

DHV11

DHV-11 FUNC TST PT3
CVDHCAO

AH-T657A-MC
FICHE 1 OF 2

OCT 1983
COPYRIGHT © 1983
MADE IN USA



A large grid of approximately 15 columns and 20 rows of small, illegible data entries. Each cell contains a small table or set of data points, likely representing test results or functional data for various components. The text is too small to be read accurately.



DHV11

DHV-11 FUNC TST PT3
CVDHCAO

AH-T657A-MC
FICHE 2 OF 2

OCT 1983
COPYRIGHT © 1983
MADE IN USA



Table with multiple columns and rows of data, likely a test log or functional test results. The data is too small to read accurately but appears to be organized in a grid format.



CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 2
CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-T656A-MC
PRODUCT NAME: CVDHCAO DHV-11 FUNC TST PART3
PRODUCT DATE: 31 OCTOBER 1983
MAINTAINER: EDSHE - DIAGNOSTICS GROUP
AUTHOR: BERT KLEINSCHMIDT
TONY GRIMSHAW
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 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 3
PROGRAM DOCUMENT

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

ORIGINAL RELEASE: 31-OCT-83 BERT KLEINSCHMIDT

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 HIERARCY PREREQUISITES
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 (YXDP+)
3.0	ERROR INFORMATION
3.1	TYPES OF ERROR MESSAGES
3.2	ERROR MESSAGES
4.0	PERFORMANCE AND PROGRESS REPORTS
5.0	TEST SUMMARIES
6.0	EXAMPLE ERROR FREE PASS

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 5
CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

1.0 GENERAL PROGRAM CONSIDERATIONS

1.1 PROGRAM ABSTRACT

CVDHC IS PART ONE OF THE DHV-11 FUNCTIONAL VERIFICATION TEST. THIS PART OF THE TEST VERIFIES THAT THE MAJOR COMMUNICATION FUNCTIONS OF THE BOARD WHICH USE THE UARTS ARE FUNCTIONING CORRECTLY. THIS PROGRAM EXERCISES THE BOARD BY TRANSMITTING AND RECEIVING LARGE BLOCKS OF DATA IN LOOPBACK.

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

1.2 SYSTEM REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE DHV FVT:

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

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

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

2.0 OPERATING INSTRUCTIONS

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 6
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

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 BE RUN. ALL OTHER TESTS WILL NOT BE RUN.

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 7
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

/PASS:DDDD EXECUTE DDDDD PASSES (DDDD = 1 TO 64000)
 /FLAGS:FLGS SET SPECIFIED FLAGS.SEE THE FLAGS SECTION
 OF THIS DOCUMENT.
 /EOP:DDDD REPORT END OF PASS MESSAGE AFTER EVERY
 DDDDD PASSES ONLY. (DDDD = 1 TO 64000)
 /UNITS:LIST TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED
 IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12
 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

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

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

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

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

2.3 FLAGS

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

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBR*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 8
PROGRAM DOCUMENT

IXR*	ERROR TYPE, NUMBER, PC, TEST AND UNIT)
	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

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 9
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

NON-ENDING EXECUTION. IN THIS CASE, EXIT FROM THE PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR BY OCCURANCE OF AN ERROR WITH THE HALT ON ERROR FLAG BEING SET. THE EXIT IS A RETURN TO COMMAND MODE. SEE EXAMPLE AT END OF 'EFFECT OF START COMMAND' SECTION.

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

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

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

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

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

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

2.4.1.5 EFFECT OF START COMMAND -

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

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 10
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

QUESTION '# UNITS (D) ?' TO WHICH THE OPERATOR SHOULD REPLY WITH THE NUMBER OF UNITS TO BE TESTED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES ARE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE COMPLETE UNIT. EACH QUESTION IS FOLLOWED BY THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE AFTER THE PARENTHESES. FOR THE ACTUAL HARDWARE P-TABLE QUESTIONS SEE THE 'HARDWARE PARAMETERS' SECTION.

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

EXAMPLE:

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

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

2.4.2 RESTART COMMAND -

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

2.4.2.1 TESTS, PASS, AND FLAGS SWITCHES -

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

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

2.4.2.3 EFFECT OF RESTART COMMAND -

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

2.4.3 CONTINUE COMMAND -

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

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

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

2.4.3.2 EFFECT OF CONTINUE COMMAND -

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

2.4.4 PROCEED COMMAND -

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

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

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

2.4.4.2 EFFECT OF PROCEED COMMAND -

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE.

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

2.4.5 ADD COMMAND -

ADD/UNITS:<UNIT-LIST>

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

2.4.7 DROP COMMAND -

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

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

2.4.9 PRINT COMMAND -

PRI(NT)

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

2.4.10 DISPLAY COMMAND -

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

2.4.10.1 EFFECT OF DISPLAY COMMAND -

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

2.4.11 FLAGS COMMAND -

FLA(GS)

2.4.11.1 EFFECT OF FLAGS COMMAND -

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

2.4.12 ZFLAGS COMMAND -

ZFL(AGS)

2.4.13 ZFLAGS COMMAND -

ALL FLAGS ARE CLEARED.

2.4.14 CONTROL CHARACTERS -

- C A CONTROL/C (C) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO COMMAND MODE.
- Z A CONTROL/Z (Z) ENTERED DURING ONE OF THE TWO OPERATOR DIALOGUES-- HARDWARE P-TABLE DIALOGUE OR SOFTWARE P-TABLE DIALOGUE CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.
- O A CONTROL/O (O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES ALL TELETYPE OUTPUT TO BE SURPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER CONTROL/O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

2.5 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER 'Y' AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN 'PRELOADED' USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 14
CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A 'Y', THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT:

1. CSR ADDRESS - THIS QUESTION REQUESTS THE CSR ADDRESS OF THE SPECIFIED DHV11.
2. VECTOR ADDRESS - THIS QUESTION REQUESTS THE INTERRUPT VECTOR ADDRESS OF THE SPECIFIED DHV1 .
3. ACTIVE LINES BIT MAP - THIS QUESTION REQUESTS AN OCTAL BIT MAP OF THE SERIAL COMMUNICATION LINES ON THE DHV11 WHICH ARE BEING SELECTED FOR TESTING. IF THE BIT IN THE BIT MAP IS SET WHICH CORRESPONDS TO A PARTICULAR LINE (I.E. BIT 3 FOR LINE 3) THAT LINE WILL BE TESTED BY THE FVT. WITH STAGGERED LOOPBACK A PAIR OF LINES WITH THE SPECIFIED TRANSMIT LINE AND ANOTHER RECEIVE LINE WILL BE TESTED. THEREFORE, TO GUARANTEE THAT BOTH THE TRANSMITTER AND RECEIVER OF A SPECIFIED LINE ARE TESTED WHEN USING THE STAGGERED LOOPBACK CONNECTOR, BOTH THE INTENDED LINE AND ITS MATE MUST BE SELECTED (IE. TO TEST LINE 1, SELECT BOTH LINE 1 AND LINE 3). IN NONSTAGGERED TESTING, A BIT IN THE ACTIVE LINES BIT MAP SELECTS THE TRANSMITTER AND RECEIVER FOR THE SAME LINE.
4. TYPE OF LOOPBACK (1=INTERNAL, 2=STAGGERED, 3=H325) - THIS QUESTION REQUESTS THE TYPE OF LOOPBACK TO BE USED IN TESTING THE DHV11. THE FOLLOWING TYPES OF LOOPBACK ARE SUPPORTED:
 - 0 INTERNAL - ONLY INTERNAL UART LOOPBACK IS TO BE USED IN TESTING THE DHV.
 - 0 STAGGERED - STAGGERED BERG CONNECTOR(S) ARE INSTALLED ON THE BERG CONNECTOR SOCKETS OF THE DHV11. FOR THE CIRCUIT CONNECTIONS OF THE STAGGERED LOOPBACK CONNECTOR SEE THE HARDWARE SECTION OF THIS DOCUMENT.
 - 0 H325 - SINGLE LINE, 25 PIN LOOPBACK CONNECTORS (TYPE H325) ARE INSTALLED ON THE LINES TO BE TESTED. THESE CONNECTORS CAN BE INSTALLED ON THE DISTRIBUTION PANEL OR ON THE END OF THE TERMINAL OR MODEM CABLE. THE H325 CONNECTORS MUST HAVE THE REMOVABLE JUMPERS INSTALLED.
5. BR LEVEL - THIS QUESTIONS REQUESTS THE INTERRUPT BR LEVEL OF THE DHV11.

2.6 SOFTWARE QUESTIONS

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

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 15
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

1. REPORT UNIT NUMBER AS EACH UNIT IS TESTED - THIS QUESTION ASKS WHETHER THE PROGRAM SHOULD REPORT THE NUMBER OF THE UNIT WHICH IT IS TESTING AS IT BEGINS TO TEST EACH UNIT.
2. NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE - THIS QUESTION ASKS FOR THE NUMBER OF DATA ERRORS WHICH SHOULD BE REPORTED INDIVIDUALLY BY THIS PROGRAM FOR EACH LINE FOR EACH TRANSMISSION TEST. ERRORS WHICH ARE NOT REPORTED INDIVIDUALLY ARE REPORTED IN SUMMARY ERROR REPORTS.

2.7 EXTENDED P-TABLE DIALOGUE

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

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

UNITS (D) ? 8<CR>

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

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

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

UNIT 4
 CSR ADDRESS (O) ? 160000<CR>
 SUB-DEVICE # (O) ? 3<CR>
 Q-FACTOR (O) 0 ? <CR>

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

UNIT 6

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 16
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

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

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

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

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

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

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

UNITS (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

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 17
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

SINCE THEY WERE ONLY SPECIFIED ONCE THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

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

UNITS (D) ? 8<CR>

UNIT 1
 CSR ADDRESS (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 AND THE QUESTION IS ASKED) QUESTIONS
3. TYPE 'R NAME', WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE 'START'
5. ANSWER THE 'CHANGE HW' QUESTION WITH 'Y'
6. ANSWER ALL THE HARDWARE QUESTIONS
7. ANSWER THE 'CHANGE SW' QUESTION WITH 'N'

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

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

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

THE GENERAL ERROR MESSAGE IS OF THE FORM:

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

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 18
 CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

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

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

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

3.2 ERROR MESSAGES

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

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

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

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

CVDHC DVC FTL ERR 01603 ON UNIT 02 TST 015 SUB 000 PC: 015244
 DEVICE REGISTER WORD READ/WRITE TEST
 BAD BIT(S) IN DEVICE TBUFFAD1 REGISTER FOR LINE 7 (D).
 EXPECTED DATA: 000000 (0).
 ACTUAL DATA: 000023 (0).

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

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 19
CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

DOCUMENT.

5.0 TEST SUMMARIES

THE FOLLOWING TESTS ARE INCLUDED WITHIN CVDHC:

1. DEVICE REGISTER ADDRESS TEST - VERIFIES THAT THE UUT REGISTERS WILL RESPOND WITH THE PROPER Q-BUS HANDSHAKING WHEN ACCESSED. VERIFIES THAT THE UUT IS AT THE PROPER ADDRESS.
2. FRAMING.ERROR TEST - VERIFIES THAT FORCED FRAMING ERRORS ARE REPORTED CORRECTLY.
3. PARITY.ERROR TEST - VERIFIES THAT FORCED PARITY ERRORS ARE REPORTED CORRECTLY.
4. BREAK TEST - VERIFIES THAT SETTING THE BREAK BIT ON ANY LINE CAUSES THAT LINE TO GO TO A SPACING STATE AND THAT CLEARING THE BREAK BIT REMOVES THE LINE FROM THE SPACING STATE.
5. NO OVERRUN.ERROR TEST - VERIFIES THAT THE UUT WILL RECEIVE THE MAXIMUM NUMBER OF CHARACTERS WITHOUT CAUSING AN OVERRUN ERROR.
6. OVERRUN.ERROR TEST - VERIFIES THAT IF MORE THAN THE MAXIMUM NUMBER OF CHARACTERS ARE SENT TO THE UUT OVERRUN ERRORS OCCUR.
7. SINGLE CHARACTER MODE TX/RX TEST - VERIFIES THAT THE UUT WILL TX AND RX DATA CORRECTLY IN SINGLE CHARACTER MODE.
8. DMA MODE TX/RX TEST - VERIFIES THAT THE UUT WILL TX AND RX DATA CORRECTLY USING DMA TRANSMISSION.
9. SPLIT SPEED TEST - VERIFIES THAT THE UUT WILL WORK WITH DIFFERENT TX AND RX SPEEDS ON EACH ACTIVE LINE.
10. REPORT BMP CODES TEST - THIS PSEUDO TEST REPORTS THE FIRST 32 BMP CODES WHICH WERE DISCOVERED IN THE FIFO DURING THE EXECUTION OF THE OTHER TESTS. THIS AVOIDS THE INTERRUPTION OF OTHER TESTS BY THESE CODES, IF THEY ARE NOT CRITICAL TO THE TESTS BEING PERFORMED.

6.0 EXAMPLE ERROR FREE PASS

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

.R CVDHCAO
CVDHCAO.BIC

DRS
CVDHC-A-0
DHV-11 FUNC TST PART3

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 20
CVDHCA.P11 12-JUL-83 11:44 PROGRAM DOCUMENT

UNIT 1; DHV-11
RESTART ADDR: 147670
DR>STA

CHANGE HW (L) ? Y

UNITS (D) ? 2

UNIT 0
CSR ADDRESS. (0) 160020 ? ^Z

UNIT 1
CSR ADDRESS: (0) 160020 ? 160040
INTERRUPT VECTOR ADDRESS: (0) 300 ? 320
ACTIVE LINE BIT MAP: (0) 377 ? <CR>
TYPE OF LOOPBACK (1=INTERNAL OR NONE, 2=STAGGERD,
3=25 PIN CONNECTOR): (0) 2 ? 1
INTERRUPT BR LEVEL: (0) 4? <CR>

CHANGE SE (L) ? Y

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

TESTING UNIT : 0

TESTING UNIT : 1

CVDHC EOP 1
0 CUMULATIVE ERRORS

TESTING UNIT : 0

^C
DR> EXIT

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 21
PROGRAM DOCUMENT

```

985      .LIST  CFQ,LOC,BIN,MEB
986      .NLIST  CND
987
988
989
990      .SBTTL  PROGRAM HEADER
991
992
993      .MCALL  SVC
994 000000' SVC ; INITIALIZE SUPERVISOR MACROS
995      000001 SVCINS= 1 ; LIST INSTRUCTIONS, SHIFTED RIGHT
996      000001 SVCTST= 1 ; LIST TEST TAGS, SHIFTED RIGHT
997      000001 SVCSUB= 1 ; LIST SUBTEST TAGS, SHIFTED RIGHT
998      000001 SVCGBL= 1 ; LIST GLOBAL TAGS, SHIFTED RIGHT
999      000001 SVCTAG= 1 ; LIST OTHER TAGS, SHIFTED RIGHT
1000
1001      .ENABL  ABS
1002      .ENABL  AMA
1003      002000 = 2000
1004
1005      002000 BGNMOD
1006
1007      :++
1008      : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
1009      : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
1010      :--
1011
1012      002000 POINTER BGNRPT,BGNSW,BGNSFT,BGNDU,ERRTBL
1013
1014
1015      002000 HEADER CVDHC,A,0,230,0,PR107
1016      002000
1017      002000 103
1018      002001 126
1019      002002 104
1020      002003 110
1021      002004 103
1022      002005 000
1023      002006 000
1024      002007 000
1025      002010
1026      002010 101
1027      002011
1028      002011 060
1029      002012
1030      002012 000000
1031      002014
1032      002014 000230
1033      002016
1034      002016 034274
1035      002020
1036      002020 034632
1037      002022
1038      002022 002152
1039      002024
1040      002024 002164

```

```

LSNAME::
        .ASCII /C/
        .ASCII /V/
        .ASCII /D/
        .ASCII /H/
        .ASCII /C/
        .BYTE 0
        .BYTE 0
        .BYTE 0
LSREV::
        .ASCII /A/
LSDEPO::
        .ASCII /0/
LSUNIT::
        .WORD 0
LSTIML::
        .WORD 230
LSHPCP::
        .WORD LSHARD
LSSPCP::
        .WORD LSSOFT
LSHPTP::
        .WORD LSHW
LSSPTP::
        .WORD LSSW

```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 22
PROGRAM HEADER

1041	002026	
1042	002026	035072
1043	002030	
1044	002030	000000
1045	002032	
1046	002032	000000
1047	002034	
1048	002034	000000
1049	002036	
1050	002036	000000
1051	002040	
1052	002040	002124
1053	002042	
1054	002042	000340
1055	002044	
1056	002044	000000
1057	002046	
1058	002046	000000
1059	002050	
1060	002050	003
1061	002051	003
1062	002052	
1063	002052	000000
1064	002054	000000
1065	002056	
1066	002056	000000
1067	002060	
1068	002060	005272
1069	002062	
1070	002062	025652
1071	002064	
1072	002064	000000
1073	002066	
1074	002066	000000
1075	002070	
1076	002070	000000
1077	002072	
1078	002072	026540
1079	002074	
1080	002074	000000
1081	002076	
1082	002076	005302
1083	002100	
1084	002100	104035
1085	002102	
1086	002102	005222
1087	002104	
1088	002104	025666
1089	002106	
1090	002106	026522
1091	002110	
1092	002110	026520
1093	002112	
1094	002112	025660
1095	002114	
1096	002114	000000

