;SELECT THE ERROR MESSAGE.
7851 034172                    ERROR
7852                    TRAP    C#ERROR
7853
7854          ;+
7855          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
7856 034174 032767 000100 146016  BIT    @BIT06,OPTION    ;EXIT WITH TEST FAILURE MESSAGE IF
7857 034202 001474                    BEQ    60#              ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
7858                    ;DURING THE SOFTWARE QUESTIONS.
7859
7860 034204 005203                    10# : INC    R3                ;SELECT THE NEXT LINE NUMBER.
7861 034206 020327 000020          CMP    R3,@NUMLNS      ;TEST FOR ALL LINES DONE.
7862 034212 002712                    BLT    6#                ;LOOP IF NOT ALL LINES DONE.
7863
7864          ;+
7865          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
7866          ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND WAITS FOR
7867          ; A RESPONSE ON THE SELECTED DCD SIGNAL.
7868          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
7869 034214 005003                    ;-
7870 034216 010300                    12# : CLR    R3                ;CLEAR THE LINE COUNTER.
7871 034220 006300          MOV    R3,R0
7872 034222 036067 002344 146000  ASL    R0
7873 034230 001455          BIT    BITTBL(R0),ACTLNS
7874                    BEQ    16#              ;DON'T TEST IF NOT ACTIVE LINE.
7875          ;+
7875          ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
    
```



```

7927
7928
7929
7930
7931
7932
7933
7934
7935
7936
7937
7938 034410
      034410
7939
7940
7941
7942 034410 032767 000002 145614
7943 034416 001002
7944 034420 000167 000400
7945 034424
      034424 012700 000240
      034430 104441
7946      000026
7947 034432 012767 000026 145622
7948 034440 012767 177777 145612
7949 034446 012767 000001 147404
7950 034454 012767 020321 147400
7951 034462 012767 010404 147374
7952
7953
7954
7955
7956
7957 034470 004767 157100
7958 034474 103402
7959 034476 000167 000322
7960
7961
7962
7963 034502 004767 156432
7964
7965
7966
7967
7968
7969
7970 034506 005003
7971 034510 010300
7972 034512 006300
7973 034514 036067 002344 145506
7974 034522 001450
7975
7976
7977
7978 034524 005000
7979 034526 012705 177777
7980 034532 004767 162112

```

```

.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
                                     T22::
;+
; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGARED LOOPBACK MODE.
;-
      BIT    #BIT1,LOPBCK    ;CHECK TYPE OF LOOPBACK MODE SELECTED.
      BNE   2#
      JMP   60#
2#:    SETPRI #PRI05        ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
                                     ;ALLOW LTC INTERRUPTS.
                                     MOV    #PRI05,R0
                                     TRAP   C#SPRI
      TNUM  == TNUM + 1      ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
      MOV   #TNUM,TSTNUM    ;SET UP THE TEST NUMBER. (84)
      MOV   #-1,CTRLCF      ;INDICATE THAT WE ARE IN A TEST.
      MOV   #1,ERRTYP       ;SET ERROR TYPE IN ERROR TABLE.
      MOV   #8401,ERRNBR    ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
      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 <<<<<.
;-
      JSR   PC,CLNRST      ;RESET THE DUT.
      BCS  4#
      JMP   60#            ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
;+
; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
;-
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.
;-
6#:    CLR   R3            ;CLEAR THE LINE COUNTER.
      MOV   R3,R0
      ASL   R0
      BIT   BITTBL(R0),ACTLNS
      BEQ   8#            ;DON'T TEST IF NOT ACTIVE LINE.
;+
; CLEAR ALL THE DUT LNCTRL REGISTERS DTR BITS.
;-
      CLR   R0            ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
      MOV   #MAPLNS,R5    ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
      JSR   PC,WTWLNLC    ;CLEAR ALL THE DUT DTR BITS.

```

```

7981 034536 012704 000050          MOV    #40.,R4
7982 034542 004767 157164          JSR    PC,DELAY          ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
7983                                     ;+
7984                                     ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
7985                                     ;-
7986 034546 004767 160776          JSR    PC,SAVMST        ;SAVE THE PRESENT MODEM STATUS STATES.
7987                                     ;+
7988                                     ; SET THE DTR FOR THE SELECTED LINE.
7989                                     ;-
7990 034552 010377 145460          MOV    R3,BCSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
7991 034556 052777 001000 145462  BIS    #BIT9,BLNCTRA   ;SET THE SELECTED LINE DTR.
7992 034564 012704 000050          MOV    #40.,R4
7993 034570 004767 157136          JSR    PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
7994                                     ;+
7995                                     ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
7996                                     ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
7997                                     ;-
7998 034574 116301 004020          MOVB   TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
7999 034600 012702 120000          MOV    #BIT15!BIT13,R2 ;IGNORE DSR AND RI ON ASSOCIATED LINE.
8000 034604 004767 157030          JSR    PC,CMPMST       ;COMPARE OLD AND NEW STAT CONTENTS.
8001 034610 103415                                     BCS    8#              ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
8002                                     ;REPORT INTERACTIONS FOUND BETWEEN DTR FOR LINE NN AND THE FOLLOWING SIGNALS:
8003 034612 012767 020322 147242  MOV    #8402.,ERRNBR   ;SELECT THE ERROR NUMBER.
8004 034620 012767 012100 147240  MOV    #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8005 034626 012701 010466          MOV    #EM8402,R1     ;SELECT THE DTR ERROR MESSAGES.
8006 034632                                     ERROR                                     ;ER8401 USES R1, R2, AND R3 VALUES.
8007 034632 104460                                     TRAP    C#ERROR
8008
8009                                     ;+
8010                                     ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
8011 034634 032767 000100 145356  BIT    #BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
8012 034642 001470                                     BEQ    60#             ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
8013                                     ;DURING THE SOFTWARE QUESTIONS.
8014                                     ;+
8015                                     ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
8016                                     ;-
8017 034644 005203                                     8#:    INC    R3        ;SELECT THE NEXT LINE NUMBER.
8018 034646 020327 000020          CMP    R3,#NUMLNS     ;TEST FOR ALL LINES DONE.
8019 034652 002716                                     BLT    6#             ;LOOP IF NOT ALL LINES DONE.
8020                                     ;+
8021                                     ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8022                                     ; THIS LOOP SETS ALL THE DTRS AND THEN CLEARS THEM INDIVIDUALLY AND CHECKS
8023                                     ; FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED RI AND DSR SIGNALS.
8024                                     ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8025                                     ;-
8026 034654 005003                                     CLR    R3             ;CLEAR THE LINE COUNTER.
8027 034656 010300 10#:    MOV    R3,R0
8028 034660 006300                                     ASL    R0
8029 034662 036067 002344 145340  BIT    BITTBL(R0),ACTLNS
8030 034670 001451                                     BEQ    12#           ;DON'T TEST IF NOT ACTIVE LINE.
8031                                     ;+
8032                                     ; SET ALL THE DUT LNCTRL REGISTERS DTR BITS.
8033                                     ;-
8034 034672 012700 001000          MOV    #BIT9,R0       ;SPECIFY THAT DTR BITS ARE TO BE SET.
8035 034676 012705 177777          MOV    #MAPLNS,R5    ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8036 034702 004767 161742          JSR    PC,WTWLNLC    ;SET ALL THE DUT DTR BITS.

```

```

8037 034706 012704 000050          MOV    #40.,R4
8038 034712 004767 157014          JSR    PC,DELAY          ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
8039
8040          ;+
8041          ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
8042 034716 004767 160626          JSR    PC,SAVMST        ;SAVE THE PRESENT MODEM STATUS STATES.
8043
8044          ;+
8045          ; CLEAR THE DTR FOR THE SELECTED LINE.
8046 034722 010377 145310          MOV    R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8047 034726 042777 001000 145312  BIC    #BIT9,@LNCTRA    ;CLEAR THE SELECTED LINE DTR.
8048 034734 012704 000050          MOV    #40.,R4
8049 034740 004767 156766          JSR    PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
8050
8051          ;+
8052          ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
8053          ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
8054 034744 116301 004020          MOV    TXRLNB(R3),R1    ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
8055 034750 012702 120000          MOV    #BIT15!BIT13,R2 ;IGNORE DSR AND RI ON ASSOCIATED LINE.
8056 034754 004767 156660          JSR    PC,CMPMST        ;COMPARE OLD AND NEW STAT CONTENTS.
8057 034760 103415          BCS    12$              ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
8058          ;REPORT INTERACTIONS FOUND BETWEEN DTR FOR LINE NN AND THE FOLLOWING SIGNALS:
8059 034762 012767 020323 147072  MOV    #8403.,ERRNBR    ;SELECT THE ERROR NUMBER.
8060 034770 012767 012100 147070  MOV    #ER8401,ERRBLK    ;SELECT THE ERROR PRINT ROUTINE.
8061 034776 012701 010466          MOV    #EM8402,R1        ;SELECT THE DTR ERROR MESSAGES.
8062 035002          ERROR          ;ER8401 USES R1, R2, AND R3 VALUES.
8063          TRAP    C#ERROR
8064
8065          ;+
8066          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
8067 035004 032767 000100 145206  BIT    #BIT06,OPTION    ;EXIT WITH TEST FAILURE MESSAGE IF
8068 035012 001404          BEQ    60$              ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
8069          ;DURING THE SOFTWARE QUESTIONS.
8070
8071          ;+
8072          ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
8073          ;-
8074 035014 005203          12$: INC    R3              ;SELECT THE NEXT LINE NUMBER.
8075 035016 020327 000020          CMP    R3,#NUMLNS      ;TEST FOR ALL LINES DONE.
8076 035022 002715          BLT    10$              ;LOOP IF NOT ALL LINES DONE.
8077
8078 035024 005067 145230          60$: CLR    CTRLCF          ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8079 035030          SETPRI #PKI07        ;DISABLE ALL INTERRUPTS.
8080          MOV    #PRI07,R0
8081 035036          TRAP    C#SPRI
035036          L10050:
035036 104401          TRAP    C#ETST
ENDTST

```