LSLADP::		
LSSTA::	.WORD	LSLAST
LSCO::	.WORD	0
LSDTYP::	.WORD	0
LSAPT::	.WORD	0
LSDTP::	.WORD	0
LS RIO::	.WORD	LSDISPATCH
LSENV1::	.WORD	PRI07
LSEXP1::	.WORD	0
LSMREV::	.WORD	0
	.BYTE	CSREVISION
	.BYTE	CS\$EDIT
LSEF::		
	.WORD	0
	.WORD	0
LSSPC::	.WORD	0
LSDEVP::	.WORD	0
LSREPP::	.WORD	LS\$VTYP
LSEXP4::	.WORD	LSRPT
LSEXP5::	.WORD	0
LSAUT::	.WORD	0
LSDUT::	.WORD	0
LSLUN::	.WORD	LSDU
LSDESP::	.WORD	0
LSLOAD::	.WORD	LSDESC
	EMT	ES\$LOAD
LSETP::	.WORD	LSERRTBL
LSICP::	.WORD	LSINIT
LSCCP::	.WORD	LSCLEAN
LSACP::	.WORD	LSAUTO
LSPRT::	.WORD	LS\$PROT
LS\$TEST::	.WORD	0

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 23
PROGRAM HEADER

1097 002116
1098 002116 000000
1099 002120
1100 002120 000000
1:01

LSPLY:: .WORD 0
LSHIME:: .WORD 0

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 24
DISPATCH TABLE

.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 10

1102		
1103		
1104		
1105		
1106		
1107		
1108		
1109	002122	
1110	002122	000012
1111	002124	
1112	002124	026656
1113	002126	027146
1114	002130	027530
1115	002132	030124
1116	002134	030550
1117	002136	031254
1118	002140	032210
1119	002142	032724
1120	002144	033552
1121	002146	034212
1122		

	.WORD	10
L\$DISPATCH::		
	.WORD	T1
	.WORD	T2
	.WORD	T3
	.WORD	T4
	.WORD	T5
	.WORD	T6
	.WORD	T7
	.WORD	T8
	.WORD	T9
	.WORD	T10

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 25
DISPATCH TABLE

1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
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.
:--

BGNHW DFPTBL

.WORD L10000-L\$HW/2

L\$HW::
DFPTBL::

.WORD 160020 ;DEFAULT CSR ADDRESS
.WORD 300 ;DEFAULT VECTOR ADDRESS
.WORD 177777 ;DEFAULT ACTIVE LINES BIT MAP
.BYTE 2 ;DEFAULT LOOPBACK MODE
.BYTE 4 ;DEFAULT BR LEVEL

ENDHW

L10000:

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 26
DEFAULT HARDWARE P-TABLE

1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167

.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.
:--

BGNSW SFPTBL

.WORD L10001-L\$SW/2

L\$SW::
SFPTBL::

002162 000002

002164
002164

OPTION:: .WORD 20
NDERPT:: .WORD 0

:BIT MAP OF PROGRAM CONTROL FLAGS
:DEFAULT NUMBER OF INDIVIDUAL DATA ERRORS TO RPT

002164 000020
002166 000000

ENDSW

L10001:

002170
002170

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 27
SOFTWARE P-TABLE

```

1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179      000010
1180      000377
1181
1182
1183      000000
1184      000002
1185      000002
1186      000004
1187      000006
1188      000010
1189      000012
1190      000014
1191      000016
1192
1193
1194      000020
1195      000030
1196      000100
1197
1198
1199      002170
1200
1201
1202
1203      100000
1204      040000
1205      020000
1206      010000
1207      004000
1208      002000
1209      001000
1210      000400
1211      000200
1212      000100
1213      000040
1214      000020
1215      000010
1216      000004
1217      000002
1218      000001
1219
1220      001000
1221      000400
1222      000200
1223      000100

      .SBTTL GLOBAL EQUATES SECTION

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

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

      ;***** DEVICE REGISTER OFFSETS FROM THE CSR'S ADDRESS *****
      CSRO==0         ;CSR REGISTER OFFSET FROM THE CSR ADDRESS
      RBUFO==2        ;RECEIVE REGISTER OFFSET FROM THE CSR ADDRESS
      TXCHRO==2       ;TRANSMIT REGISTER OFFSET FROM THE CSR ADDRESS
      LPRO==4         ;LINE PARAMETER REGISTER OFFSET FROM THE CSR ADDRESS
      STATO==6        ;STATUS REGISTER OFFSET FROM THE CSR ADDRESS
      LNCTRO==10      ;LINE CONTROL REGISTER OFFSET FROM THE CSR ADDRESS
      TXAD10==12      ;TRANSMIT ADDRESS 1 REGISTER OFFSET FROM THE CSR ADDRESS
      TXAD20==14      ;TRANSMIT ADDRESS 2 REGISTER OFFSET FROM THE CSR ADDRESS
      TXBFCO==16      ;TRANSMIT COUNT REGISTER OFFSET FROM THE CSR ADDRESS

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

      EQUALS

      ; BIT DIFINITIONS
      BIT15== 100000
      BIT14== 40000
      BIT13== 20000
      BIT12== 10000
      BIT11== 4000
      BIT10== 2000
      BIT09== 1000
      BIT08== 400
      BIT07== 200
      BIT06== 100
      BIT05== 40
      BIT04== 20
      BIT03== 10
      BIT02== 4
      BIT01== 2
      BIT00== 1

      BIT9== BIT09
      BIT8== BIT08
      BIT7== BIT07
      BIT6== BIT06
    
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 28
GLOBAL EQUATES SECTION

```

1224      000040      BIT5== BIT05
1225      000020      BIT4== BIT04
1226      000010      BIT3== BIT03
1227      000004      BIT2== BIT02
1228      000002      BIT1== BIT01
1229      000001      BIT0== BIT00
1230      ;
1231      ; EVENT FLAG DEFINITIONS
1232      ; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
1233      ;
1234      000040      EF.START==      32.      ; START COMMAND WAS ISSUED
1235      000037      EF.RESTART==     31.      ; RESTART COMMAND WAS ISSUED
1236      000036      EF.CONTINUE==    30.      ; CONTINUE COMMAND WAS ISSUED
1237      000035      EF.NEW==        29.      ; A NEW PASS HAS BEEN STARTED
1238      000034      EF.PWR==        28.      ; A POWER-FAIL/POWER-UP OCCURRED
1239      ;
1240      ;
1241      ; PRIORITY LEVEL DEFINITIONS
1242      ;
1243      000340      PRI07== 340
1244      000300      PRI06== 300
1245      000240      PRI05== 240
1246      000200      PRI04== 200
1247      000140      PRI03== 140
1248      000100      PRI02== 100
1249      000040      PRI01== 40
1250      000000      PRI00== 0
1251      ;
1252      ; OPERATOR FLAG BITS
1253      ;
1254      000004      EVL==          4
1255      000010      LOT==         10
1256      000020      ADR==         20
1257      000040      IDU==         40
1258      000100      ISR==        100
1259      000200      UAM==        200
1260      000400      BOE==        400
1261      001000      PNT==       1000
1262      002000      PRI==       2000
1263      004000      IXE==       4000
1264      010000      IBE==      10000
1265      020000      IER==      20000
1266      040000      LOE==      40000
1267      100000      HOE==     100000
1268

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 29
GLOBAL EQUATES SECTION

1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324

.SBTTL GLOBAL DATA SECTION

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

: UNIT VARIABLE AREA

RXVECA:: .WORD 300 ;RX VECTOR ADDRESS.
TXVECA:: .WORD 304 ;TX VECTOR ADDRESS.
ACTLNS:: .WORD 377 ;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::
CSRA:: .WORD 160000 ;DHV-11 CSR ADDRESS
TXCHA:: RBUFA:: .WORD 160002 ;DHV-11 RECEIVE/TRANSMIT BUFFER ADDRESS
LPRA:: .WORD 160004 ;DHV-11 LINE PARAMETER REGISTER ADDRESS
STATA:: .WORD 160006 ;DHV-11 STATUS REGISTER ADDRESS
LNCTRA:: .WORD 160010 ;DHV-11 LINE CONTROL REGISTER ADDRESS
TXAD1A:: .WORD 160012 ;DHV-11 TRANSMIT BUFFER 1 REGISTER ADDRESS
TXAD2A:: .WORD 160014 ;DHV-11 TRANSMIT BUFFER 2 REGISTER ADDRESS
TXBFCA:: .WORD 160016 ;DHV-11 TRANSMIT BUFFER COUNT REGISTER ADDRESS

: ASSORTED GLOBAL VARIABLES:

CTRLCF:: .WORD 0 ;STORAGE FOR THE CONTROL-C FLAG.
TSTNUM:: .WORD 1 ;STORAGE FOR THE TEST NUMBER.
IBM:: .WORD 0 ;INACTIVE TX/RX BITS MASK.
LGRP1M:: .WORD 31463 ;BIT MAP OF LINES IN LINE GROUP I.
LGRP2M:: .WORD 146314 ;BIT MAP OF LINES IN LINE GROUP II.
IESTAT:: .WORD 0 ;STORAGE FOR STATES OF THE DUT INT ENABLE BITS.
PASCNT:: .WORD 0 ;STO'G FOR PASS COUNT USED IN ROM VERSION# TST.
WORD1:: .WORD 0 ;LOCATION FOR PASSING INDIRECT PARAMETERS.
RXTOUT:: .WORD 0 ;TIME-OUT VALUE FOR WAITING FOR LAST RX CHAR.
SAVTEN:: .WORD 0 ;STORAGE FOR TX.ENABLE STATES, (TXROFF, TXRON).
SAVPRI:: .WORD 0 ;STO'G FOR PROCESSOR PRIORITY, (TXROFF, TXRON).
TXENBM:: .WORD 0 ;STORAGE FOR TX.ENABLE STATES, (BUFFER MGM'NT).
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.

: LINE TIME CLOCK VARIABLES AND STORAGE.

CLKCSR:: .WORD 177546 ;CSR ADDRESS OF THE LTC.
CLKBRL:: .WORD PRI06 ;INTERRUPT PRIORITY LEVEL OF THE LTC.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 30
GLOBAL DATA SECTION

1325	002264	000100	CLKVEC:: .WORD	100	: INTERRUPT VECTOR ADDRESS OF THE LTC.
1326	002266	000074	CLKHRZ:: .WORD	60.	: INTERRUPT FREQUENCY OF THE LTC.
1327	002270	000000	TIMER1:: .WORD	0	: HARDWARE CLOCK COUNTER #1.
1328	002272	000000	TIMER2:: .WORD	0	: HARDWARE CLOCK COUNTER #2.
1329	002274	000170	TIMER3:: .WORD	120.	: HARDWARE BREAK COUNTER LOCATION.
1330	002276	000170	BCOUNT:: .WORD	120.	: BREAK COUNT VALUE IN CLOCK TICKS.
1331	002300	000021	MSTICK:: .WORD	17.	: NUMBER OF MILLI-SECONDS PER LTC TICK.
1332	002302	000062	MSLCNT:: .WORD	62	: LOOP COUNT (USED BY MSLOOP) TO DELAY 1 MS.

```

1333
1334
1335
1336
:*****
:      MEMORY MANAGEMENT VARIABLES AND FLAGS.
:*****

```

1337	002304	177572	MMSRO:: .WORD	177572	: ADDRESS OF MEM MGT STATUS REGISTER #0.
1338	002306	000000	MMPRES:: .WORD	0	: MEM MGT PRESENT FLAG (0 IF MM NOT PRESENT).
1339	002310	000000	MMENAB:: .WORD	0	: MEM MGT ENABLED FLAG (0 IF MM NOT ENABLED).

```

1340
1341 002312
1342 002312 172340
1343 002314 172342
1344 002316 172344
1345 002320 172346
1346 002322 172350
1347 002324 172352
1348 002326 172354
1349 002330 172356
1350 002332

```

PARATB::					: BASE OF MEM MGT PAR ADDRESS TABLE.
	PAR0A::	.WORD	172340		: ADDRESS OF MEM MGT PAR #0.
	PAR1A::	.WORD	172342		: ADDRESS OF MEM MGT PAR #1.
	PAR2A::	.WORD	172344		: ADDRESS OF MEM MGT PAR #2.
	PAR3A::	.WORD	172346		: ADDRESS OF MEM MGT PAR #3.
	PAR4A::	.WORD	172350		: ADDRESS OF MEM MGT PAR #4.
	PAR5A::	.WORD	172352		: ADDRESS OF MEM MGT PAR #5.
	PAR6A::	.WORD	172354		: ADDRESS OF MEM MGT PAR #6.
	PAR7A::	.WORD	172356		: ADDRESS OF MEM MGT PAR #7.

```

1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369

```

PARATE::					: END OF PAR ADDRESS TABLE.
:*****					
: 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.

```

1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
:*****
:      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.

```

:*****

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 31
GLOBAL DATA SECTION

```

1381      ;*      TRANSMISSION AND RECEPTION VARIABLES, POINTERS, AND FLAGS.
1382      ;*****
1383 002404 000000  CHRTOT:: .WORD 0      ;TOTAL RECEIVED CHARACTER COUNTER.
1384 002406 000000  ERSMRF:: .WORD 0      ;'PRINT ERROR SUMMARY' FLAGS.
1385 002410 000000  TXDNF:: .WORD 0      ;TRANSMISSION DONE FLAGS.
1386 002412 000000  RXDNF:: .WORD 0      ;RECEPTION DONE FLAGS.
1387 002414 000000  TXDBLF:: .WORD 0     ;'TX HAS BEEN DISABLED' FLAG.
1388      ;*****
1389      ;      STORAGE AREA FOR THE BMP CODE QUEUE.
1390      ;*****
1391 002416 000000  BMPCPQ:: .WORD 0      ;POINTER USED TO ACCESS THE NEXT CELL IN QUE.
1392 002420 000100  BMPCQB:: .BLKW 64.    ;STORAGE FOR 32 CELLS, TEST# PLUS BMP CODE.
1393 002620      BMPCQE::      ;LAST ADDRESS PLUS 2 OF THE BMP CODE QUEUE.
1394      ;*****
1395      ;*      RECEIVE BUFFER AND ASSOCIATED VARIABLES.
1396      ;*****
1397 002620 000000  RXBOPT:: .WORD 0     ;RX BUFFER OUTPUT POINTER.
1398 002622 000000  RXBIPT:: .WORD 0     ;RX BUFFER INPUT POINTER.
1399 002624 000000  RXBCNT:: .WORD 0     ;COUNT OF NUMBER OF CHARS IN RX BUFFER.
1400 002626      RXBSTA::      ;LABEL AT BEGINNING OF THE RX BUFFER.
1401 002626 000100      ;LEAVE ENOUGH ROOM FOR A FULL BUFFER.
1402 003026 000000  RXBEND:: .WORD 0     ;LABEL AFTER END OF RX BUFFER.
1403      ;*****
1404      ;*      TX/RX CONTROL BLOCK.
1405      ;*****
1406 003030      CBB::      ;BASE OF TX/RX CONTROL BLOCK.
1407 003030 000000  CBLPRA:: .WORD 0     ;LINE PARAMETER REGISTER CONTENTS.
1408 003032 000000  CBLNCA:: .WORD 0     ;LINE CONTROL REGISTER CONTENTS.
1409 003034 000000  CBDPAA:: .WORD 0     ;START ADDRESS OF DATA PATTERN.
1410 003036 000000  CBDPLA:: .WORD 0     ;LENGTH OF DATA PATTERN.
1411 003040 000000  CBDPNA:: .WORD 0     ;NUMBER OF REPEAT TRANSMISSIONS OF THE DATA PATTERN.
1412 003042 000000  CBMAPA:: .WORD 0     ;BIT MAP OF LINES TO INITIALISE.
1413 003044 000000  CBLPBA:: .WORD 0     ;LOOPBACK MODE (AS IN LOPBCK).
1414 003046 000000  CBOFSA:: .WORD 0     ;AMOUNT OF OFFSET BETWEEN EACH TX START.
1415      ;*****
1416      ;*      TRANSMISSION AND RECEPTION TABLES OF POINTERS AND COUNTERS.
1417      ;*****
1418 003050 000020  DPENDB:: .BLKW 16.    ;TABLE OF END ADDRESSES OF DATA PATTERNS.
1419 003110 000020  DPLENB:: .BLKW 16.    ;TABLE OF LENGTH OF DATA PATTERNS FOR LINES.
1420 003150 000020  EXCNTB:: .BLKW 16.    ;EXTRA RECEIVED CHARACTER COUNTERS TABLE.
1421 003210 000020  ERCNTB:: .BLKW 16.    ;CHARACTER RECEIVE ERROR COUNTERS TABLE.
1422 003250 000020  TXPTRB:: .BLKW 16.    ;TRANSMISSION DATA POINTERS TABLE.
1423 003310 000020  RXPTRB:: .BLKW 16.    ;RECEPTION DATA POINTERS TABLE.
1424 003350 000020  CHCNTB:: .BLKW 16.    ;NUMBER OF CHARACTERS TO BE TXED AND RXED.
1425 003410 000020  TXCNTB:: .BLKW 16.    ;TRANSMISSION CHARACTER COUNTERS TABLE.
1426 003450 000020  RXCNTB:: .BLKW 16.    ;RECEPTION CHARACTER COUNTERS TABLE.
1427      ;*****
1428      ;      GENERAL TABLE AND BUFFER AREA--513 WORDS.
1429      ;*****
1430 003510      BUFBAS::      ;BASE OF MEMORY BUFFER.
1431 003510 000200  ERLTBL:: .BLKW 128.   ;FIRST HALF OF GENERAL TABLE OR BUFFER.
1432 004110 000100  BUFMID:: .BLKW 64.    ;SECOND HALF OF GENERAL TABLE OR BUFFER.
1433 004310 000100  BUF3QT:: .BLKW 64.    ;LAST QUARTER OF THE BUFFER AREA.
1434 004510      BUFEND::      ;END OF GENERAL PURPOSE MEMORY BUFFER.
1435 004510 000020  ENDETB:: .BLKW 16.    ;BUFFER OVERFLOW SPACE.
1436      ;*****

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 32
GLOBAL DATA SECTION

1437
1438
1439 004550
1440 004550 000004
1441 004560 000004
1442 004570 000004
1443 004600 000004
1444 004610 000004
1445 004620 000004
1446 004630 000004
1447 004640 000004
1448 004650 000004
1449 004660 000004
1450 004670 000004
1451 004700 000004
1452 004710 000004
1453 004720 000004
1454 004730 000004
1455 004740 000004
1456 004750
1457
1458
1459
1460 004750
1461 004750 000000
1462 004752 000010
1463 004754 000020
1464 004756 000030
1465 004760
1466 004760
1467 004760 000000
1468 004762 073400
1469 004764 177400
1470 004766
1471 004766
1472 004766 000000
1473 004770 000200
1474 004772
1475 004772
1476 004772 000000
1477 004774 000040
1478 004776 000140
1479 005000
1480
1481
1482
1483
1484
1485 005000
1486 005000 156470
1487 005002 167070
1488 005004 177470
1489 005006
1490
1491
1492