```

8083 .SBTTL HARDWARE TEST - RTSINT -
8084 ;*****
8085 ;* - REQUEST TO SEND SIGNAL INTERACTIONS TEST -
8086 ;*
8087 ;* THIS TEST VERIFIES THAT THE RTS SIGNAL (AND THE LOOPED BACK DCD AND CTS
8088 ;* STATUS SIGNALS) DO NOT INTERACT WITH ANY OTHER MODEM STATUS SIGNALS.
8089 ;* IT WILL ONLY BE PERFORMED IF EITHER 25 PIN OR STAGGERED LOOPBACK IS
8090 ;* SPECIFIED. THIS TEST IS PERFORMED ON ALL ACTIVE LINES.
8091 ;*
8092 ;-----*****
8093
8094 035040 BGNTST
8095 035040 T23::
8096 ;*
8097 ; ONLY PERFORM THIS TEST IF THE DUT IS IN EXTERNAL OR STAGGERED LOOPBACK MODE.
8098 035040 032767 000002 145164 BIT #BIT1,LOPBCK ;CHECK TYPE OF LOOPBACK MODE SELECTED.
8099 035046 001002 BNE 2#
8100 035050 000167 000400 JMP 60# ;EXIT THIS TEST IF IN INTERNAL LOOPBACK.
8101 035054 2# SETPRI #PRIOS ;ALLOW LTC INTERRUPTS.
8102 035054 012700 000240 MOV #PRIOS,RO
8103 035062 104441 TRAP C#SPRI
8104 035062 000027 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8105 035070 012767 000027 145172 MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (85)
8106 035070 012767 177777 145162 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
8107 035076 012767 000001 146754 MOV #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
8108 035104 012767 020465 146750 MOV #8501.,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
8109 035112 012767 010511 146744 MOV #EM8501,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8110 ;*
8111 ; RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
8112 ; CLEAR TX AND RX INTERRUPT ENABLE BITS.
8113 ; THIS SUBROUTINE REPORTS ERROR >>>> 8501 <<<<<.
8114 035120 004767 156450 JSR PC,CLNRST ;RESET THE DUT.
8115 035124 103402 BCS 4#
8116 035126 000167 000322 JMP 60# ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
8117 ;*
8118 ; SET UP THE TX/RX ASSOCIATED LINE NUMBER TABLE.
8119 035132 004767 156002 4# JSR PC,ASLNTL ;SET UP THE ASSOCIATED LINE TABLES.
8120 ;*
8121 ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8122 ; THIS LOOP CLEARS ALL THE RTSS AND THEN SETS THEM INDIVIDUALLY AND CHECKS
8123 ; FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED DCD AND CTS SIGNALS.
8124 ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8125 ;*
8126 035136 005003 CLR R3 ;CLEAR THE LINE COUNTER.
8127 035140 010300 MOV R3,RO
8128 035142 006300 ASL RO
8129 035144 036067 002344 145056 BIT BITTBL(RO),ACTLNS
8130 035152 001450 BEQ 8# ;DON'T TEST IF NOT ACTIVE LINE.
8131 ;*
8132 ; CLEAR ALL THE DUT LNCTRL REGISTERS RTS BITS.
8133 ;*
8134 035154 005000 CLR RO ;SPECIFY THAT ALL LNCTRL BITS TO BE CLEARED.
8135 035156 012705 177777 MOV #MAPLNS,R5 ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8136 035162 004767 161462 JSR PC,WTWLNLC ;CLEAR ALL THE DUT RTS BITS.
    
```

```

8137 035166 012704 000050          MOV    #40.,R4
8138 035172 004767 156534          JSR    PC,DELAY          ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
8139
8140          ;*
8141          ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
8142 035176 004767 160346          JSR    PC,SAVMST        ;SAVE THE PRESENT MODEM STATUS STATES.
8143
8144          ;*
8145          ; SET THE RTS FOR THE SELECTED LINE.
8146 035202 010377 145030          MOV    R3,@CSRA        ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8147 035206 052777 010000 145032  BIS    #BIT12,@LNCTRA   ;SET THE SELECTED LJNE RTS.
8148 035214 012704 000050          MOV    #40.,R4
8149 035220 004767 156506          JSR    PC,DELAY        ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
8150
8151          ;*
8152          ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
8153          ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
8154 035224 116301 004020          MOV    TXRLNB(R3),R1   ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
8155 035230 012702 014000          MOV    #BIT12!BIT11,R2 ;IGNORE DCD AND CTS ON ASSOCIATED LINE.
8156 035234 004767 156400          JSR    PC,CMPHST       ;COMPARE OLD AND NEW STAT CONTENTS.
8157 035240 103415          BCS    #             ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
8158          ;REPORT INTERACTIONS FOUND BETWEEN RTS FOR LINE NN AND THE FOLLOWING SIGNALS:
8159 035242 012767 020466 146612  MOV    #8502.,ERRNBR   ;SELECT THE ERROR NUMBER.
8160 035250 012767 012100 146610  MOV    #ER8401,ERRBLK  ;SELECT THE ERROR PRINT ROUTINE.
8161 035256 012701 010573          MOV    #EM8502,R1     ;SELECT THE RTS ERROR MESSAGES.
8162 035262          ERROR          ;ER1901 USES R1, R2, AND R3 VALUES.
8163          TRAP    C#ERROR
8164
8165          ;*
8166          ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
8167 035264 032767 000100 144726  BIT    #BIT06,OPTION   ;EXIT WITH TEST FAILURE MESSAGE IF
8168 035272 001470          BEQ    #60#           ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
8169          ;DURING THE SOFTWARE QUESTIONS.
8170
8171          ;*
8172          ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
8173 035274 005203          ;*
8174 035276 020327 000020 80:   INC    R3             ;SELECT THE NEXT LINE NUMBER.
8175 035302 002716          CMP    R3,#NUMLNS     ;TEST FOR ALL LINES DONE.
8176          BLT    #6#           ;LOOP IF NOT ALL LINES DONE.
8177
8178          ;*
8179          ; SET UP A LOOP WHICH HANDLES ONE LINE PER ITERATION.
8180          ; THIS LOOP SETS ALL THE RTSS AND THEN CLEARS THEM INDIVIDUALLY AND CHECKS
8181          ; FOR ANY RESPONSES ON SIGNALS OTHER THAN THE ASSOCIATED DCD AND CTS SIGNALS.
8182          ; THIS LOOP WILL CLEAR THE TX.IE AND RX.IE BITS IF THEY ARE SET.
8183 035304 005003          ;*
8184 035306 010300 100:   CLR    R3             ;CLEAR THE LINE COUNTER.
8185 035310 006300          MOV    R3,R0
8186 035312 036067 002344 144710  ASL    R0
8187 035320 001451          BIT    BITTBL(R0),ACTLNS
8188          BEQ    #12#           ;DON'T TEST IF NOT ACTIVE LINE.
8189          ;*
8190          ; SET ALL THE DUT LNCTRL REGISTERS RTS BITS.
8191 035322 012700 010000          ;*
8192 035326 012705 177777          MOV    #BIT12,R0     ;SPECIFY THAT RTS BITS ARE TO BE SET.
8192 035332 004767 161312          MOV    #MAPLNS,R5    ;SPECIFY THAT ALL LNCTRLS ARE TO BE CHANGED.
8192 035332 004767 161312          JSR    PC,WTWLC      ;SET ALL THE DUT RTS BITS.

```