```

:          TABLE OF DATA PATTERN RESYNC QUEUES.
:*****
DPRSQB::
      .BLKW 4      ;DATA PATTERN RESYNC QUEUES TABLE BASE.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 0.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 1.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 2.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 3.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 4.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 5.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 6.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 7.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 8.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 9.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 10.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 11.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 12.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 13.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 14.
      .BLKW 4      ;DATA PATTERN RESYNC QUEUE FOR LINE 15.
DPRSQE::
      ;END OF DATA PATTERN RESYNC QUEUES TABLE.
:*****
:          SINGLE CHARACTER MODE LPR FIELD TABLES.
:*****
SCBCTB::
      .WORD 0      ;BASE OF NUMBER OF BITS PER CHAR FIELDS TABLE.
      .WORD 10     ;5 BITS/CHAR LPR FIELD.
      .WORD 20     ;6 BITS/CHAR LPR FIELD.
      .WORD 30     ;7 BITS/CHAR LPR FIELD.
      .WORD 30     ;8 BITS/CHAR LPR FIELD.
SCBCTE::
      ;END OF NUMBER OF BITS/CHAR FIELDS TABLE.
SCBRTB::
      .WORD 0      ;BASE OF BAUDRATE FIELDS TABLE.
      .WORD 73400  ;50 BAUD LPR FIELDS.
      .WORD 177400 ;1.2K BAUD LPR FIELDS.
      .WORD 177400 ;38.4K BAUD LPR FIELDS.
SCBRTE::
      ;END OF BAUDRATE FIELDS TABLE.
SCNSTB::
      .WORD 0      ;BASE OF NUMBER OF STOP BITS FIELDS TABLE.
      .WORD 200    ;1 STOP BIT LPR FIELD.
      .WORD 200    ;2 STOP BITS LPR FIELD.
SCNSTE::
      ;END OF BAUDRATE FIELDS TABLE.
SCTPTB::
      .WORD 0      ;BASE OF TYPE OF PARITY FIELDS TABLE.
      .WORD 40     ;NO PARITY LPR FIELD.
      .WORD 140    ;ODD PARITY LPR FIELD.
      .WORD 140    ;EVEN PARITY LPR FIELD.
SCTPTE::
      ;END OF TYPE OF PARITY FIELDS TABLE.
:*****
:          DMA MODE LPR FIELD TABLES.
:          SET UP WITH SPECIFIED BAUDRATES, 1 STOP BIT, ODD PARITY, 8 BITS/CHAR.
:*****
DLPRTB::
      .WORD 156470 ;BASE OF DMA TEST LPR FIELDS TABLE.
      .WORD 167070 ;9.6K BAUD.
      .WORD 177470 ;19.2K BAUD.
      .WORD 177470 ;38.4K BAUD.
DLP RTE::
      ;END OF DMA TEST LPR FIELDS TABLE.
:*****
:          SPLIT SPEED LPR PARAMETER TABLE.
:*****
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 33
GLOBAL DATA SECTION

1493	005006	
1494	005006	170070
1495	005010	007470
1496	005012	000001
1497	005014	000120
1498	005016	070470
1499	005020	013470
1500	005022	000001
1501	005024	000016
1502	005026	115070
1503	005030	124470
1504	005032	000001
1505	005034	000002
1506	005036	

```

SPLPRB::
      .WORD 170070 ;BASE OF SPLIT SPEED LPR TABLE.
      .WORD 7470   ;TX: 38.4K, RX: 50 BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 1      ;TX: 50, RX: 38.4K BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 80     ;NUMBER OF REPEAT TRANSMISSIONS AT 50 BAUD.
      .WORD 70470 ;NUMBER OF REPEAT TRANSMISSIONS AT 38.4K BAUD.
      .WORD 13470 ;TX: 1200, RX: 75 BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 1      ;TX: 75, RX: 1200 BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 16     ;NUMBER OF REPEAT TRANSMISSIONS AT 75 BAUD.
      .WORD 115070;NUMBER OF REPEAT TRANSMISSIONS AT 1200 BAUD.
      .WORD 124470;TX: 2000, RX:2400 BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 1      ;TX: 2400, RX:2000 BAUD, 1 STOP ODD PAR 8 BITS.
      .WORD 2      ;NUMBER OF REPEAT TRANSMISSIONS AT 2400 BAUD.
      .WORD 2      ;NUMBER OF REPEAT TRANSMISSIONS AT 2000 BAUD.

```

1507		
1508		
1509		
1510	005036	000
1511	005037	001
1512	005040	010
1513	005041	017
1514	005042	063
1515	005043	074
1516	005044	125
1517	005045	177
1518	005046	200
1519	005047	252
1520	005050	303
1521	005051	314
1522	005052	360
1523	005053	367
1524	005054	376
1525	005055	377
1526	005056	
1527	005056	000
1528	005057	001
1529	005060	010
1530	005061	017

```

SPLPRE::
:*****
:      SINGLE CHARACTER DATA PATTERN TABLE.
:*****
SDPBAS::.BYTE 0 ;START OF SINGLE CHARACTER DATA PATTERN TABLE.
      .BYTE 1
      .BYTE 10
      .BYTE 17
      .BYTE 63
      .BYTE 74
      .BYTE 125
      .BYTE 177
      .BYTE 200
      .BYTE 252
      .BYTE 303
      .BYTE 314
      .BYTE 360
      .BYTE 367
      .BYTE 376
      .BYTE 377
SDPEND::
      .BYTE 0 ;END OF SINGLE CHARACTER DATA PATTERN TABLE.
      .BYTE 1 ;START OF FIRST SHORT DATA PATTERN OVERFLOW AREA.
      .BYTE 10
      .BYTE 17

```

1531		
1532		
1533		
1534		
1535	005062	125
1536	005063	252
1537	005064	124
1538	005065	253
1539	005066	122
1540	005067	255
1541	005070	112
1542	005071	265
1543	005072	052
1544	005073	325
1545	005074	152
1546	005075	225
1547	005076	132
1548	005077	245

```

:*****
:      SINGLE CHARACTER DATA PATTERN TABLE NUMBER TWO.
:*****
SDP2B::.BYTE 125 ;START OF SECOND SHORT DATA PATTERN.
      .BYTE 252
      .BYTE 124
      .BYTE 253
      .BYTE 122
      .BYTE 255
      .BYTE 112
      .BYTE 265
      .BYTE 52
      .BYTE 325
      .BYTE 152
      .BYTE 225
      .BYTE 132
      .BYTE 245

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 34
GLOBAL DATA SECTION

1549	005100	126	.BYTE	126
1550	005101	251	.BYTE	251
1551	005102			
1552	005102	125	.BYTE	125
1553	005103	252	.BYTE	252
1554	005104	124	.BYTE	124
1555	005105	253	.BYTE	253
1556	005106	122	.BYTE	122
1557	005107	255	.BYTE	255
1558	005110	112	.BYTE	112
1559	005111	265	.BYTE	265
1560	005112	052	.BYTE	52
1561	005113	325	.BYTE	325
1562	005114	152	.BYTE	152
1563	005115	225	.BYTE	225
1564	005116	132	.BYTE	132
1565	005117	245	.BYTE	245
1566	005120	126	.BYTE	126
1567	005121	251	.BYTE	251

SDP2E::

:END OF SECOND SHORT DATA PATTERN.
:START OF SECOND SHORT DATA PATTERN OVERFLOW AREA.

: SINGLE CHARACTER SAFE PROPORTIONAL DELAY TABLE.

1571	005122	372	PROTBL::	.BYTE	250.	:DELAY IN MILLI SECONDS AT 50 BAUD
1572	005123	252		.BYTE	170.	:DELAY IN MILLI SECONDS AT 75 BAUD
1573	005124	167		.BYTE	119.	:DELAY IN MILLI SECONDS AT 110 BAUD
1574	005125	143		.BYTE	99.	:DELAY IN MILLI SECONDS AT 134.5 BAUD
1575	005126	132		.BYTE	90.	:DELAY IN MILLI SECONDS AT 150 BAUD
1576	005127	062		.BYTE	50.	:DELAY IN MILLI SECONDS AT 300 BAUD
1577	005130	036		.BYTE	30.	:DELAY IN MILLI SECONDS AT 600 BAUD
1578	005131	024		.BYTE	20.	:DELAY IN MILLI SECONDS AT 1200 BAUD
1579	005132	021		.BYTE	17.	:DELAY IN MILLI SECONDS AT 1800 BAUD
1580	005133	020		.BYTE	16.	:DELAY IN MILLI SECONDS AT 2000 BAUD
1581	005134	017		.BYTE	15.	:DELAY IN MILLI SECONDS AT 2400 BAUD
1582	005135	015		.BYTE	13.	:DELAY IN MILLI SECONDS AT 4800 BAUD
1583	005136	014		.BYTE	12.	:DELAY IN MILLI SECONDS AT 7200 BAUD
1584	005137	014		.BYTE	12.	:DELAY IN MILLI SECONDS AT 9600 BAUD
1585	005140	013		.BYTE	11.	:DELAY IN MILLI SECONDS AT 19200 BAUD
1586	005141	012		.BYTE	10.	:DELAY IN MILLI SECONDS AT 38400 BAUD
1587				.EVEN		

:* 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.

1594	005142		TXRXLB::			:BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
1595	005142	000000		.WORD	0	:TX/RX LINE OFFSET FOR RX/TX LINE 0.
1596	005144	000002		.WORD	2.	:TX/RX LINE OFFSET FOR RX/TX LINE 1.
1597	005146	000004		.WORD	4.	:TX/RX LINE OFFSET FOR RX/TX LINE 2.
1598	005150	000006		.WORD	6.	:TX/RX LINE OFFSET FOR RX/TX LINE 3.
1599	005152	000010		.WORD	8.	:TX/RX LINE OFFSET FOR RX/TX LINE 4.
1600	005154	000012		.WORD	10.	:TX/RX LINE OFFSET FOR RX/TX LINE 5.
1601	005156	000014		.WORD	12.	:TX/RX LINE OFFSET FOR RX/TX LINE 6.
1602	005160	000016		.WORD	14.	:TX/RX LINE OFFSET FOR RX/TX LINE 7.
1603	005162	000020		.WORD	16.	:TX/RX LINE OFFSET FOR RX/TX LINE 8.
1604	005164	000022		.WORD	18.	:TX/RX LINE OFFSET FOR RX/TX LINE 9.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 35
GLOBAL DATA SECTION

1605 005166 000024
1606 005170 000026
1607 005172 000030
1608 005174 000032
1609 005176 000034
1610 005200 000036
1611 005202

.WORD 20.
.WORD 22.
.WORD 24.
.WORD 26.
.WORD 28.
.WORD 30.

:TX/RX LINE OFFSET FOR RX/TX LINE 10.
:TX/RX LINE OFFSET FOR RX/TX LINE 11.
:TX/RX LINE OFFSET FOR RX/TX LINE 12.
:TX/RX LINE OFFSET FOR RX/TX LINE 13.
:TX/RX LINE OFFSET FOR RX/TX LINE 14.
:TX/RX LINE OFFSET FOR RX/TX LINE 15.
:END OF TX/RX LINE NUMBER ASSOCIATION TABLE.
:GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.

TXRXLE::

.EVEN

: * 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.

1612
1613
1614
1615
1616
1617
1618
1619

STGTRB::

.BYTE 4.
.BYTE 6.
.BYTE 0
.BYTE 2.
.BYTE 12.
.BYTE 14.
.BYTE 8.
.BYTE 10.
.BYTE 20.
.BYTE 22.
.BYTE 16.
.BYTE 18.
.BYTE 28.
.BYTE 30.
.BYTE 24.
.BYTE 26.

:BASE OF STAGGERED TX/RX LINE NUMBER TABLE.
:TX/RX LINE OFFSET FOR RX/TX LINE 0.
:TX/RX LINE OFFSET FOR RX/TX LINE 1.
:TX/RX LINE OFFSET FOR RX/TX LINE 2.
:TX/RX LINE OFFSET FOR RX/TX LINE 3.
:TX/RX LINE OFFSET FOR RX/TX LINE 4.
:TX/RX LINE OFFSET FOR RX/TX LINE 5.
:TX/RX LINE OFFSET FOR RX/TX LINE 6.
:TX/RX LINE OFFSET FOR RX/TX LINE 7.
:TX/RX LINE OFFSET FOR RX/TX LINE 8.
:TX/RX LINE OFFSET FOR RX/TX LINE 9.
:TX/RX LINE OFFSET FOR RX/TX LINE 10.
:TX/RX LINE OFFSET FOR RX/TX LINE 11.
:TX/RX LINE OFFSET FOR RX/TX LINE 12.
:TX/RX LINE OFFSET FOR RX/TX LINE 13.
:TX/RX LINE OFFSET FOR RX/TX LINE 14.
:TX/RX LINE OFFSET FOR RX/TX LINE 15.
:GUARANTEE THAT NEXT TABLE IS ON WORD BOUNDARY.

1620 005202
1621 005202 004
1622 005203 006
1623 005204 000
1624 005205 002
1625 005206 014
1626 005207 016
1627 005210 010
1628 005211 012
1629 005212 024
1630 005213 026
1631 005214 020
1632 005215 022
1633 005216 034
1634 005217 036
1635 005220 030
1636 005221 032

.EVEN
ERRTBL

LSERRTBL::

1638 005222
1639 005222
1640 005222 000000
1641 005224 000000
1642 005226 000000
1643 005230 000000
1644
1645

ERRTYP:: .WORD 0
ERRNBR:: .WORD 0
ERRMSG:: .WORD 0
ERRBLK:: .WORD 0

.EVEN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 36
GPR HANDLING ROUTINES FOR SUBROUTINE CALLS.

1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682

```

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

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 37
GPR FRAME ACCESS EQUATES

.SBTTL GPR FRAME ACCESS EQUATES

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

1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697

000036
000016
000014
000012
000010
000006
000004
000002

LPCSLT== 36 :OFFSET FOR LAST RETURN PC.
PCSLT== 16 :OFFSET FOR RETURN PC.
R5SLOT== 14 :OFFSET FOR R5.
R4SLOT== 12 :OFFSET FOR R4.
R3SLOT== 10 :OFFSET FOR R3.
R2SLOT== 6 :OFFSET FOR R2.
R1SLOT== 4 :OFFSET FOR R1.
ROSLOT== 2 :OFFSET FOR R0.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 38
GLOBAL MACRO DEFINITION - SAVE -

1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721

```

.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

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 39
GLOBAL MACRO DEFINITION - PASS -

1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769

```

.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,@(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
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
.NLIST
.ENDM PASS

```


CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 40
GLOBAL SUBROUTINE - PREG05 -

1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822

```

.SBTTL GLOBAL SUBROUTINE - PREG05 -
*****
* PRESERVE REGISTERS R0 THROUGH R5 FOR SUBROUTINE CALLS.
*
* INPUTS: THE RETURN ADDRESS BACK INTO THE CALLING ROUTINE MUST BE IN
* GPR R5. (I.E.- MACROS USE "JSR R5,PREG05".)
*
* OUTPUTS: REGISTERS R0 THROUGH R5 ARE SAVED ON THE STACK.
*
* CALLING SEQUENCE: SAVE ;MACRO EXPANSION CALLS PREG05.
* [SUBROUTINE CODE] ;
* PASS ;MACRO EXPANSION RECALLS PREG05.
*
* COMMENTS: THIS ROUTINE IS RE-ENTRANT.
*
* PARAMETERS MAY BE PASSED OUT OF A SUBROUTINE BY MODIFYING THE
* REGISTER SAVE AREA ON THE STACK. USE THE PASS GPRN MACRO
* TO RETURN GPR VALUES INTACT.
* USE THE RNSLOT OFFSETS FROM THE SP TO PASS OTHER PARAMETERS.
* [EXAMPLE: MOV VALUE,R0SL0T(SP) ]
* MAKE SURE THE SP IS AT ITS ORIGINAL VALUE WHEN YOU DO THIS.
*
* SUBORDINATE ROUTINES CALLED: NONE.
*****
PREG05: ;R5 HAS BEEN LOADED ON THE STACK BY THE SUBROUTINE CALL
MOV R4,-(SP) ;SAVE R4
MOV R3,-(SP) ;SAVE R3
MOV R2,-(SP) ;SAVE R2
MOV R1,-(SP) ;SAVE R1
MOV R0,-(SP) ;SAVE R0
MOV R5,-(SP) ;PUSH RETURN PC ON TOP OF STACK
MOV R5SL0T(SP),R5 ;RESTORE R5 TO VALUE IT HAD BEFORE CALLS
JSR PC,@(SP)+ ;CALL THE SUBROUTINE AT THE RETURN ADDRESS
;FROM THE PREG05 CALL, PUTTING THE PRESENT
;PC ON THE STACK AS A RETURN ADDRESS INTO
;THIS (PREG05) ROUTINE.

;+++
;THE FOLLOWING CODE IS EXECUTED WHEN THE CALLING ROUTINE DOES A
;RETURN [JSR PC,@(SP)+] USING THE PC DEPOSITED ON THE STACK ABOVE.
;---
PREGRT: MOV (SP)+,R5 ;PUT RETURN PC IN R5.
MOV (SP)+,R0 ;RESTORE R0.
MOV (SP)+,R1 ;RESTORE R1.
MOV (SP)+,R2 ;RESTORE R2.
MOV (SP)+,R3 ;RESTORE R3.
MOV (SP)+,R4 ;RESTORE R4.
RTS R5 ;RETURN TO THE SUBROUTINE WHICH CALLED PREG05.
;RESTORING R5 IN THE PROCESS.

```

005232
005232 010446
005234 010346
005236 010246
005240 010146
005242 010046
005244 010546
005246 016605 000014

005252 004736

005254 012605
005256 012600
005260 012601
005262 012602
005264 012603
005266 012604

005270 000205

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 41
CVDHCA.P11 12-JUL-83 11:44 GLOBAL TEXT SECTION

1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853

.SBTTL GLOBAL TEXT SECTION

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

:
: NAMES OF DEVICES SUPPORTED BY PROGRAM

:
: DEV1YP <DHV-11>

LSDVTYP::
.ASCIZ /DHV-11/
.EVEN

: TEST DESCRIPTION

:
: DESCRIPT <DHV-11 FUNCT TEST PART3>

L\$DESC::
.ASCIZ /DHV-11 FUNCT TE

.EVEN

.EVEN

005272
005272
005272 044104 026526 030461
005300 000
005302
005302
005302
005302 044104 026526 030461
005310 043040 047125 052103
005316 052040 051505 020124
005324 040520 052122 000063

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 42
GLOBAL TEXT SECTION

1854
1855
1856
1857
1858
1859
1860
1861

:
: FORMAT STATEMENTS USED IN PRINT CALLS
:

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 43
CVDHCA.P11 12-JUL-83 11:44 GLOBAL TEXT SECTION

```
1862
1863 .NLIST BIN
1864 .SBTTL GLOBAL MESSAGE AREA
1865 : ***** FORMAT STATEMENTS *****
1866 005332 MFUNIT:: .ASCIZ /%N% TESTING UNIT :%D4%N/
1867 005340
1868 005346
1869 005354
1870 005362
1871 005363 EF0503:: .ASCIZ /%T%N/
1872 005370 EF1601:: .ASCIZ /%A %T% ABORTED %N/
1873 005376
1874 005404
1875 005412
1876 005414 EF1603:: .ASCIZ /%A ACTUAL DATA: %06%A (0).%N/
1877 005422
1878 005430
1879 005436
1880 005444
1881 005452
1882 005456 EF6201:: .ASCIZ \%A FRAMING/PARITY ERROR DETECTION AND REPORTING BAD ON LINES:%D2%A : %D2%N\
1883 005464
1884 005472
1885 005500
1886 005506
1887 005514
1888 005522
1889 005530
1890 005536
1891 005544
1892 005552
1893 005560
1894 005566
1895 005571 EF6202:: .ASCIZ /%A CHAR RECEIVED WITH FRAMING ERROR BIT %T%A, SHOULD BE %T%N/
1896 005576
1897 005604
1898 005612
1899 005620
1900 005626
1901 005634
1902 005642
1903 005650
1904 005656
1905 005664
1906 005667 EF6203:: .ASCIZ /%A CHAR RECEIVED WITH PARITY ERROR BIT %T%A, SHOULD BE %T%N/
1907 005674
1908 005702
1909 005710
1910 005716
1911 005724
1912 005732
1913 005740
1914 005746
1915 005754
1916 005762
1917 005764 EF6401:: .ASCIZ /%A %D2%N/
```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 44
CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

1918 005772
1919 006000
1920 006006
1921 006014
1922 006022
1923 006030
1924 006033 EF7801:: .ASCIZ /%T% ON LINE %D2% DECIMAL.%N/
1925 006040
1926 006046
1927 006054
1928 006062
1929 006070
1930 006071 EF9001:: .ASCIZ /%A UNEXPECTED %T% FOUND IN RECEIVE CHAR FIFO:%N/
1931 006076
1932 006104
1933 006112
1934 006120
1935 006126
1936 006134
1937 006142
1938 006150
1939 006153 EF9002:: .ASCIZ /%A CODE IS ASSOCIATED WITH LINE: %D2%N/
1940 006160
1941 006166
1942 006174
1943 006202
1944 006210
1945 006216
1946 006224
1947 006225 EF9003:: .ASCIZ /%A CODE IS: %03%N/
1948 006232
1949 006240
1950 006246
1951 006254 EF9004:: .ASCIZ /%A %T% VALUE: %03%N/
1952 006262
1953 006270
1954 006276
1955 006304 EF9005:: .ASCIZ /%A %T% VALUE: NONE%N/
1956 006312
1957 006320
1958 006326
1959 006334
1960 006335 EF9006:: .ASCIZ /%A %T% %D2%N/
1961 006342
1962 006350
1963 006354 EF9007:: .ASCIZ /%A CHARACTER RECEIVED WITH ERROR FLAG(S) SET ON LINE %D2%N/
1964 006362
1965 006370
1966 006376
1967 006404
1968 006412
1969 006420
1970 006426
1971 006434
1972 006442
1973 006450 EF9008:: .ASCIZ /%A CHARACTER READ AS: %03%N/

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 45
 CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

1974 006456
 1975 006464
 1976 006472
 1977 006500
 1978 006506
 1979 006507 EF9009:: .ASCIZ /%A %TXA ERROR FLAG SET.%N/
 1980 006514
 1981 006522
 1982 006530
 1983 006536
 1984 006544
 1985 006546 EF9010:: .ASCIZ /%A NUMBER OF ERRORS DETECTED ON LINE %D2%A IS %D5%A/
 1986 006554
 1987 006562
 1988 006570
 1989 006576
 1990 006604
 1991 006612
 1992 006620
 1993 006626
 1994 006634
 1995 006635 EF9012:: .ASCII /%A LINE%D2%A ONLY %TX%D5%A BYTES OF%D5%A BYTE/
 1996 006642
 1997 006650
 1998 006656
 1999 006664
 2000 006672
 2001 006700
 2002 006706
 2003 006711 .ASCIZ / DATA PAT'N TX'D FROM LINE%D2%A/
 2004 006716
 2005 006724
 2006 006732
 2007 006740
 2008 006746
 2009 006751 EF9013:: .ASCIZ /%A DATA PATTERN NOT COMPLETELY %TX%A/
 2010 006756
 2011 006764
 2012 006772
 2013 007000
 2014 007006
 2015 007014
 2016 007016 EF9019:: .ASCIZ /%A %TXA %O6%A/
 2017 007024
 2018 007032
 2019 007035 EF9020:: .ASCIZ /%A TOO FEW TX.ACTIONS GENERATED ON LINE %D2%A/
 2020 007042
 2021 007050
 2022 007056
 2023 007064
 2024 007072
 2025 007100
 2026 007106
 2027 007114
 2028 007116 EF9101:: .ASCIZ /%N/
 2029 007121 EF9103:: .ASCIZ /%A ERROR CONDITION ON LINE %D2%A/

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 46
 CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

```

2030 007126
2031 007134
2032 007142
2033 007150
2034 007156
2035 007164
2036 007167 EF9301:: .ASCIZ /%A %T%D2%A, BMP CODE REPORTED :%03%N/
2037 007174
2038 007202
2039 007210
2040 007216
2041 007224
2042 007232
2043 007235 EF9302:: .ASCIZ /%A OVERFLOW OCCURRED (MORE THAN 31 BMP CODES FOUND IN QUEUE)%N/
2044 007242
2045 007250
2046 007256
2047 007264
2048 007272
2049 007300
2050 007306
2051 007314
2052 007322
2053 007330
2054          :***** MESSAGE AREA *****
2055 007335 EM0103:: .ASCIZ /DEVICE REGISTER ACCESS ERRORS/
2056 007342
2057 007350
2058 007356
2059 007364
2060 007372
2061 007373 EM0509:: .ASCIZ /SET/
2062 007377 EM1601:: .ASCIZ /TIMEOUT OCCURRED WAITING FOR MASTER RESET TO CLEAR/
2063 007404
2064 007412
2065 007420
2066 007426
2067 007434
2068 007442
2069 007450
2070 007456
2071 007462 EM5303:: .ASCIZ /BMP CODE FOUND IN FIFO, TEST INVAILEDATED/
2072 007470
2073 007476
2074 007504
2075 007512
2076 007520
2077 007526
2078 007533 EM6201:: .ASCIZ /FRAMING ERROR TEST /
2079 007540
2080 007546
2081 007554
2082 007557 EM6202:: .ASCIZ /CLEAR /
2083 007564
2084 007566 EM6301:: .ASCIZ /PARITY ERROR TEST /
2085 007574

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 47
CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

2086 007602
2087 007610
2088 007611 EM6401:: .ASCIZ /BREAK GENERATION TEST /
2089 007616
2090 007624
2091 007632
2092 007640 EM6402:: .ASCIZ / BREAK NOT RECEIVED ON LINE(S):/
2093 007646
2094 007654
2095 007662
2096 007670
2097 007676
2098 007701 EM6601:: .ASCIZ /NO OVERRUN ERROR TEST/
2099 007706
2100 007714
2101 007722
2102 007727 EM6602:: .ASCIZ / OVERRUN ERROR REPORTED WHEN NONE FORCED/
2103 007734
2104 007742
2105 007750
2106 007756
2107 007764
2108 007772
2109 010000
2110 010001 EM6701:: .ASCIZ /OVERRUN ERROR TEST/
2111 010006
2112 010014
2113 010022
2114 010024 EM6702:: .ASCIZ / NO OVERRUN ERROR REPORTED, OVERRUN FORCED/
2115 010032
2116 010040
2117 010046
2118 010054
2119 010062
2120 010070
2121 010076
2122 010101 EM9001:: .ASCIZ /SINGLE CHARACTER MODE TEST /
2123 010106
2124 010114
2125 010122
2126 010130
2127 010135 EM9003:: .ASCIZ /MODEM STATUS CODE/
2128 010142
2129 010150
2130 010156
2131 010157 EM9004:: .ASCIZ /SELFTEST CODE/
2132 010164
2133 010172
2134 010175 EM9006:: .ASCIZ /CHARACTER RECEIVED ON INACTIVE LINE, LINE:/
2135 010202
2136 010210
2137 010216
2138 010224
2139 010232
2140 010240
2141 010246

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 48
CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

2142 010250 EM9007:: .ASCIZ /UNEXPECTED CHAR RECEIVED AFTER RX COMPLETE ON LINE/
2143 010256
2144 010264
2145 010272
2146 010300
2147 010306
2148 010314
2149 010322
2150 010330
2151 010333 EM9008:: .ASCIZ /RECEIVED CHAR MISCOMPARE AGAINST TX DATA ON LINE/
2152 010340
2153 010346
2154 010354
2155 010362
2156 010370
2157 010376
2158 010404
2159 010412
2160 010414 EM9009:: .ASCIZ /EXPECTED OR CORRECT/
2161 010422
2162 010430
2163 010436
2164 010440 EM9010:: .ASCIZ /ACTUAL OR MEASURED /
2165 010446
2166 010454
2167 010462
2168 010464 EM9011:: .ASCIZ /OVERRUN/
2169 010472
2170 010474 EM9012:: .ASCIZ /FRAMING/
2171 010502
2172 010504 EM9013:: .ASCIZ /PARITY/
2173 010512
2174 010513 EM9014:: .ASCIZ /SUMMARY REPORTS FOR LINES WITH EXCESSIVE NUMBERS OF ERRORS:/
2175 010520
2176 010526
2177 010534
2178 010542
2179 010550
2180 010556
2181 010564
2182 010572
2183 010600
2184 010606
2185 010607 EM9015:: .ASCIZ /TRANSMITTED/
2186 010614
2187 010622
2188 010623 EM9016:: .ASCIZ /RECV'D/
2189 010630
2190 010632 EM9017:: .ASCII / FIFO WILL NOT PURGE (DATA.VALID STUCK SET),/
2191 010640
2192 010646
2193 010654
2194 010662
2195 010670
2196 010676
2197 010704

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 49
CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

2198 010707 .ASCIZ / REMAINDER OF TEST SKIPPED./
2199 010714
2200 010722
2201 010730
2202 010736
2203 010743 EM9025:: .ASCIZ /MORE THAN TWICE THE EXPECTED NUMBER OF CHARACTERS RECEIVED./
2204 010750
2205 010756
2206 010764
2207 010772
2208 011000
2209 011006
2210 011014
2211 011022
2212 011030
2213 011036
2214 011037 EM9026:: .ASCIZ / LPR CONTENTS: /
2215 011044
2216 011052
2217 011060
2218 011063 EM9027:: .ASCIZ /EXTRA CHAR RECEIVED WITHIN DATA PATTERN ON LINE/
2219 011070
2220 011076
2221 011104
2222 011112
2223 011120
2224 011126
2225 011134
2226 011142
2227 011143 EM9028:: .ASCIZ /SINGLE CHAR MISSING FROM RECEIVED DATA ON LINE/
2228 011150
2229 011156
2230 011164
2231 011172
2232 011200
2233 011206
2234 011214
2235 011222 EM9030:: .ASCIZ /%A (NO TX COMPLETION INTERRUPTS RECEIVED)%N/
2236 011230
2237 011236
2238 011244
2239 011252
2240 011260
2241 011266
2242 011274
2243 011277 EM9101:: .ASCIZ /DMA TRANSMISSION MODE TEST /
2244 011304
2245 011312
2246 011320
2247 011326
2248 011333 EM9102:: .ASCIZ /DMA_START BIT SET AFTER RESET OR TX.ACTION ON LINE(S):/
2249 011340
2250 011346
2251 011354
2252 011362
2253 011370

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 50
CVDHCA.P11 12-JUL-83 11:44 GLOBAL MESSAGE AREA

2254 011376
2255 011404
2256 011412
2257 011420
2258 011422 EM9104:: .ASCIZ / UNEXPECTED DATA FOUND IN FIFO FROM LINE: /
2259 011430
2260 011436
2261 011444
2262 011452
2263 011460
2264 011466
2265 011474
2266 011476 EM9201:: .ASCIZ /SPLIT SPEED TEST /
2267 011504
2268 011512
2269 011520 EM9301:: .ASCIZ /BMP CODE REPORT/
2270 011526
2271 011534
2272 011540 EM9302:: .ASCIZ /BMP CODE FOUND IN TEST /
2273 011546
2274 011554
2275 011562
2276 011570 EM9303:: .ASCIZ /THE LAST BMP CODE WAS FOUND IN TEST /
2277 011576
2278 011604
2279 011612
2280 011620
2281 011626
2282 011634
2283 011635 EM9304:: .ASCIZ /UNEXPECTED BMP CODES FOUND DURING THIS PASS/
2284 011642
2285 011650
2286 011656
2287 011664
2288 011672
2289 011700
2290 011706
2291 .EVEN
2292 .LIST BIN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 51
GLOBAL MESSAGE AREA