```

8193 035336 012704 000050      MOV    #40.,R4
8194 035342 004767 156364      JSR    PC,DELAY      ;DELAY FOR 40 MS TO ALLOW SIGNALS TO SETTLE.
8195
8196      ;*
8197      ; RECORD THE STATES OF THE MODEM STATUS SIGNALS.
8198 035346 004767 160176      JSR    PC,SAVMST     ;SAVE THE PRESENT MODEM STATUS STATES.
8199
8200      ;*
8201      ; CLEAR THE RTS FOR THE SELECTED LINE.
8202 035352 010377 144660      MOV    R3,@CSRA     ;SELECT THE SELECTED LINE IND.ADR.REG FIELD.
8203 035356 042777 010000 144662  BIC    #BIT12,@LNCTRA ;CLEAR THE SELECTED LINE RTS.
8204 035364 012704 000050      MOV    #40.,R4
8205 035370 004767 156336      JSR    PC,DELAY     ;ALLOW 40 MS FOR STATUS SIGNALS TO STABILIZE.
8206
8207      ;*
8208      ; CHECK THE PRESENT DUT STAT REGISTER CONTENTS AGAINST PREVIOUS.
8209      ; IF ANY UNDESIRED CHANGES HAVE TAKEN PLACE, REPORT THE ERRORS.
8210 035374 116301 004020      MOV    TXRLNB(R3),R1 ;SELECT SPECIAL TREATMENT FOR ASSOCIATED LINE.
8211 035400 012702 014000      MOV    #BIT12:BIT11,R2 ;IGNORE DCD AND CTS ON ASSOCIATED LINE.
8212 035404 004767 156230      JSR    PC,CHPMST    ;COMPARE OLD AND NEW STAT CONTENTS.
8213 035410 103415      BCS    12#          ;SKIP ERROR REPORT IF NO DISCREPANCIES FOUND.
8214      ;REPORT INTERACTIONS FOUND BETWEEN RTS FOR LINE NN AND THE FOLLOWING SIGNALS:
8215 035412 012767 020467 146442  MOV    #8503.,ERRNBR ;SELECT THE ERROR NUMBER.
8216 035420 012767 012100 146440  MOV    #ER8401,ERRBLK ;SELECT THE ERROR PRINT ROUTINE.
8217 035426 012701 010573      MOV    #EM8502,R1   ;SELECT THE RTS ERROR MESSAGES.
8218 035432 104460      ERROR      ;ER1901 USES R1, R2, AND R3 VALUES.
8219      TRAP    C#ERROR
8220
8221      ;*
8222      ; EXIT THE TEST IF EXTENDED ERROR REPORTING HAS NOT BEEN ENABLED
8223 035434 032767 000100 144556  BIT    #BIT06,OPTION ;EXIT WITH TEST FAILURE MESSAGE IF
8224 035442 001404      BEQ    60#         ;NO EXTENDED ERROR REPORTING HAS BEEN REQUESTED
8225      ;DURING THE SOFTWARE QUESTIONS.
8226
8227      ;*
8228      ; SELECT THE NEXT LINE AND LOOP IF NOT ALL POSSIBLE LINES HAVE BEEN HANDLED.
8229 035444 005203 12#      INC    R3          ;SELECT THE NEXT LINE NUMBER.
8230 035446 020327 000020      CMP    R3,#NUMLNS ;TEST FOR ALL LINES DONE.
8231 035452 002715      BLT    10#        ;LOOP IF NOT ALL LINES DONE.
8232
8233 035454 005067 144600 60#      CLR    CTRLCF     ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8234 035460 035460 012700 000340  SETPRI #PRI07     ;DISABLE ALL INTERRUPTS.
8235      MOV    #PRI07,R0
8236 035466 104401      TRAP    C#SPRI
8237 035466      ENDTST
8238 035466 104401      L10051: TRAP    C#ETST

```

```

8238
8239
8240
8241
8242
8243
8244
8245
8246
8247
8248 035470
      035470
8249
8250 035470 000030
8251 035476 012767 000030 144564
8252 035504 012767 177777 144554
8253 035510 016702 144706
8254 035514 012703 002420
8255 035516 020203
8256
8257
8258
8259
8260
8261 035520 012701 011175
8262 035524
      035524 104455
      035526 022125
      035530 011060
      035532 012674
8263
8264 035534 012767 002420 144654
8265
8266 035542 005067 144512
8267 035546
      035546
      035546 104401

```

```

.SBTTL  HARDWARE TEST          - REP8MP -
; * * * * *
; *          - 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
; *
; *          T24::
      TNUM == TNUM + 1      ; INCREMENT THE ASSEMBLY TIME TEST COUNTER.
      MOV  #TNUM,TSTNUM    ; SET UP THE TEST NUMBER. (93)
      MOV  #-1,CTRLCF      ; INDICATE THAT WE ARE IN A TEST.
      MOV  #BMPQB,R2       ; GET THE CONTENTS OF THE POINTER.
      MOV  #BMPQB,R3       ; GET THE START ADDRESS OF THE QUEUE.
      CMP  R2,R3           ; SEE IF THE POINTER HAS MOVED FROM THE BASE.
      BEQ  600             ; 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"
      MOV  #EM9304,R1      ; PASS THE FIRST MESSAGE TO BE REORTED.
      ERDF 9301,EM9301,ER9301 ; >>>> ERROR #9301 <<<<<.
; *
; *          TRAP  C#ERDF
; *          .WORD 9301
; *          .WORD EM9301
; *          .WORD ER9301
; *
      MOV  #BMPQB,BMPQB    ; SET POINTER BACK TO THE BEGINING OF THE QUE.
; *
; *          600: CLR  CTRLCF      ; INDICATE THAT WE ARE NOT WITHIN A TEST.
; *          ENDTST
; *
; *          L10052: TRAP  C#ETST

```

```

8270 ;*****
8271 ;
8272 ;           FVTB.HWQ
8273 ;
8274 ;*****
8276
8277
8278 .SBTTL  HARDWARE PARAMETER CODING SECTION
8279
8280
8281
8282 ;**
8283 ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
8284 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
8285 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
8286 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
8287 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
8288 ; WITH THE OPERATOR.
8289 ;--
8290
8291 035550          BGNHRD
      035550 000027
      035552
                                L#HARD:: .WORD L10053-L#HARD/2
8292
8302 ;DEVICE CSR ADDRESS QUESTION:
8303 035552          GPRMA  HMPTQ1,0,0,160000,177776,YES
      035552 000031
      035554 035630
      035556 160000
      035560 177776
                                .WORD  T#CODE
                                .WORD  HMPTQ1
                                .WORD  T#LOLIM
                                .WORD  T#HILIM
8304 ;DEVICE INTERRUPT VECTOR QUESTION:
8305 035562          GPRMA  HMPTQ2,2,0,40,776,YES
      035562 001031
      035564 035646
      035566 000040
      035570 000776
                                .WORD  T#CODE
                                .WORD  HMPTQ2
                                .WORD  T#LOLIM
                                .WORD  T#HILIM
8306 ;ACTIVE LINES BIT MAP QUESTION:
8307 035572          GPRMD  HMPTQ3,4,0,MAPLNS,0,177777,YES
      035572 002032
      035574 035701
      035576 177777
      035600 000000
      035602 177777
                                .WORD  T#CODE
                                .WORD  HMPTQ3
                                .WORD  MAPLNS
                                .WORD  T#LOLIM
                                .WORD  T#HILIM
8308 ;TYPE OF LOOPBACK QUESTION:
8309 035604          GPRMD  HMPTQ4,6,0,377,1,3,YES
      035604 003032
      035606 035727
      035610 000377
      035612 000001
      035614 000003
                                .WORD  T#CODE
                                .WORD  HMPTQ4
                                .WORD  377
                                .WORD  T#LOLIM
                                .WORD  T#HILIM
8310 ;INTERRUPT BR LEVEL QUESTION:
8311 035616          GPRMD  HMPTQ5,6,0,177400,0,6,YES
      035616 003032
      035620 036005
      035622 177400
      035624 000000
      035626 000006
                                .WORD  T#CODE
                                .WORD  HMPTQ5
                                .WORD  177400
                                .WORD  T#LOLIM
                                .WORD  T#HILIM

```

```

8312
8313
8314 035630
      035630
8315
8322
8323 035630    103    123    122
      035633    040    101    104
      035636    104    122    105
      035641    123    123    072
      035644    040    000
8324 035646    111    116    124
      035651    105    122    122
      035654    125    120    124
      035657    040    126    105
      035662    103    124    117
      035665    122    040    101
      035670    104    104    122
      035673    105    123    123
      035676    072    040    000
8325 035701    101    103    124
      035704    111    126    105
      035707    040    114    111
      035712    116    105    040
      035715    102    111    124
      035720    040    115    101
      035723    120    072    040
      035726    000
8326 035727    124    131    120
      035732    105    040    117
      035735    106    040    114
      035740    117    117    120
      035743    102    101    103
      035746    113    040    050
      035751    061    075    111
      035754    116    124    105
      035757    122    116    101
      035762    114    054    062
      035765    075    110    063
      035770    062    067    067
      035773    054    063    075
      035776    110    063    062
      036001    065    051    072
      036004    000
8327 036005    111    116    124
      036010    105    122    122
      036013    125    120    124
      036016    040    102    122
      036021    040    114    105
      036024    126    105    114
      036027    072    040    000
    
```

ENDHRD

L10053: .EVEN

HWPTQ1: .ASCIZ /CSR ADDRESS: /

HWPTQ2: .ASCIZ /INTERRUPT VECTOR ADDRESS: /

HWPTQ3: .ASCIZ /ACTIVE LINE BIT MAP: /

HWPTQ4: .ASCIZ /TYPE OF LOOPBACK (1=INTERNAL,2=H3277,3=H325):/

HWPTQ5: .ASCIZ /INTERRUPT BR LEVEL: /

.EVEN

8328
8329

```

8332 ;*****
* 8333 ;
8334 ;           FVTA.SWQ
8335 ;
8336 ;*****

8338
8339
8340 .SBTTL  SOFTWARE PARAMETER CODING SECTION
8341
8342 ;**
8343 ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
8344 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
8345 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
8346 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
8347 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
8348 ; WITH THE OPERATOR.
8349 ;--
8350
8351 036032          BGNSFT
      036032 000014
      036034                                .WORD L10054-L$SOFT/2
                                           L$SOFT::

8352
8361 ;UNIT NUMBER PRINTOUT QUESTION:
8362 036034          GPRML  SWPTQ1,0,20,YES
      036034 000130                                .WORD  T$CODE
      036036 036064                                .WORD  SWPTQ1
      036040 000020                                .WORD  20

8363 ;EXTENDED ERROR REPORTING QUESTION:
8364 036042          GPRML  SWPTQ2,0,100,YES
      036042 000130                                .WORD  T$CODE
      036044 036140                                .WORD  SWPTQ2
      036046 000100                                .WORD  100

8365 ;*
8366 ; IF EXTENDED ERROR REPORTING IS NOT REQUIRED THEN SKIP THE NEXT QUESTION.
8367 ;*
8368 036050          XFERF  ENDD
      036050 006044                                .WORD  T$CODE

8369
8370 ;NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE QUESTION:
8371 036052          GPRMD  SWPTQ3,2,0,177777,0,177777,YES
      036052 001052                                .WORD  T$CODE
      036054 036173                                .WORD  SWPTQ3
      036056 177777                                .WORD  177777
      036060 000000                                .WORD  T$LOLIM
      036062 177777                                .WORD  T$HILIM

8372
8373          .EVEN
8374
8375 036064          ENDD:  ENDSFT
      036064                                .EVEN
                                           L10054:

8376
8377
8384 036064          122    105    120          SWPTQ1: .ASCIZ  /REPORT UNIT NUMBER AS EACH UNIT IS TESTED: /
      036067          117    122    124
      036072          040    125    116
      036075          111    124    040

```

	036100	116	125	115
	036103	102	105	122
	036106	040	101	123
	036111	040	105	101
	036114	103	110	040
	036117	125	116	111
	036122	124	040	111
	036125	123	040	124
	036130	105	123	124
	036133	105	104	072
	036136	040	000	
8385	036140	105	130	124
	036143	105	116	104
	036146	105	104	040
	036151	105	122	122
	036154	117	122	040
	036157	122	105	120
	036162	117	122	124
	036165	111	116	107
	036170	072	040	000
8386	036173	116	125	115
	036176	102	105	122
	036201	040	117	106
	036204	040	111	116
	036207	104	111	126
	036212	111	104	125
	036215	101	114	040
	036220	104	101	124
	036223	101	040	105
	036226	122	122	117
	036231	122	123	040
	036234	124	117	040
	036237	122	105	120
	036242	117	122	124
	036245	040	117	116
	036250	040	101	040
	036253	114	111	116
	036256	105	072	040
8387	036261	000		

SWPTQ2: .ASCIZ /EXTENDED ERROR REPORTING: /

SWPTQ3: .ASCIZ /NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE: /

.EVEN