2293
2294
2295
2296
2297
2298
2299
2300
2301

.SBTTL GLOBAL ERROR REPORT SECTION

:++
: THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
: USED BY MORE THAN ONE TEST TO OUTPUT ADDITIONAL ERROR INFORMATION. PRINTB
: (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
:--

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 52
GLOBAL ERROR REPORTING ROUTINE

- ER0101 -

2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357

```

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

```

```

011712          BGNMSG ER0101
011712          ER0101::
011712          SAVE          ;SAVE THE GPR CONTENTS.
011712          JSR          R5,PREG05 ;CALL REGISTER SAVE SUBRT.
004537 005232
011716 032705 000001      BIT      #BIT0,R5 ;TEST FOR READ ERROR.
011722 001410          BEQ      2$      ;SKIP READ ERROR MSG IF NO READ ERROR.
011724          PRINTB #MSG1 ;PRINT READ ERROR MESSAGE.
012746 012016          MOV      #MSG1,-(SP)
011730 012746 000001      MOV      #1,-(SP)
011734 010600          MOV      SP,R0
011736 104414          TRAP    C$PNTB
011740 062706 000004      ADD      #4,SP
011744 032705 000002      2$: BIT      #BIT1,R5 ;TEST FOR WRITE ERROR.
011750 001410          BEQ      4$      ;SKIP WRITE ERROR MSG IF NO WRITE ERROR.
011752          PRINTB #MSG2 ;PRINT WRITE ERROR MESSAGE.
012746 012074          MOV      #MSG2,-(SP)
011752 012746 000001      MOV      #1,-(SP)
011762 010600          MOV      SP,R0
011764 104414          TRAP    C$PNTB
011766 062706 000004      ADD      #4,SP
011772          4$: PRINTX #MSG3 ;SUGGEST THAT DHV MAY BE AT WRONG ADDRESS.
012746 012153          MOV      #MSG3,-(SP)
011772 012746 000001      MOV      #1,-(SP)
012002 010600          MOV      SP,R0
012004 104415          TRAP    C$PNTX
012006 062706 000004      ADD      #4,SP
012012          PASS          ;RESTORE THE GPR CONTENTS.
012012 004736          JSR          PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
012014          ENDMSG
012014          L10002:
104423          TRAP    C$MSG

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 53
GLOBAL ERROR REPORTING ROUTINE

- ER0101 -

2358	012016	040445	052502	020123	MSG1:: .ASCIZ /%ABUS TIME-OUT TRAP CAUSED BY READ ATTEMPT.%N/
2359	012024	044524	042515	047455	
2360	012032	052125	052040	040522	
2361	012040	020120	040503	051525	
2362	012046	042105	041040	020131	
2363	012054	042522	042101	040440	
2364	012062	052124	046505	052120	
2365	012070	022456	000116		
2366	012074	040445	052502	020123	MSG2:: .ASCIZ /%ABUS TIME-OUT TRAP CAUSED BY WRITE ATTEMPT.%N/
2367	012102	044524	042515	047455	
2368	012110	052125	052040	040522	
2369	012116	020120	040503	051525	
2370	012124	042105	041040	020131	
2371	012132	051127	052111	020105	
2372	012140	052101	042524	050115	
2373	012146	027124	047045	000	
2374	012153	045	042101	053110	MSG3:: .ASCIZ /%ADHV MAY BE AT THE WRONG Q-BUS ADDRESS.%N%N/
2375	012160	046440	054501	041040	
2376	012166	020105	052101	052040	
2377	012174	042510	053440	047522	
2378	012202	043516	050440	041055	
2379	012210	051525	040440	042104	
2380	012216	042522	051523	022456	
2381	012224	022516	000116		
2382					
2383					.EVEN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 54
GLOBAL ERROR REPORTING ROUTINE

- ER0503 -

2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415

```

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

```

BGNMSG ER0503

ER0503::

PRINTB #EF0503,R1 ;PRINT THE MESSAGE.

```

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

```

ENDMSG

L10003:

TRAP C\$MSG

```

012230
012230
012230
012230 010146
012232 012746 005363
012236 012746 000002
012242 010600
012244 104414
012246 062706 000006
012252
012252
012252 104423

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 55
GLOBAL ERROR REPORTING ROUTINE

- ER1603 -

2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463

012254
012254
012254
012254 004537 005232
012260
012260 010146
012262 012746 005363
012266 012746 000002
012272 010600
012274 104414
012276 062706 000006
012302 013702 005226
012306
012306 010246
012310 012746 005370
012314 012746 000002
012320 010600
012322 104414
012324 062706 000006
012330
012330 004736
012332
012332
012332 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.
*
* 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.
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
PASS ;RESTORE THE CONTENTS OF THE GPRS.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
ENDMSG
L10004:
TRAP C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 56
GLOBAL ERROR REPORTING ROUTINE

- ER6201 -

2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519

```

.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER6201 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH IS INTENDED FOR USE IN THE
* FRAMING ERROR AND PARITY ERROR TESTS. IT REPORTS ERROR INFORMATION
* WHEN A CHARACTER HAS BEEN READ FROM THE DUT WITH THE INCORRECT
* COMBINATION OF FRAMING AND PARITY ERROR BITS.
*
* INPUTS: R2 - DATA BYTE READ FROM THE DUT, INCLUDING ERROR FLAGS.
* R3 - LINE NUMBER MULTIPLIED BY 2.
* R5 - MESSAGE FLAGS, WHICH MESSAGES TO REPORT.
* BIT1 AND BIT3 - INDICATE WHICH MESSAGES ARE TO BE
* REPORTED, FRAMING OR PARITY RESPECTIVELY.
* BIT0 AND BIT 2 - 'SET'/'CLEAR' MESSAGE FOR
* FRAMING AND PARITY ERRORS BITS.
*
* OUTPUTS: MESSAGES ARE PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: INCLUDE THE LABEL 'ER6201' AS THE MESSAGE POINTER
* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
*
* COMMENTS: THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
* THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD OF THE DUT
* CSR MAY BE ALTERED.
*
* SUBORDINATE ROUTINES USED: PRTLPR.
*****

```

```

012334
012334
012334 004537 005232
012340 016304 005142
012344 006203
012346 006204
012350
012350 010446
012352 010346
012354 012746 005456
012360 012746 000003
012364 010600
012366 104414
012370 062706 000010
012374 012704 007557
012400 012701 007373
012404 032705 000002
012410 001427
012412 032705 000001
012416 001403
012420 010401
012422 012704 007373
012426
012426 010146

```

```

BGNMSG ER6201
SAVE JSR ER6201::
;SAVE THE CONTENTS OF THE GPR'S.
R5,PREG05 ;CALL REGISTER SAVE SUBRT.

MOV TXRXLB(R3),R4 ;GET THE ASSOCIATED TX LINE NUMBER.
ASR R3 ;CALCULATE THE RX LINE NUMBER.
ASR R4 ;CALCULATE THE ASSOCIATED LINE NUMBER.
PRINTB #EF6201,R3,R4 ;REPORT THE ERROR TYPE AND LINE NUMBERS.

MOV R4,-(SP)
MOV R3,-(SP)
MOV #EF6201,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #10,SP

;+
; REPORT FRAMING ERROR PROBL...
;-
MOV #EM6202,R4 ;SELECT THE 'ERROR BIT CLEAR' MESSAGE.
MOV #EM0509,R1 ;SELECT EXPECTED 'ERROR BIT SET' MESSAGE.
BIT #BIT1,R5 ;TEST IF FRAMING ERROR MESSAGE TO BE REPORTED.
BEQ 6$ ;BRANCH TO REPORT PARITY ERROR.
BIT #BIT0,R5 ;TEST 'ERROR BIT SET/CLEAR' MESSAGE FLAG.
BEQ 2$ ;BRANCH TO REPORT ERROR BIT 'CLEAR'.
MOV R4,R1 ;SELECT EXPECTED 'CLEAR' STATE MESSAGE.
MOV #EM0509,R4 ;SELECT THE 'ERROR BIT SET' MESSAGE.
2$: PRINTX #EF6202,R4,R1 ;REPORT THE SOURCE OF THE PROBLEM.
MOV R1,-(SP)

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 58
GLOBAL ERROR REPORTING ROUTINE

- ER6401 -

2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER6401 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ADDITIONAL ERROR
* INFORMATION AFTER THE ERROR MESSAGE HEADER.
* 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.
*****
```

012562
012562
012562
012562 004537 005232
012566 005002
012570 012703 000010
012574 010146
012576 012746 005363
012602 012746 000002
012606 010600
012610 104414
012612 062706 000006
012616 000241
012620 006205
012622 103011
012624 010246
012626 012746 005764
012632 012746 000002
012636 010600
012640 104414
012642 062706 000006
012646 005202
012650 020302
012652 001362
012654 004736
012656

```
BGNMSG ER6401
SAVE ;SAVE THE CONTENTS OF THE GPRS.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
ER6401::
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$PNTB
ADD #6,SP
CLC ;CLEAR CARRY.
2$: 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$PNTB
ADD #6,SP
4$: INC R2 ;INCREMENT LINE COUNT.
CMP R3,R2 ;CHECK IF MAX LINE COUNT EXCEEDED.
BNE 2$ ;LOOP IF NOT DONE.
60$: PASS ;RESTORE THE SAVED CONTENTS OF THE GPRS.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
ENDMSG
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 59
GLOBAL ERROR REPORTING ROUTINE

- ER6401 -

2617 012656
2618 012656 104423

L10006: TRAP C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 60
GLOBAL ERROR REPORTING ROUTINE

- ER7801 -

2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654

```

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

```

```

012660
012660
012660 010346
012662 010146
012664 012746 006033
012670 012746 000003
012674 010600
012676 104414
012700 062706 000010
012704
012704
012704 104423

```

BGNMSG ER7801

ER7801::

PRINTB #EF7801,R1,R3 ;PRINT THE MESSAGE.

```

MOV R3,-(SP)
MOV R1,-(SP)
MOV #EF7801,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #10,SP

```

ENDMSG

L10007:

TRAP C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 61
GLOBAL ERROR REPORTING ROUTINE

- ER9001 -

2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702

012706
012706
012706 010146
012706 012746 006071
012714 012746 000002
012720 010600
012722 104414
012724 062706 000006
012730
012730 010446
012732 012746 006153
012736 012746 000002
012742 010600
012744 104415
012746 062706 000006
012752
012752 010246
012754 012746 006225
012760 012746 000002
012764 010600
012766 104415
012770 062706 000006
012774
012774
012774 104423

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9001 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH REPORTS AN UNEXPECTED
* CODE WHICH HAS BEEN FOUND IN THE DUT CSR. THIS CODE CAN BE A BMP
* CODE, A SELF-TEST CODE, OR A MODEM STATUS CODE.
*
* INPUTS: R1 - ADDRESS OF MESSAGE TO PRINT FIRST.
* R2 - SINGLE BYTE CODE WHICH HAS BEEN READ FROM THE DUT.
* R4 - LINE NUMBER ASSOCIATED WITH THE CODE.
*
* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: INCLUDE THE LABEL 'ER9001' 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 ER9001

ER9001::

PRINTB #EF9001,R1 ;REPORT TYPE OF CODE FOUND.

MOV R1,-(SP)
MOV #EF9001,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP

PRINTX #EF9002,R4 ;REPORT THE LINE NUMBER OF THE CODE.

MOV R4,-(SP)
MOV #EF9002,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTX
ADD #6,SP

PRINTX #EF9003,R2 ;REPORT THE CODE WHICH WAS FOUND.

MOV R2,-(SP)
MOV #EF9003,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTX
ADD #6,SP

ENDMSG

L10010:

TRAP C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 62
GLOBAL ERROR REPORTING ROUTINE

- ER9002 -

2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758

012776
012776
012776 006203
013000 042702 177400
013004
013004 010346
013006 010146
013010 012746 006335
013014 012746 000003
013020 010600
013022 104414
013024 062706 000010
013030
013030 010246
013032 012746 010440
013036 012746 006254
013042 012746 000003
013046 010600
013050 104415
013052 062706 000010
013056 005704
013060 100414
013062
013062 010446
013064 012746 010414
013070 012746 006254
013074 012746 000003
013100 010600
013102 104415
013104 062706 000010
013110 000412
013112
013112 012746 010414

.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::

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.
BMI 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)
MOV SP,R0
TRAP C\$PNTX
ADD #10,SP
BR 60\$;EXIT THIS ROUTINE.
2\$: PRINTX #EF9005,#EM9009 ;PRINT MESSAGE INDICATING NO EXPECTED DATA.
MOV #EM9009,-(SP)

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 63
GLOBAL ERROR REPORTING ROUTINE

- ER9002 -

2759	013116	012746	006304
2760	013122	012746	000002
2761	013126	010600	
2762	013130	104415	
2763	013132	062706	000006
2764	013136	004737	020530
2765	013142		
2766	013142		
2767	013142	104423	

608: JSR PC,PRTLPR
ENDMSG

:PRINT CONTENTS OF THE LPR REGISTER.

L10011:

MOV	#EF9005,-(SP)
MOV	#2,-(SP)
MOV	SP,R0
TRAP	C\$PNTX
ADD	#6,SP
TRAP	C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 64
GLOBAL ERROR REPORTING ROUTINE

- ER9003 -

2768
2769
2770
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
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823

```

.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9003 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH IS INTENDED FOR USE IN THE
* TRANSMISSION AND RECEPTION TESTS. IT REPORTS ERROR INFORMATION WHEN
* A CHARACTER HAS BEEN READ FROM THE DUT WITH AN ERROR FLAG OR FLAGS
* SET (IE. OVER-RUN, FRAMING, OR PARITY FLAG).
*
* INPUTS: R2 - DATA BYTE READ FROM THE DUT, INCLUDING ERROR FLAGS.
* R3 - LINE NUMBER MULTIPLIED BY 2.
*
* OUTPUTS: A MESSAGES IS PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: INCLUDE THE LABEL 'ER9003' AS THE MESSAGE POINTER
* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
*
* COMMENTS: THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
* THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD OF THE DUT
* CSR MAY BE ALTERED.
*
* SUBORDINATE ROUTINES USED: NONE.
*****

```

BGNMSG ER9003

ER9003::

```

ASR R3 ;CALCULATE THE LINE NUMBER.
PRINTB #EF9007,R3 ;REPORT THE ERROR TYPE AND LINE NUMBER.
MOV R3,-(SP)
MOV #EF9007,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP
MOV R2,R1 ;EXTRACT THE RECEIVED CHARACTER FROM THE
BIC #177400,R1 ; PASSED IN CHAR VALUE WITH FLAGS.
PRINTX #EF9008,R1 ;REPORT THE VALUE OF THE RECEIVED CHAR.
MOV R1,-(SP)
MOV #EF9008,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #6,SP

```

REPORT OVERRUN FLAG SET IF NECESSARY.

```

MOV #EM9011,R1 ;SELECT THE OVERRUN ERROR MESSAGE.
BIT #BIT14,R2 ;CHECK OVERRUN ERROR FLAG IN PASSED IN CHAR.
BEQ 2$ ;SKIP ERROR IF OVERRUN ERROR FLAG WAS CLEAR.
JSR PC,50$ ;REPORT THE OVERRUN ERROR FLAG WAS SET.

```

REPORT FRAMING FLAG SET IF NECESSARY.

```

2$: MOV #EM9012,R1 ;SELECT THE FRAMING ERROR MESSAGE.
BIT #BIT13,R2 ;CHECK FRAMING ERROR FLAG IN PASSED IN CHAR.
BEQ 4$ ;SKIP ERROR IF FRAMING ERROR FLAG WAS CLEAR.
JSR PC,50$ ;REPORT THE FRAMING ERROR MESSAGE.

```

```

013144
013144
013144 006203
013146
013150 012746 006354
013154 012746 000002
013160 010600
013162 104414
013164 062706 000006
013170 010201
013172 042701 177400
013176
013176 010146
013200 012746 006450
013204 012746 000002
013210 010600
013212 104415
013214 062706 000006
013220 012701 010464
013224 032702 000000
013230 001402
013232 004737 013274
013236 012701 010474
013242 032702 020000
013246 001402
013250 004737 013274

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 65
GLOBAL ERROR REPORTING ROUTINE

- ER9003 -

```

2824
2825
2826
2827 013254 012701 010504
2828 013260 032702 010000
2829 013264 001415
2830 013266 004737 013274
2831 013272 000412
2832
2833
2834
2835
2836 013274
2837 013274 010146
2838 013276 012746 006507
2839 013302 012746 000002
2840 013306 010600
2841 013310 104415
2842 013312 062706 000006
2843 013316 000207
2844
2845 013320 004737 020530
2846 013324
2847 013324
2848 013324 104423

;+
; REPORT PARITY FLAG SET IF NECESSARY.
;-
4$:  MOV  #EM9013,R1      ;SELECT THE PARITY ERROR MESSAGE.
      BIT  #BIT12,R2     ;CHECK PARITY ERROR FLAG IN PASSED IN CHAR.
      BEQ  60$           ;EXIT ROUTINE IF PARITY ERRO FLAG WAS CLEAR.
      JSR  PC,50$        ;REPORT THE PARITY ERROR MESSAGE.
      BR   60$           ;EXIT THIS ROUTINE.

;+
; LOCAL SUBROUTINE TO REPORT AN ERROR FLAG STATUS.
;-
50$: PRINTX #EF9009,R1
                                MOV  R1,-(SP)
                                MOV  #EF9009,-(SP)
                                MOV  #2,-(SP)
                                MOV  SP,R0
                                TRAP C$PNTX
                                ADD  #6,SP

                                RTS   PC

60$:  JSR  PC,PRTLPR      ;REPORT THE LPR CONTENTS FOR THIS LINE.
      ENDMSG

                                L10012:
                                TRAP  C$MSG

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 66
GLOBAL ERROR REPORTING ROUTINE

- ER9004 -

2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
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

013326
013326

013326 012746 010513
013326 012746 005363
013336 012746 000002

013342 010600
013344 104414
013346 062706 000006

013352 005002
013354 013703 002406

013360 005004
013362 000241
013364 006003
013366 103013
013370
013370 016446 003210
013374 010246
013376 012746 006546
013402 012746 000003
013406 010600
013410 104415
013412 062706 000010
013416 012405
013420 005202
013422 005703
013424 001356

013426
013426
013426 104423

```
.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9004 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH REPORTS ERROR SUMMARIES
* FOR LINES WHICH HAVE EXCEEDED THE SPECIFIED MAXIMUM NUMBER OF
* INDIVIDUAL RECEPTION ERRORS.
*
* INPUTS: R1 - ADDRESS OF MESSAGE TO PRINT FIRST.
* ERCNTB - LABEL AT BASE OF LINE ERROR COUNTERS TABLE.
* ERSMRF - 'REPORT ERROR SUMMARY FOR LINE' FLAGS.
*
* OUTPUTS: A MESSAGE IS PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: INCLUDE THE LABEL 'ER9004' AS THE MESSAGE POINTER
* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
*
* COMMENTS: THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
* THE CONTENTS OF GPR'S R2, R3, R4, AND R5 ARE DESTROYED.
*
* SUBORDINATE ROUTINES USED: NONE.
*****
```

BGNMSG ER9004

ER9004::

PRINTB #EF0503,#EM9014 ;REPORT THE SECONDARY ERROR MESSAGE.

```
MOV #EM9014,-(SP)
MOV #EF0503,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP
```

```
CLR R2 ;CLEAR THE LINE COUNTER.
MOV ERSMRF,R3 ;GET THE ERROR SUMMARY FLAGS.
CLR R4 ;CLEAR 'LINE COUNTER TIMES 2' OFFSET.
2$: CLC ;CLEAR THE CARRY FOR THE FOLLOWING ROTATE.
ROR R3 ;SHIFT ANOTHER ERROR SUMMARY FLAG INTO CARRY.
BCC 4$ ;SKIP PRINTING MESSAGE IF FLAG FOR LINE CLEAR.
PRINTX #EF9010,R2,ERCNTB(R4)
```

```
MOV ERCNTB(R4),-(SP)
MOV R2,-(SP)
MOV #EF9010,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP
```

```
4$: MOV (R4)+,R5 ;INCREMENT THE LINE OFFSET BY 2.
INC R2 ;INCREMT THE LINE COUNTER.
TST R3 ;CHECK THE ERROR SUMMARY FLAGS.
BNE 2$ ;IF MORE FLAGS SET, LOOP TO DO OTHER LINES.
```

ENDMSG

L10013:

```
TRAP C$MSG
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 67
GLOBAL ERROR REPORTING ROUTINE

- ER9005 -

2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957

```

.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9005 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH REPORTS INCOMPLETE DATA
* TRANSMISSIONS OR RECEPTIONS.
*
* INPUTS: R1 - EITHER 'TRANSMITTED' OR 'RECEIVED' TO INDICATE TX OR RX.
* R2 - BIT MAP OF LINES WHICH DID NOT COMPLETE TX OR RX.
* R4 - ADDRESS OF BASE OF THE CORRECT CHARACTER COUNTERS TABLE.
* DPLENB - LABEL AT BASE OF DATA PATTERN LENGTH TABLE.
* EM9015 - SYMBOLIC ADDRESS OF THE 'TRANSMITTED' MESSAGE.
*
* OUTPUTS: A MESSAGE IS PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: INCLUDE THE LABEL 'ER9005' AS THE MESSAGE POINTER
* PARAMETER IN THE DIAG SUPER ERROR REPORT MACRO CALL.
*
* COMMENTS: THE MESSAGE IS PRINTED AS BASIC AND EXTENDED ERROR INFORMATION.
* THE CONTENTS OF THE INDIRECT ADDRESS FIELD IN THE DUT CSR MAY
* BE ALTERED.
*
* SUBORDINATE ROUTINES USED: PRTLPR.
*****

```

```

BGNMSG ER9005
SAVE ;SAVE THE CONTENTS OF THE GPR'S.
R5,PREG05 ;CALL REGISTER SAVE SUBRT.
PRINTB #EF9013,R1 ;REPORT THE SECONDARY ERROR MESSAGE.
MOV R1,-(SP)
MOV #EF9013,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP CSPNTB
ADD #6,SP
CLR R3 ;CLEAR THE LINE COUNTER.
CMP #EM9015,R1 ;CHECK IF ADDRESS CORRESPONDS TO TX MESSAGE.
BNE 6$ ;BRANCH IF RECEPTION MESSAGE TO BE PRINTED.
;+
; PERFORM TX INCOMPLETE ERROR MESSAGE REPORTING.
;-
PRINTX #EM9030 ;PRINT 'NO TX COMPLETION INTERRUPTS RECEIVED'
MOV #EM9030,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP CSPNTX
ADD #4,SP
2$: CLC ;CLEAR THE CARRY FOR THE FOLLOWING ROTATE.
ROR R2 ;SHIFT 'TX NOT DONE' FLAG INTO CARRY.
BCC 4$ ;SKIP PRINTING MESSAGE IF FLAG FOR LINE CLEAR.
PRINTX #EF9020,R3 ;PRINT 'TOO FEW TX.ACTIONS GENERATED' MSG.
MOV R3,-(SP)
MOV #EF9020,-(SP)
MOV #2,-(SP)
MOV SP,R0

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 68
GLOBAL ERROR REPORTING ROUTINE

- ER9005 -

```

2958 013530 104415                                TRAP    C$PNTX
2959 013532 062706 000006                        ADD     #6,SP
2960 013536 004737 020530                        JSR     PC,PRTLPR      ;REPORT CONTENTS OF LPR REGISTER FOR THIS LINE.
2961 013542 005203 4$: INC R3                ;INCREMENT LINE COUNTER.
2962 013544 005702      TST R2                ;CHECK THE 'TX NOT DONE FLAGS'.
2963 013546 001357      BNE 2$                ;IF MORE FLAGS SET, LOOP TO DO OTHER LINES.
2964 013550 000440      BR 10$               ;EXIT THIS ROUTINE.
2965
2966      ;+
2967      ; PERFORM RX INCOMPLETE ERROR MESSAGE REPORTING.
2968 013552 000241 6$: CLC                ;CLEAR THE CARRY FOR THE FOLLOWING ROTATE.
2969 013554 006002      ROR R2                ;SHIFT 'RX NOT DONE' FLAG INTO CARRY.
2970 013556 103031      BCC 8$                ;SKIP PRINTING MESSAGE IF FLAG FOR LINE CLEAR.
2971 013560 006303      ASL R3                ;SHIFT LINE # TO GIVE CORRECT TABLE OFFSET.
2972 013562 016305 005142      MOV TXRXLB(R3),R5    ;GET THE 'ASSOCIATED' RECEIVE LINE OFFSET.
2973 013566 010246      MOV R2,-(SP)         ;SAVE THE 'RX NOT DONE' FLAGS ON THE STACK.
2974 013570 010502      MOV R5,R2            ;COPY THE ASSOCIATED TX LINE OFFSET.
2975 013572 016505 003350      MOV CHCNTB(R5),R5   ;GET THE TOTAL NUMBER OF EXPECTED CHARS.
2976 013576 006202      ASR R2                ;SHIFT THE TABLE OFFSET TO GIVE A LINE NUMBER.
2977 013600 006203      ASR R3                ;SHIFT TABLE OFFSET TO GIVE LINE NUMBER.
2978 013602      PRINTX #EF9012,R3,R1,(R4),R5,R2 ;REPORT NUMBER OF CHARS ON LINE.
2979 013602 010246      MOV R2,-(SP)
2980 013604 010546      MOV R5,-(SP)
2981 013606 011446      MOV (R4),-(SP)
2982 013610 010146      MOV R1,-(SP)
2983 013612 010346      MOV R3,-(SP)
2984 013614 012746 006635      MOV #EF9012,-(SP)
2985 013620 012746 000006      MOV #6,-(SP)
2986 013624 010600      MOV SP,R0
2987 013626 104415      TRAP    C$PNTX
2988 013630 062706 000016      ADD     #16,SP
2989 013634 012602      MOV (SP)+,R2        ;RESTORE THE 'RX NOT DONE' FLAGS.
2990 013636 004737 020530      JSR     PC,PRTLPR      ;REPORT CONTENTS OF LPR REGISTER FOR THIS LINE.
2991 013642 005724 8$: TST (R4)+            ;INCREMENT THE CHARACTER COUNTER TABLE.
2992 013644 005203      INC R3                ;INCREMENT THE LINE COUNTER.
2993 013646 005702      TST R2                ;CHECK THE 'RX NOT DONE FLAGS'.
2994 013650 001340      BNE 6$                ;IF MORE FLAGS SET, LOOP TO DO OTHER LINES.
2995 013652 10$: PASS 6$                ;RESTORE THE CONTENTS OF THE GPRS.
2996 013652 004736      JSR     PC,@(SP)+     ;RETURN TO PREG05 SUBRT.
2997 013654      ENDMSG
2998 013654      L10014:
2999 013654 104423      TRAP    C$MSG

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 69
GLOBAL ERROR REPORTING ROUTINE

- FR9102 -

3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055

```

.SBTTL GLOBAL ERROR REPORTING ROUTINE - ER9102 -
*****
* THIS IS AN ERROR REPORTING SUBROUTINE WHICH PRINTS ADDITIONAL ERROR
* INFORMATION AFTER THE ERROR MESSAGE HEADER.
* THIS ROUTINE IS PASSED A BIT MAP WHICH SPECIFIES THE LINES FOR WHICH
* THE ERROR CONDITION SHOULD BE REPORTED.
*
* INPUTS: R1 - ADDRESS OF THE MESSAGE TO BE PRINTED BY THIS ROUTINE.
* R2 - BIT MAP OF LINES FOR WHICH TO REPORT ERRORS.
*
* OUTPUTS: MESSAGES ARE PRINTED AT THE OPERATOR CONSOLE.
*
* CALLING SEQUENCE: LOAD THE ADDRESS OF THE MESSAGE IN R1.
* LOAD THE BIT MAP OF LINES WITH ERRORS IN R2.
* INCLUDE THE LABEL 'ER9102' AS THE MESSAGE POINTER
* (ERRBLK) IN THE DIAG SUPER ERROR REPORT MACRO CALL.
*
* COMMENTS: THE OUTPUT FORMAT OF THIS MESSAGE IS:
* 'TEXT MESSAGE POINTED TO BY R1'
* 'ERROR CONDITION ON LINE NN'
* 'ERROR CONDITION ON LINE ...'
* THE TOP MESSAGE, AND THE MESSAGE FOR EACH LINE ARE PRINTED
* AS BASIC ERROR INFORMATION.
*
* SUBORDINATE ROUTINES USED: NONE.
*****

```

```

BGNMSG ER9102
ER9102::
SAVE ;SAVE THE CONTENTS OF THE GPRS.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.

PRINTB #EF0503,R1 ;PRINT THE FIRST LINE OF THE MESSAGE.
MOV R1,-(SP)
MOV #EF0503,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP

2$: CLR R3 ;CLEAR THE LINE NUMBER.
LC ;PREPARE TO ROTATE NEXT BIT OUT OF MAP.
ROR R2 ;GET THE NEXT BIT OF THE BIT MAP.
BCC 4$ ;SKIP PRINTING MESSAGE IF THE BIT IS CLEAR.
PRINTB #EF9103,R3 ;REPORT THIS LINE HAD THE ERROR.
MOV R3,-(SP)
MOV #EF9103,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #6,SP

4$: INC R3 ;INCREMENT THE LINE COUNTER.
TST R2 ;CHECK THE BIT MAP.
BNE 2$ ;LOOP IF NOT ALL SET BITS REMOVED FROM BIT MAP.
PRINTB #EF9101 ;PRINT A BLANK LINE.
MOV #EF9101,-(SP)
MOV #1,-(SP)

```

```

013656
013656
013656 004537 005232
013662
013662 010146
013664 012746 005363
013670 012746 000002
013674 010600
013676 104414
013700 062706 000006
013704 005003
013706 000241
013710 006002
013712 103011
013714
013714 010346
013716 012746 007121
013722 012746 000002
013726 010600
013730 104414
013732 062706 000006
013736 005203
013740 005702
013742 001361
013744
013744 012746 007116
013750 012746 000001

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 70
GLOBAL ERROR REPORTING ROUTINE

- ER9102 -

3056 013754 010600
3057 013756 104414
3058 013760 062706 000004
3059 013764
3060 013764 004736
3061 013766
3062 013766
3063 013766 104423

60\$: PASS
ENDMSG

JSR

;RESTORE THE SAVED CONTENTS OF THE GPRS.
PC,@(SP)+ ;RETURN TO PREG05 SUBRT.

L10015: TRAP C\$MSG

MOV SP,RO
TRAP C\$PNTB
ADD #4,SP

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 71
GLOBAL ERROR REPORTING ROUTINE

- ER9301 -

3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119

```

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

```

BGNMSG ER9301

ER9301::

```

SAVE ;SAVE THE GPRS ON THE STACK.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.

```

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

```

```

2$: MOV #BMPCQB,R3 ;GET THE START ADDRESS OF THE BMP CODE QUEUE.
MOV #EM9302,R5 ;GET THE MESSAGE TO BE REPORTED.
MOV (R3)+,R1 ;GET THE NUMBER OF THE TEST THAT WAS EXECUTING.
MOV (R3)+,R4 ;GET BMP CODE THAT WAS REPORTED OFF THE QUEUE.
JSR PC,50$ ;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.
-

```

```

CMP R2,#BMPCQE-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.
PRINTX #EF9302 ;REPORT OVERFLOW CONDITION.

```

```

MOV #EF9302,-(SP)
MOV #1,-(SP)
MOV SP,R0

```

```

013770
013770
013770 004537 005232
013774
013774 010146
013776 012746 005363
014002 012746 000002
014006 010600
014010 104414
014012 062706 000006
014016 012703 002420
014022 012705 011540
014026 012301
014030 012304
014032 004737 014114
014036 020302
014040 103772
014042 020227 002614
014046 001036
014050 005762 000002
014054 001433
014056 012301
014060 011304
014062 012705 011570
014066
014066 012746 007235
014072 012746 000001
014076 010600

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 72
GLOBAL ERROR REPORTING ROUTINE

- ER9301 -

3120	014100	104415					TRAP	C\$PNTX
3121	014102	062706	000004				ADD	#4,SP
3122	014106	004737	014114					
3123	014112	000414						
3124								
3125	014114			50\$:	PRINTX	#EF9301,R5,R1,R4		;PRINT THE MESSAGE.
3126	014114	010446					MOV	R4,-(SP)
3127	014116	010146					MOV	R1,-(SP)
3128	014120	010546					MOV	R5,-(SP)
3129	014122	012746	007167				MOV	#EF9301,-(SP)
3130	014126	012746	000004				MOV	#4,-(SP)
3131	014132	010600					MOV	SP,R0
3132	014134	104415					TRAP	C\$PNTX
3133	014136	062706	000012				ADD	#12,SP
3134	014142	000207						
3135	014144			60\$:	RTS	PC		;RETURN.
3136	014144	004736			PASS			;RESTORE THE GPR CONTENTS.
3137						JSR		PC,@(SP)+
3138	014146				ENDMSG			;RETURN TO PREG05 SUBRT.
3139	014146							
3140	014146	104423					L10016:	TRAP C\$MSG

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 73
GLOBAL SUBROUTINES SECTION

.SBTTL GLOBAL SUBROUTINES SECTION

3141
3142
3143
3144
3145
3146
3147

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

CVDHCAJ DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 74
GLOBAL SUBROUTINE - ALTFLD -