```

8389
8390
8391 :*****
8392 :
8393 :           FVTSKL6.P11
8394 :*****
8395
8396
8397
8398 036262 $PATCH:
8399 036262      .BLKW  24
8400
8407
8408
8409
8410
8411 036332      LASTAD
                                .EVEN
                                .WORD  0
                                .WORD  0
      036332 000000
      036334 000000
8412 036336
8413
8414
8415
8416
8417
8418
8419
8420      000001      .END

```

SYMBOL TABLE	
ACTLNS	002230 G
ADR	= 000020 G
ADRPTR	013556 G
ALTFLD	013066 G
ASLNTL	013140 G
ASSEMB	= 000010
BCOUNT	002330 G
BITTBL	002344 G
BIT0	= 000001 G
BIT00	= 000001 G
BIT01	= 000002 G
BIT02	= 000004 G
BIT03	= 000010 G
BIT04	= 000020 G
BIT05	= 000040 G
BIT06	= 000100 G
BIT07	= 000200 G
BIT08	= 000400 G
BIT09	= 001000 G
BIT1	= 000002 G
BIT10	= 002000 G
BIT11	= 004000 G
BIT12	= 010000 G
BIT13	= 020000 G
BIT14	= 040000 G
BIT15	= 100000 G
BIT2	= 000004 G
BIT3	= 000010 G
BIT4	= 000020 G
BIT5	= 000040 G
BIT6	= 000100 G
BIT7	= 000200 G
BIT8	= 000400 G
BIT9	= 001000 G
BMPCQB	002420 G
BMPCQE	002620 G
BMPCQP	002416 G
BOE	= 000400 G
BRLEVL	002233 G
BUFBAS	002660 G
BUFEND	003660 G
BUFHID	003260 G
BUFPTR	002256 G
BUF3QT	003460 G
CALMSL	013250 G
CHKBMP	013474 G
CKTRAP	013544 G
CLKBRL	002314 G
CLKCSR	002312 G
CLKHRZ	002320 G
CLKINT	016730 G
CLKVEC	002316 G
CLNRST	013574 G
CLR16W	013616 G
CPMST	013640 G
CSRA	002236 G
CSRO	= 000000 G
CTRLCF	002260 G
C#AU	= 000052
C#AUTO	= 000061
C#BRK	= 000022
C#BSEG	= 000004
C#BSUB	= 000002
C#CEFG	= 000045
C#CLCK	= 000062
C#CLEA	= 000012
C#CLOS	= 000035
C#CLP1	= 000006
C#CVEC	= 000036
C#DCLN	= 000044
C#DODU	= 000051
C#DRPT	= 000024
C#DU	= 000053
C#EDIT	= 000003
C#ERDF	= 000055
C#ERHR	= 000056
C#ERRO	= 000060
C#ERSF	= 000054
C#ERSO	= 000057
C#ESCA	= 000010
C#ESEG	= 000005
C#ESUB	= 000003
C#ETST	= 000001
C#EXIT	= 000032
C#GETB	= 000026
C#GETW	= 000027
C#GHAN	= 000043
C#GPHR	= 000042
C#GPLO	= 000030
C#GPRI	= 000040
C#INIT	= 000011
C#INLP	= 000020
C#MANI	= 000050
C#MEM	= 000031
C#MSG	= 000023
C#OPEN	= 000034
C#PNTB	= 000014
C#PNTF	= 000017
C#PNTS	= 000016
C#PNTX	= 000015
C#QIO	= 000377
C#RDSU	= 000007
C#REFG	= 000047
C#RESE	= 000033
C#REVI	= 000003
C#FLA	= 000021
C#RPT	= 000025
C#SEFG	= 000046
C#SPRI	= 000041
C#SVEC	= 000037
C#TPRI	= 000013
DELAY	013732 G
DFPTBL	002206 G
DIAGMC	= 000000
DODMA	013772 G
DRADRT	002236 G
DROP	017726
EDROP	020004
EF.CON	= 000036 G
EF.NEW	= 000035 G
EF.PWR	= 000034 G
EF.RES	= 000037 G
EF.STA	= 000040 G
EF0503	004217 G
EF0505	004224 G
EF1601	004277 G
EF3001	004331 G
EF3002	004400 G
EF6401	004447 G
EF7801	004516 G
EF8401	004554 G
EF8402	004646 G
EF9001	004763 G
EF9002	005045 G
EF9003	005117 G
EF9004	005146 G
EF9005	005176 G
EF9006	005227 G
EF9019	005246 G
EF9301	005265 G
EF9302	005343 G
EM0101	014516 G
EM0102	014602 G
EM0103	005443 G
EM0525	005501 G
EM0526	005571 G
EM1601	005661 G
EM4001	005744 G
EM4002	005776 G
EM4101	006032 G
EM4102	006064 G
EM4103	006120 G
EM4901	006204 G
EM4902	006245 G
EM5001	006277 G
EM5101	006336 G
EM5102	006373 G
EM5103	006431 G
EM5201	006461 G
EM5202	006514 G
EM5301	006552 G
EM5302	006606 G
EM5303	006666 G
EM5401	006737 G
EM5402	007005 G
EM5501	007046 G
EM5601	007112 G
EM5701	007167 G
EM6401	007244 G
EM6402	007301 G
EM6601	007342 G
EM6602	007377 G
EM6701	007451 G
EM6702	007503 G
EM7801	007560 G
EM7802	007622 G
EM7901	007653 G
EM7902	007715 G
EM8001	007746 G
EM8002	010012 G
EM8101	010056 G
EM8102	010121 G
EM8201	010164 G
EM8202	010230 G
EM8301	010274 G
EM8302	010340 G
EM8401	010404 G
EM8402	010466 G
EM8403	010472 G
EM8404	010476 G
EM8405	010501 G
EM8406	010505 G
EM8501	010511 G
EM8502	010573 G
EM9009	010577 G
EM9010	010623 G
EM9017	010647 G
EM9026	010760 G
EM9104	011004 G
EM9301	011060 G
EM9302	011100 G
EM9303	011130 G
EM9304	011175 G
ENDD	036064
ENDET8	003660 G
ENDIT	017646
ERLTBL	002660 G
ERRBLK	004066 G
ERRMSG	004064 G
ERRNBR	004062 G
ERRTYP	004060 G
ER0101	011252 G
ER0503	011604 G
ER1603	011642 G
ER6401	011734 G
ER7801	012042 G
ER8401	012100 G
ER9001	012356 G
ER9002	012456 G
ER9101	012634 G
ER9301	012674 G
EVL	= 000004 G
E#END	= 002100
E#LOAD	= 000035
FDATA	002244 G
FDATE	= 000006 G
FINACT	014062 G
FSLSA	002244 G
FSLSO	= 000006 G
F#AU	= 000015
F#AUTO	= 000020
F#BGN	= 000040
F#CLEA	= 000007
F#DU	= 000016
F#END	= 000041
F#HARD	= 000004
F#HW	= 000013
F#INIT	= 000006
F#JMP	= 000050
F#MOD	= 000000
F#MSG	= 000011
F#PROT	= 000021
F#PWR	= 000017
F#RPT	= 000012
F#SEG	= 000003
F#SOFT	= 000005
F#SRV	= 000010
F#SUB	= 000002
F#SW	= 000014
F#TEST	= 000001
GETPRM	017440
GPRSOB	002404 G
G#CNT0	= 000200
G#DELM	= 000372
G#DISP	= 000003
G#EXCP	= 000400
G#HILI	= 000002
G#LOLI	= 000001
G#NO	= 000000
G#OFFS	= 000400
G#OFSI	= 000376
G#PRMA	= 000001
G#PRMD	= 000002
G#PRML	= 000000
G#RADA	= 000140
G#RADB	= 000000
G#RADD	= 000040
G#RADL	= 000120
G#RADO	= 000020
G#XFER	= 000004
G#YES	= 000010
HELP	= 000000
HOE	= 100000 G
HMPTQ1	035630
HMPTQ2	035646
HMPTQ3	035701
HMPTQ4	035727
HMPTQ5	036005
IBE	= 010000 G
IDU	= 000040 G
IER	= 020000 G
IESTAT	002264 G
INDATP	014142 G
INDTPX	014172 G
ISR	= 000100 G