```

3148 .SBTTL GLOBAL SUBROUTINE - ALTFLD -
3149 :+ *****
3150 :+ - ALTER DEVICE REGISTER FIELDS ROUTINE -
3151 :+ THIS SUBROUTINE ALTERS THE SPECIFIED FIELD OF THE SPECIFIED DEVICE
3152 :+ REGISTER FOR THE SPECIFIED LINES. THIS ROUTINE CAN BE USED TO SET
3153 :+ OR CLEAR BITS WITHIN SELECTED FIELDS OF SELECTED REGISTERS.
3154 :+ USE EXAMPLES: SET RX.BAUD.RATE FIELDS ON LINES 3 AND 6.
3155 :+ CLEAR TX.DMA BITS ON ALL LINES.
3156 :+
3157 :+ INPUTS: R1 - ADDRESS OF THE REGISTERS TO ALTER.
3158 :+ R2 - BIT FIELDS SET TO DESIRED STATES.
3159 :+ R3 - BIT MAP OF LINES FOR WHICH TO ALTER REGISTER.
3160 :+ R4 - MASK OF BITS TO ALTER (1 INDICATES CHANGE BIT).
3161 :+ CSRA - CONTAINS THE ADDRESS OF THE DEVICE CSR.
3162 :+ IESTAT - SAVED STATES OF THE INTERRUPT ENABLE BITS.
3163 :+
3164 :+ OUTPUTS: DEVICE REGISTERS - SPECIFIED REGISTER FIELDS ALTERED.
3165 :+ CSR IND.ADR.REG FIELD - DESTROYED.
3166 :+
3167 :+ CALLING SEQUENCE: JSR PC,ALTFLD
3168 :+
3169 :+ COMMENTS: THIS ROUTINE READS THE SPECIFIED REGISTERS FOR ALL LINES
3170 :+ WITH NUMBERS LOWER THAN THE HIGHEST SPECIFIED LINE.
3171 :+ THIS ROUTINE DOES NOT READ THE CSR.
3172 :+
3173 :+ SUBROUTINES CALLED: NONE.
3174 :+-----
3175
3176 014150 ALTFLD:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
3177 014150 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3178
3179 :+
3180 :+ SET UP TO LOOP FOR EACH LINE:
3181 :+ PREPARE THE WORD TO BE ORED INTO THE REGISTER CONTENTS.
3182 :+ SET UP THE WORD TO WRITE INTO THE IND.ADR.REG FIELD OF THE CSR.
3183 :+-----
3184 014154 010400 MOV R4,R0 ;CALCULATE THE NEW CONTENTS OF THE
3185 014156 005100 COM R0 ; REGISTER FIELDS WHICH ARE TO BE
3186 014160 040002 BIC R0,R2 ; ALTERED BY THIS ROUTINE.
3187 014162 013705 002234 MOV IESTAT,R5 ;SET UP TO WRITE IND.ADR.REG FIELD TO 0.
3188
3189 :+
3190 :+ LOOP ONCE FOR EACH LINE, ALTERING THE SPECIFIED FIELD IN THE SPECIFIED
3191 :+ REGISTER IF THE LINE HAS BEEN SELECTED FOR ALTERING.
3192 :+ EXIT THE LOOP IF NO MORE LINES TO ALTER, OR IF WE HAVE ALTERED THE MAX
3193 :+ ALLOWABLE NUMBER OF LINES (AS SPECIFIED BY NUMLNS).
3194 :+-----
3194 014166 000241 CLC ;PREPARE FOR ROTATE, 'TST R5' DOES THIS BELOW.
3195 014170 006003 2$: ROR R3 ;GET THE LINE SELECT BIT FOR THIS LINE.
3196 014172 103006 BCC 4$ ;SKIP SETUP IF LINE IS NOT SELECTED.
3197 014174 010577 166002 MOV R5,@CSRA ;SET DUT CSR IND.ADR.REG FIELD TO THIS LINE.
3198 014200 011100 MOV (R1),R0 ;GET THE PRESENT CONTENTS OF THE REG TO ALTER.
3199 014202 040400 BIC R4,R0 ;CLEAR THE BIT FIELDS WE ARE TO ALTER.
3200 014204 050200 BIS R2,R0 ;OR IN THE NEW STATES OF THE FIELDS.
3201 014206 010011 MOV R0,(R1) ;WRITE THE NEW REGISTER CONTENTS TO THE REG.
3202 014210 005205 4$: INC R5 ;SET LINE NUMBER TO THE NEXT LINE.
3203 014212 005703 TST R3 ;CHECK FOR UNHANDLED LINES, CLEAR CARRY FLAG.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 75
GLOBAL SUBROUTINE

- ALTFLD -

3204 014214 001365
3205
3206 014216
3207 014216 004736
3208 014220 000207

BNE 28

;LOOP IF SELECTED LINE(S) IS NOT HANDLED.

60\$: PASS

;RESTORE GPRS.

ISR

PC,@(SP)+

;RETURN TO PREG05 SUBRT.

RTS PC

;RETURN TO CALLING ROUTINE.

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 76
GLOBAL SUBROUTINE

- CALMSL -

3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264

```

.SBTTL GLOBAL SUBROUTINE - CALMSL -
:++ *****
:  - CALIBRATE MILLI SECOND LOOP COUNT SUBROUTINE -
:  THIS SUBROUTINE CALIBRATES THE TIMING LOOP WHICH IS USED IN THE MSLOOP
:  ROUTINE. THIS SUBROUTINE CALCULATES A VALUE FOR THE MSLCNT VARIABLE
:  WHICH IS THE NUMBER OF SOFTWARE LOOPS WHICH TAKES 1 MS TO EXECUTE IN
:  THE MSLOOP ROUTINE. THIS ROUTINE CALIBRATES THE COUNT BY USING THE
:  LINE TIME CLOCK (LTC), SO IF NO LTC IS AVAILABLE THE DEFAULT VALUE FOR
:  THE DELAY COUNT MUST BE USED.
:
:  INPUTS:  MSLCNT - DEFAULT 1 MS DELAY LOOP COUNT VALUE, OR
:           VALUE FROM PREVIOUS CALIBRATION.
:           MSTICK - NUMBER OF MS PER LTC CLOCK TICK.
:           TIMER1 - TIMER COUNTER CHANGED BY LTC INTERRUPT SERVICE RTN.
:           CLKHRZ - NUMBER OF LTC CLICKS PER SECOND (50 OR 60).
:
:  OUTPUTS:  CARRY - SET IF LTC IS AVAILABLE, AND NEW CALIBRATION PERFORMED.
:           MSLCNT - NEW 1 MS DELAY LOOP COUNT VALUE IF LTC AVAILABLE, OR
:           UNCHANGED IF NO LTC IS AVAILABLE.
:
:  CALLING SEQUENCE:  JSR  PC,CALMSL
:
:  COMMENTS:
:
:  SUBORDINATE ROUTINES CALLED: UNSDIV,OOPS.
:-- *****
CALMSL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR  R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          CLR  62$ ;CLEAR THE 2ND TIME FLAG.
:
:  SYNCHRONIZE WITH THE LTC.
2$:  MOV  #1,R5 ;SET OUTER LOOP COUNTER TO 1 LOOP.
          ;INCREASE THE VALUE LOADED INTO THIS COUNTER IF THE < **
          ;FOLLOWING LOOP FAILS ON FUTURE, FASTER PROCESSORS. < **
          CLR  R0 ;CLEAR THE WAIT FOR CLOCK INT COUNTER.
          MOV  #1,TIMER1 ;SET UP COUNT OF 1 TO SYNCH WITH LTC.
          TST  TIMER1 ;CHECK FOR COUNTER HAVING GONE TO ZERO.
          BEQ  6$ ;JUMP OUT OF LOOP IF LTC HAS INTERRUPTED.
          INC  R0 ;COUNT THIS ITERATION OF THE INNER LOOP.
          BNE  4$ ;LOOP IF COUNTER HAS NOT TURNED OVER.
          DEC  R5 ;DECREMENT THE INNER LOOP COUNTER.
          BGT  4$ ;LOOP IF OUTER LOOP COUNT NOT UP.
:
:  IF WE GOT NO LTC INTERRUPT, INDICATE THAT THERE IS NO LTC AVAILABLE.
:  LTC MUST BE FLAKEY, OR NOT REALLY AN LTC AT ALL.
:
          CLR  CLKHRZ ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
          CLC ;INDICATE FAILURE FOR RETURN.
          BR  60$ ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
:
:  WE ARE NOW SYNCHRONIZED WITH THE LTC.
:  SET UP FOR THE CALIBRATION LOOP.
:--

```

014222			
014222	004537	005232	
014226	005037	014442	
014232	012705	000001	
014236	005000		
014240	012737	000001	002270
014246	005737	002270	
014252	001410		
014254	005200		
014256	001373		
014260	005305		
014262	003371		
014264	005037	002266	
014270	000241		
014272	000461		

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 77
GLOBAL SUBROUTINE

- CALMSL -

```

3265 014274 012704 002270      6$:   MOV     #TIMER1,R4      ;WILL TEST TIMER1 IN THE LOOP BELOW.
3266 014300 005001              CLR     R1                ;CLEAR THE OUTER LOOP COUNTER.
3267 014302 005002              CLR     R2                ;INDICATE TO CHECK ALL BITS OF TIMER1.
3268 014304 005003              CLR     R3                ;INDICATE TO CHECK FOR TIMER1 CLEAR.
3269 014306 012714 000001      MOV     #1,(R4)           ;LOAD TIMER1 WITH COUNT OF 1.
3270
3271 014312 013705 002302      8$:   MOV     MSLCNT,R5      ;LOAD MS LOOP COUNT.
3272 014316 011400      10$:  MOV     (R4),R0          ;GET THE TIMER1 VALUE.
3273 014320 010037 014444      MOV     R0,64$           ;SAVE WORD (LIKE IN THE REAL LOOP).
3274 014324 040200              BIC     R2,R0             ;LEAVE ALL THE BITS.
3275 014326 020003              CMP     R0,R3            ;COMPARE AGAINST ZERO.
3276 014330 000261              SEC                     ;SET CARRY IN CASE OF SUCCESS.
3277 014332 001406              BEQ     12$              ;EXIT LOOP IF TIMER1 HAS CLEARED.
3278 014334 005305              DEC     R5               ;COUNT DOWN THE INSIDE MS LOOP COUNT.
3279 014336 001367              BNE     10$              ;LOOP IF MS NOT UP.
3280 014340 005301              DEC     R1               ;DECREMENT THE MS TIME COUNT.
3281 014342 001363              BNE     8$               ;KEEP LOOPING.
3282 014344 004737 020024      JSR     PC,OOPS          ;WE OVERFLOWED, SOMETHING IS WRONG, ABORT.
3283
3284      ;+
3285      ; WE HAVE NOW HAVE LOOP COUNT INFORMATION FOR ONE CLOCK TICK.
3286      ; WE HAVE NEGATIVE OF NUMBER OF OUTER LOOPS IN R1, EACH IS MSLCNT INNER LOOPS.
3287      ; WE HAVE THE PORTION OF THE LAST OUTER LOOP NOT EXECUTED, IN R5.
3288      ; NOW WE CALCULATE THE TOTAL NUMBER OF INNER LOOPS EXECUTED.
3289      ;-
3289 014350 005401      12$:  NEG     R1                ;GET NUMBER OF OUTER LOOPS.
3290 014352 013702 002302      MOV     MSLCNT,R2        ;GET THE NUMBER OF INNER LOOPS PER OUTER LOOP.
3291 014356 010203              MOV     R2,R3            ;COPY NUMBER OF LOOPS FOR MULTIPLY.
3292 014360 160502              SUB     R5,R2            ;CALC # OF INNER LOOPS DONE IN LAST OUTER LOOP
3293 014362 010204              MOV     R2,R4            ; AND ADD TO ACCUMULATOR LSWORD.
3294 014364 005005              CLR     R5               ;CLEAR ACCUMULATOR MSWORD.
3295 014366 005301      14$:  DEC     R1                ;CHECK R1 FOR 0 CONDITION
3296 014370 100403              BMI     16$              ; SKIP MULTIPLICATION IF ZERO
3297 014372 060304              ADD     R3,R4            ;MULTIPLY NUMBER OF INNER
3298 014374 005505              ADC     R5               ; LOOPS PER OUTER LOOP BY
3299 014376 000773              BR     14$              ;NUMBER OF OUTER LOOPS PERFORMED.
3300
3301      ;+
3302      ; DIVIDE THE TOTAL NUMBER OF INNER LOOPS BY THE NUMBER OF MS PER LTC TICK.
3303      ;-
3303 014400 013701 002300      16$:  MOV     MSTICK,R1        ;# OF MS PER LTC TICK IS DIVISOR.
3304 014404 010403              MOV     R4,R3            ;LSWORD OF LOOP COUNT IS LSWORD OF DIVIDEND.
3305 014406 010502              MOV     R5,R2            ;MSWORD OF LOOP COUNT IS MSWORD OF DIVIDEND.
3306 014410 004737 024310      JSR     PC,UNSDIV        ;DIVIDE NUMBER OF LOOPS BY MS PER LTC TICK.
3307 014414 103402              BCS     18$              ;BYPASS OOPS IF WE'RE OK.
3308 014416 004737 020024      JSR     PC,OOPS          ;CLOCK ROUTINES ARE NOT LONG ENOUGH, OR BUG.
3309 014422 010137 002302      18$:  MOV     R1,MSLCNT        ;SET NEW VALUE FOR MS LOOP COUNT.
3310 014426 005137 014442      COM     62$              ;SET THE 2ND ITERATION FLAGS IF 1ST ITERATION.
3311 014432 001277              BNE     2$               ;BRANCH IF ONLY ONE ITERATION DONE.
3312 014434 000261              SEC                     ;SET THE SUCCESS FLAG FOR EXIT.
3313
3314      60$:  PASS                    ;RESTORE GPRS,
3315 014436 004736              PC,@(SP)+                ;RETURN TO PREGOS SUBRT.
3316 014440 000207              RTS     PC                ; CARRY - SUCCESS FLAG. SET IF SUCCESS.
3317
3318 014442 000000      62$:  .WORD  0                ;2ND CALIBRATION ITERATION FLAGS.
3319 014444 000000      64$:  .WORD  0                ;DUMMY WORD FOR STORAGE OF THE READ WORD.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 78
GLOBAL SUBROUTINE

- CHKBMP -

```

3320 .SBTTL GLOBAL SUBROUTINE - CHKBMP -
3321 :+*****
3322 :* - CHECK IF CHARACTER IS A BMP CODE -
3323 :* THIS SUBROUTINE IS USED TO CHECK FOR BMP CODES.
3324 :* IF A BMP CODE IS DETECTED, IT WILL BE SAVED ON THE QUEUE TO BE REPORTED
3325 :* LATER. THE CARRY IS USED AS A FLAG TO INDICATE A CODE HAS BEEN FOUND.
3326 :*
3327 :* INPUTS: R2 - CONTAINS THE DATA TO BE CHECKED.
3328 :*
3329 :* OUTPUTS: R1 - CONTAINS THE MESSAGE TO BE REPORTED.
3330 :* ERRBLK - CONTAINS THE ERROR REPORTING ROUTINE.
3331 :* CARRY BIT IS USED TO INDICATE A BMP CODE FOUND, CARRY SET.
3332 :*
3333 :* CALLING SEQUENCE: JSR PC,CHKBMP
3334 :*
3335 :* COMMENTS:
3336 :*
3337 :* SUBORDINATE ROUTINES CALLED: SAVBMP.
3338 :--*****
3339
3340 014446 CHKBMP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
3341 014446 004537 005232 ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
3342 014452 012700 170301 ;MOV #170301,R0 ;SET UP THE FLAGS OF A BMP CODE.
3343 014456 040200 ;BIC R2,R0 ;TRY TO CLEAR THE BMP CODE FLAGS.
3344 014460 001011 ;BNE 2$ ;IF NOT A BMP CODE, EXIT WITH FAILURE.
3345 014462 004737 022400 ;JSR PC,SAVBMP ;SAVE THE BMP CODE ON THE QUEUE.
3346 014466 012701 007462 ;MOV #EM5303,R1 ;PASS THE MESSAGE TO BE REPORTED.
3347 014472 012737 012254 005230 ;MOV #ER1603,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
3348 014500 000261 ;SEC ;PASS FLAG TO INDICATE SUCCESS, BMP CODE FOUND.
3349 014502 000401 ;BR 60$ ;EXIT.
3350 014504 000241 2$: CLC ;PASS FLAG TO INDICATE FAILURE.
3351 014506 60$: PASS R1 ;RESTORE GPRS, EXCEPT
3352 014506 010166 000004 ;MOV R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
3353 014512 004736 ;JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
3354
3355
3356 014514 000207 ;RTS PC ;R1 - CONTAINS THE ADDRESS OF ERROR MESSAGE.
;CARRY BIT - SET INDICATES SUCCESS.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 79
GLOBAL SUBROUTINE