IXE	=	004000	G	L\$HARD	035552	G	L10036	026550	PRI07	=	000340	G	TXDSBL	016074	G		
I\$AU	=	000041		L\$HIME	002120	G	L10037	027174	PRTLPR	014700	G	TXENBL	016170	G			
I\$AUTO	=	000041		L\$HPCP	002016	G	L10040	027712	PUFIFO	014762	G	TXIEO	016264	G			
I\$CLN	=	000041		L\$HPTP	002022	G	L10041	030656	PUFIFR	015044	G	TXINTC	002300	G			
I\$DU	=	000041		L\$HM	002206	G	L10042	031412	RBUFA	002240	G	TXINTF	002302	G			
I\$HRD	=	000041		L\$ICP	002104	G	L10043	032146	RBUFO	=	000002	G	TXRLNB	004020	G		
I\$INIT	=	000041		L\$INIT	017036	G	L10044	032616	READBX	015246	G	TXRLNE	004040	G			
I\$MOD	=	000041		L\$LADP	002026	G	L10045	033266	RESETT	015330	G	TXRXLB	003760	G			
I\$MSG	=	000041		L\$LAST	036336	G	L10046	033736	RXBDBTX	=	000030	G	TXRXLE	004020	G		
I\$PROT	=	000040		L\$LOAD	002100	G	L10047	034406	RXBETX	=	000020	G	TXVECA	002226	G		
I\$PTAB	=	000041		L\$LUN	002074	G	L10050	035036	RXBFUL	=	000100	G	T\$ARGC	=	000002	G	
I\$PWR	=	000041		L\$MREV	002050	G	L10051	035466	RXCNTB	003720	G	T\$CODE	=	001052			
I\$RPT	=	000041		L\$NAME	002000	G	L10052	035546	RXIEO	015442	G	T\$ERRN	=	022125			
I\$SEG	=	000041		L\$PRIO	002042	G	L10053	035630	RXINTC	002274	G	T\$EXCP	=	000000			
I\$SETU	=	000041		L\$PROT	017030	G	L10054	036064	RXINTF	002276	G	T\$FLAG	=	000050			
I\$SFT	=	000041		L\$PRT	002112	G	MAPCNT	014270	RXTIMO	=	000002	G	T\$GMAN	=	000000		
I\$SRV	=	000041		L\$REPP	002062	G	MAPLNS	=	177777	G	RXTMA	002240	G	T\$HII I	=	177777	
I\$SUB	=	000041		L\$REV	002010	G	MFUNIT	004166	RXVECA	002224	G	T\$LAST	=	000001			
I\$TST	=	000041		L\$RPT	017022	G	MMENAB	002342	ROSLOT	=	000002	G	T\$LOLI	=	000000		
J\$JMP	=	000167		L\$SOFT	036034	G	MMPRES	002340	R1SLOT	=	000004	G	T\$LSYM	=	010000		
LGRP1M	002266	G		L\$SPC	002056	G	MMSRO	002336	R2SLOT	=	000006	G	T\$LTNO	=	000030		
LGRP2M	002270	G		L\$SPCP	002020	G	MSG1	011370	R3SLOT	=	000010	G	T\$NEST	=	177777		
LINBIT	014242	G		L\$SPTP	002024	G	MSG2	011446	R4SLOT	=	000012	G	T\$NSO	=	000000		
LNCTRA	002246	G		L\$STA	002030	G	MSG3	011525	R5SLOT	=	000014	G	T\$NS1	=	000005		
LNCTRO	=	000010	G	L\$SW	002220	G	MSLCNT	002334	SAVBMP	015502	G	T\$PTNU	=	000000			
LOE	=	040000	G	L\$TEST	002114	G	MSLGET	014322	SAVMST	015550	G	T\$SAVL	=	177777			
LOPCK	002232	G		L\$TIML	002014	G	MLOOP	014436	SETPAR	015614	G	T\$SEGL	=	177777			
LOT	=	000010	G	L\$UNIT	002012	G	MSTICK	002332	SFPTBL	002220	G	T\$SUBN	=	000000			
LPCSLT	=	000036	G	L10000	002216		NDERPT	002222	G	SKPSTS	015662	G	T\$TAGL	=	177777		
LPRA	002242	G		L10001	002224		NEWPAS	017420	STGTRB	004040	G	T\$TAGN	=	010055			
LPRO	=	000004	G	L10002	011366		NEWRES	017412	STSTB	002620	G	T\$TEMP	=	000000			
L\$ACP	002110	G		L10003	011640		NEWSTA	017102	STSTE	002660	G	T\$TEST	=	000030			
L\$APT	002036	G		L10004	011732		NUMLNS	=	000020	G	SVCGBL	=	000000				
L\$AU	020012	G		L10005	012040		OOPS	014452	G	SVCINS	=	000001					
L\$AUT	002070	G		L10006	012076		OPTION	002220	G	SVCSUB	=	000001					
L\$AUTO	017662	G		L10007	012354		O\$APTS	=	000000		SVCTAG	=	000001				
L\$CCP	002106	G		L10010	012454		O\$AU	=	000000		SVCTST	=	000001				
L\$CLEA	017664	G		L10011	012632		O\$BGNR	=	000001		SMPTQ1	036064					
L\$CO	002032	G		L10012	012672		O\$BGNS	=	000001		SMPTQ2	036140					
L\$DEPO	002011	G		L10013	013064		O\$DU	=	000001		SMPTQ3	036173					
L\$DESC	004140	G		L10014	017026		O\$ERRT	=	000001		S\$LSYM	=	010000				
L\$DESC	002076	G		L10016	017660		O\$GNSW	=	000001		TIMER1	002322	G				
L\$DEVP	002060	G		L10017	017662		O\$POIN	=	000001		TIMER2	002324	G				
L\$DISP	002124	G		L10020	017700		O\$SETU	=	000000		TIMER3	002326	G				
L\$DLY	002116	G		L10021	020010		PASCNT	002272	G	TNUM	=	000030	G				
L\$DTP	002040	G		L10022	020016		PCSLT	=	000016	G	TP4FLG	002306	G				
L\$DTP	002034	G		L10023	020300		PNT	=	001000	G	TP4RTN	017000	G				
L\$DU	017702	G		L10024	020704		PREGRT	004112	G	TP4VEC	002304	G					
L\$DUT	002072	G		L10025	021342		PREGO5	004070		TSABRT	015740	G					
L\$DVTY	004130	G		L10026	022142		PRI	=	002000	G	TSTNUM	002262	G				
L\$EF	002052	G		L10027	022742		PRI00	=	000000	G	TXAD1A	002250	G				
L\$ENVI	002044	G		L10030	023364		PRI01	=	000040	G	TXAD10	=	000012	G			
L\$ERRT	004060	G		L10031	024056		PRI02	=	000100	G	TXAD2A	002252	G				
L\$ETP	002102	G		L10032	024364		PRI03	=	000140	G	TXAD20	=	000014	G			
L\$EXP1	002046	G		L10033	024674		PRI04	=	000200	G	TXBFCA	002254	G				
L\$EXP4	002064	G		L10034	025374		PRI05	=	000240	G	TXBFCO	=	000016	G			
L\$EXP5	002066	G		L10035	026072		PRI06	=	000300	G	TXDATP	016052	G				

SYMBOL TABLE

T19	032620 G	T24	035470 G	T8	024060 G	WAIBIS	016534 G	X\$ALWA=	000000
T2	020302 G	T3	020706 G	T9	024366 G	WAITTX	016610 G	X\$FALS=	000040
T20	033270 G	T4	021344 G	UAM	000200 G	WORD1	002310 G	X\$OFFS=	000400
T21	033740 G	T5	022144 G	UNITN	002234 G	WTWLNC	016650 G	X\$TRUE=	000020
T22	034410 G	T6	022744 G	UNSDIV	016324 G	WTWLPR	016700 G	\$PATCH	036262 G
T23	035040 G	T7	023366 G	WAIBIC	016460 G				

. ABS. 03633E 000
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 28661 WORDS (112 PAGES)

DYNAMIC MEMORY: 20060 WORDS (77 PAGES)

ELAPSED TIME: 00:04:01

PARTB.BIN,PARTB.LST/-SP-SVC34R/ML,PARTB.P11