- CHKEXT -

3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412

```

.SBTTL GLOBAL SUBROUTINE - CHKEXT -
:++ *****
:
: - CHECK FOR EXTRA CHARACTER ROUTINE -
: THIS SUBROUTINE CHECKS FOR THE CONDITION WHICH INDICATES THAT AN EXTRA
: CHARACTER HAS BEEN RECEIVED DURING THE RECEPTION OF A DATA PATTERN.
: IF THIS ROUTINE DETERMINES THAT IT IS LIKELY THAT AN EXTRA CHARACTER
: HAS BEEN RECEIVED IT INDICATES THIS IN THE STATUS INFORMATION RETURNED
: TO THE CALLING ROUTINE.
:
: INPUTS: R3 - RX LINE NUMBER MULTIPLIED BY 2 (OFFSET INTO WORD TABLES).
: R4 - BASE ADDRESS OF RESYNC QUE CONTAINING RX CHARS.
: R5 - MASK OF "INACTIVE" (NON-DATA) BITS OF RX AND TX CHARS.
: CHCNTB - BASE OF NUMBER OF CHARS TO TX ON EACH LINE TABLE.
: RXCNTB - BASE OF THE RX CHARACTER COUNTERS TABLE.
: RXPTRB - BASE OF THE RX CHARACTER POINTERS TABLE.
: TXRXLB - BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
:
: OUTPUTS: CARRY - SET IF EXTRA CHARACTER CONDITION IS VERIFIED.
:
: CALLING SEQUENCE: JSR PC,CHKEXT
:
: COMMENTS: THE FOLLOWING SYMBOLS ARE USED IN LINE COMMENTS:
: CHR0 - CHARACTER AT BOTTOM OF RESYNC QUE (FIRST RECEIVED).
: CHR1, CHR2 - 2 CHARACTERS RECEIVED AFTER CHR0.
: EXPO - CHARACTER EXPECTED TO BE RECEIVED NEXT.
: EXP1, EXP2 - CHARACTER EXPECTED TO BE RECEIVED AFTER EXPO, ETC.
:
: SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
CHKEXT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;CALL REGISTER SAVE SUBRT.
JSR R5,PREG05
MOV RXPTRB(R3),R2 ;GET THE RX DATA POINTER.
TST (R4)+ ;INCREMENT R4 BY 2 TO POINT TO CHR1.
MOV (R4)+,R0 ;GET CHR1 FROM THE QUE, DATA.VALID INTO N FLAG.
BPL 52$ ;EXIT WITH "FAILURE" IF CHR1 NOT VALID.
BIC R5,R0 ;REMOVE INACTIVE BITS FROM CHR1 VALUE.
MOVB (R2)+,R1 ;GET EXPO FROM THE DATA PATTERN.
BIC R5,R1 ;REMOVE INACTIVE BITS FROM EXPO VALUE.
CMPB R1,R0 ;COMPARE CHR1 AND EXPO.
BNE 52$ ;EXIT WITH "FAILURE" IF CHR1 <> EXPO.
MOV RXCNTB(R3),R0 ;COMPARE THE PRESENT RX CHARACTER COUNT PLUS 1
; WITH THE EXPECTED NUMBER OF CHARS TO RX ON
INC R0 ; LINE (NUMBER TRANSMITTED AND LOOPED BACK) TO
MOV TXRXLB(R3),R1 ; DETERMINE IF CHR1 IS LAST EXPECTED CHAR.
CMP R0,CHCNTB(R1) ;EXIT WITH "SUCCESS" IF CHR1 IS LAST CHAR.
BEQ 50$ ;GET CHR2 FROM THE QUE, DATA.VALID INTO N FLAG.
MOV (R4),R0 ;EXIT WITH "SUCCESS" IF CHR1 WAS LAST IN QUE.
BPL 50$ ;REMOVE INACTIVE BITS FROM CHR2 VALUE.
BIC R5,R0 ;GET THE EXP1 VALUE.
MOVB (R2),R1 ;REMOVE INACTIVE BITS FROM EXP1 VALUE.
BIC R5,R1 ;COMPARE CHR2 AND EXP1.
CMP R0,R1 ;EXIT WITH "FAILURE" IF CHR2 <> EXP1.
BNE 52$
:
: +
: IT IS LIKELY THAT WE RECEIVED AN EXTRA CHARACTER WITHIN THE DATA PATTERN.
: INDICATE "SUCCESS" AND EXIT.

```

CVE
CVE

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 80
GLOBAL SUBROUTINE - CHKEXT -

```

3413
3414 014604 000261      :-
3415 014606 000401      50$:   SEC           ;SET THE SUCCESS FLAG.
3416                                     BR      60$         ;EXIT THE ROUTINE.
3417
3418      :-+
3419      ; WE DIDN'T RECEIVE A SINGLE EXTRA CHARACTER AT THIS POINT IN THE DATA PATTERN.
3420      ; INDICATE 'FAILURE' AND EXIT.
3421      :-
3422 014610 000241      52$:   CLC           ;CLEAR THE SUCCESS FLAG.
3423 014612                                     60$:   PASS
3424 014612 004736                                     JSR
3425 014614 000207      RTS      PC           ;RESTORE GPRS.
                                           PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
                                           ;CARRY - SET IF SUCCESS (EXTRA CHAR RXED).

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 81
GLOBAL SUBROUTINE

- CHKLOS -

3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481

```

.SBTTL GLOBAL SUBROUTINE - CHKLOS -
*****
- CHECK FOR LOST CHARACTER ROUTINE -
THIS SUBROUTINE CHECKS FOR THE CONDITION WHICH INDICATES THAT A CHAR
HAS BEEN 'LOST' FROM THE LOOPED BACK DATA PATTERN DURING A TRANSMISSION
AND RECEPTION TEST. IF THIS ROUTINE DETERMINES THAT IT IS LIKELY THAT
A CHARACTER HAS BEEN LOST, IT INDICATES THIS IN THE STATUS INFORMATION
RETURNED TO THE CALLING ROUTINE.

INPUTS:      R3 - RX LINE NUMBER MULTIPLIED BY 2 (OFFSET INTO WORD TABLES).
              R4 - BASE ADDRESS OF RESYNC QUE CONTAINING RX CHARS.
              R5 - MASK OF 'INACTIVE' (NON-DATA) BITS OF RX AND TX CHARS WITH
                  ALL SET BITS IN A SINGLE, LEFT JUSTIFIED GROUP.
              CHCNTB - BASE OF NUMBER OF CHARS TO TX ON EACH LINE TABLE.
              RXCNTB - BASE OF THE RX CHARACTER COUNTERS TABLE.
              RXPTRB - BASE OF THE RX CHARACTER POINTERS TABLE.
              TXRXLB - BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.

OUTPUTS:     CARRY - SET IF LOST CHARACTER CONDITION IS VERIFIED.

CALLING SEQUENCE:  JSR    PC,CHKLOS

COMMENTS:  THE FOLLOWING SYMBOLS ARE USED IN LINE COMMENTS:
            CHR0 - CHARACTER AT BOTTOM OF RESYNC QUE (FIRST RECEIVED).
            CHR1, CHR2 - 2 CHARACTERS RECEIVED AFTER CHR0.
            EXPO - CHARACTER EXPECTED TO BE RECEIVED NEXT.
            EXP1, EXP2 - CHARACTER EXPECTED TO BE RECEIVED AFTER EXPO, ETC.

SUBORDINATE ROUTINES CALLED: NONE.
*****
CHKLOS:: SAVE
            JSR    R5,PREG05 ;SAVE CONTENTS OF GPRS R0 THRU R5.
            ;CALL REGISTER SAVE SUBRT.
            MOV    RXCNTB(R3),R1 ;COMPARE THE PRESENT RX CHARACTER COUNT PLUS 1
            INC    R1 ;WITH THE EXPECTED NUMBER OF CHARS TO RX ON
            MOV    TXRXLB(R3),R0 ;LINE (NUMBER TXED AND LOOPED BACK) TO
            MOV    CHCNTB(R0),R2 ;DETERMINE IF THE POSSIBLE LOST CHAR
            CMP    R1,R2 ;WOULD BE THE LAST EXPECTED RX CHAR.
            BEQ    52$ ;EXIT WITH 'FAILURE' IF LOST CHR WOULD BE LAST.
            INC    R1 ;DETERMINE (AS ABOVE) IF CHR0 WOULD BE THE LAST
            SUB    R2,R1 ;RX CHAR AND SAVE RESULT FOR LATER.
            MOV    RXPTRB(R3),R2 ;GET THE RX DATA POINTER.
            INC    R2 ;CALCULATE POINTER TO EXP1 LOCATION.
            MOVB   (R2)+,R0 ;GET EXP1 VALUE FROM DATA PATTERN.
            SUB    (R4)+,R0 ;COMPARE CHR0 AND EXP1 VALUES.
            BIC    R5,R0 ;REMOVE INACTIVE BITS FROM RESULT. (NO ACTIVE
            ;BITS ALLOWED TO LEFT OF ANY INACTIVE BITS.)
            BNE    52$ ;EXIT WITH 'FAILURE' IF CHR0 <> EXP1.
            TST    R1 ;CHECK CHR0 TEST RESULT SAVED ABOVE.
            BEQ    50$ ;EXIT WITH 'SUCCESS' IF CHR0 IS LAST CHAR.
            MOV    (R4),R1 ;GET CHR1 FROM THE QUE, DATA.VALID INTO N FLAG.
            BPL    50$ ;EXIT WITH 'SUCCESS' IF CHR0 WAS LAST QUE CHAR.
            MOVB   (R2),R0 ;GET THE EXP2 VALUE FROM THE DATA PATTERN.
            SUB    R0,R1 ;COMPARE THE EXP2 AND THE CHR1 VALUES.
            BIC    R5,R1 ;REMOVE INACTIVE BITS FROM RESULT OF COMPARE.
            ;(NO ACTIVE BITS LEFT OF INACTIVE BITS.)
            BNE    52$ ;EXIT WITH 'FAILURE' IF CHR1 <> EXP2.

```

C
C

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 82
GLOBAL SUBROUTINE

- CHKLOS -

```

3482
3483
3484      ;+
3485      ; IT IS LIKELY THAT WE LOST A CHARACTER FROM THE DATA PATTERN.
3486      ; INDICATE 'SUCCESS' AND EXIT.
3487      ;-
3487 014706 000261 50$:      SEC          ;SET THE SUCCESS FLAG.
3488 014710 000401      BR          60$      ;EXIT THE ROUTINE.
3489
3490      ;+
3491      ; WE DIDN'T LOSE A SINGLE EXTRA CHARACTER AT THIS POINT IN THE DATA PATTERN.
3492      ; INDICATE 'FAILURE' AND EXIT.
3493      ;-
3494 014712 000241 52$:      CLC          ;CLEAR THE SUCCESS FLAG.
3495
3496 014714      60$:      PASS          ;RESTORE GPRS.
3497 014714 004736      JSR          PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
3498 014716 000207      RTS          PC      ;CARRY - SET IF SUCCESS (LOST CHAR LIKELY).

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 83

GLOBAL SUBROUTINE

- CHRMSK -

3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536

```

.SBTTL GLOBAL SUBROUTINE - CHRMSK -
++ *****
* - FORM A BIT MASK OF UNUSED TX/RX BITS ROUTINE -
* THIS SUBROUTINE CONSTRUCTS A BIT MASK OF CHARACTER BITS WHICH ARE NOT
* USED DURING TRANSMISSION AND RECEPTION. THIS MASK CAN BE USED
* TO REMOVE THE FLAGS, LINE NUMBER, DATA.VALID BITS, AND UNUSED DATA BITS
* FROM A CHARACTER WORD WHICH HAS BEEN READ FROM THE DUT FIFO.
*
* INPUTS: R1 - DUT LPR CONTENTS USED TO DETERMINE CHARACTER LENGTH.
*
* OUTPUTS: IBM - BIT MASK OF UNUSED TX/RX BITS (INCLUDING UPPER BYTE):
*           EXAMPLES: 177400 RETURNED FOR 8 BITS/CHAR.
*                   177700 RETURNED FOR 6 BITS/CHAR.
*
* CALLING SEQUENCE: JSR PC,CHRMSK
*
* COMMENTS: IF THIS MASK IS TO BE USED TO JUST REMOVE THE INACTIVE BITS
*           WITHIN THE DATA BYTE OF A WORD READ FROM THE DUT FIFO, THE
*           UPPER BYTE OF THE MASK MUST BE CLEARED.
*
* SUBORDINATE ROUTINES CALLED: NONE.
;-- *****

```

```

CHRMSK:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                BIS #177740,R1 ;PREPARE TO COUNT BITS SHIFTED INTO MASK BY
                ; USING THE LPR BITS/CHAR FIELD CONTENTS.
                MOV #177400,R3 ;CLEAR THE UNUSED BIT MAP LOWER BYTE.
2$: ADD #10,R1 ;DETERMINE IF ANOTHER BIT WOULD BE TOO MANY.
        BCS 4$ ;EXIT THE SHIFT LOOP IF IT WOULD BE TOO MANY.
        ASR R3 ;SHIFT A BIT INTO THE UNUSED BIT MASK LOW BYTE.
        BR 2$ ;LOOP TO CHECK FOR DONE.

4$: MOV R3,IBM ;LOAD THE INACTIVE BITS MASK STORAGE IN MEMORY.

60$: PASS

                JSR ;RESTORE GPRS.
                PC,@(SP)+ ;RETURN TO PREG05 SUBRT.

                RTS PC

```

```

014720 004537 005232
014720 052701 177740
014730 012703 177400
014734 062701 000010
014740 103402
014742 006203
014744 000773
014746 010337 002226
014752
014752 004736
014754 000207

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 84
GLOBAL SUBROUTINE - CKCHR -

3537
3538
3539
3540
3541
3542
3543
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
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592

```

.SBTTL GLOBAL SUBROUTINE - CKCHR -
+*****
+ - CHECK CHARACTER FOR ERRORS ROUTINE -
+ THIS SUBROUTINE CHECKS THE CHARACTER AT THE BOTTOM OF THE RESYNC QUEUE
+ TO DETERMINE IF IT IS CORRECT. POINTERS AND COUNTERS WHICH ARE RELATED
+ TO THE RECEPTION OF THE CHARACTER ARE UPDATED. IF THE CHARACTER IS
+ INCORRECT, AN ANALYSIS OF THE ERROR IS DONE AND PARAMETERS ARE SET UP
+ FOR THE REPORTING OF THE CORRECT ERROR.
+
+ INPUTS: R3 - LINE OFFSET FOR ACCESS OF WORD TABLES OF LINE VARIABLES.
+ R4 - BASE ADDRESS OF THE RESYNC QUEUE FOR THIS LINE.
+ R5 - MASK OF THE INACTIVES BITS IN A TX OR RX CHAR BYTE.
+ BITTBL - TABLE OF WORDS WITH BITS SET FOR USE IN FORMING MAPS.
+ DPRSQ - DATA PATTERN RESYNC QUE WITH VALID CHAR AT BOTTOM.
+ EXCNTB - BASE OF THE EXTRA CHARACTER COUNTERS TABLE.
+ RXDNF - RECEIVE DONE FLAGS.
+ RXPTRB - BASE OF THE RX CHARACTER POINTERS TABLE.
+ ERROR MESSAGE LABELS - EM6407,EM6408,EM6427,EM6428
+
+ OUTPUTS: R1 - CONTAINS THE ADDRESS OF THE ERROR MESSAGE TO BE REPORTED.
+ R2 - CONTAINS THE ACTUAL RECEIVED DATA.
+ R4 - CONTAINS THE EXPECTED DATA.
+ CARRY - "SUCCESS" FLAG (SET IF NO ERROR IS FOUND).
+ FOLLOWING VARIABLES UPDATED FOR LINE ON WHICH CHAR WAS RECEIVED:
+ EXCNT - COUNT OF THE NUMBER OF EXTRA CHARS RECEIVED ON LINE.
+ RXCNT - COUNT OF THE NUMBER OF CHARACTERS RECEIVED ON LINE.
+ RXPTR - UPDATED TO POINT TO THE NEXT EXPECTED CHAR ON LINE.
+ ERRBLK - CONTENTS DESTROYED.
+
+ CALLING SEQUENCE: JSR PC,CKCHR
+
+ COMMENTS:
+
+ SUBORDINATE ROUTINES CALLED: CHKEXT,CHKLOS,UPDCHR.
+*****
CKCHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
; JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
+
+ CHECK FOR THE RX OF A CHAR AFTER RX SHOULD BE COMPLETE ON THIS LINE.
+
+ BIT BITTBL(R3),RXDNF ;TEST THE RX DONE FLAG FOR THIS LINE.
+ BEQ 2$ ;SKIP ERROR REPORT IF RX NOT COMPLETE ON LINE.
+
+ WE HAVE RECEIVED AN EXTRA CHARACTER ON THIS LINE.
+ SET UP FOR ERROR REPORT AND EXIT TO REPORT THE ERROR.
+ COUNT THE EXTRA CHARACTER.
+ EXIT TO REPORT 'UNEXPECTED CHAR RECEIVED AFTER RX COMPLETE ON LINE NN'
+
+ MOV #EM9007,R1 ;SELECT 'EXTRA CHAR ON LINE' ERROR MESSAGE.
+ MOV (R4),R2 ;GET THE ACTUAL DATA FOR ERROR REPORT.
+ BIC R5,R2 ;REMOVE THE INACTIVE BITS.
+ BIS #BIT15,R4 ;INDICATE 'NONE' EXPECTED DATA FOR ERROR RPT.
+ BR 12$ ;GO COUNT EXTRA CHAR AND EXIT WITH 'FAILURE'.
+
+ GET THE POINTER TO THE NEXT EXPECTED RECEIVE DATA CHARACTER.
+
+*****

```

```

014756 004537 005232
014762 036337 002332 002412
014770 001407
014772 012701 010250
014776 011402
015000 040502
015002 052704 100000
015006 000452

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 85
GLOBAL SUBROUTINE

- CKCHR -

```

3593 015010 016302 003310 2$:      MOV      RXPTRB(R3),R2
3594
3595      :+
3596      : COMPARE THE ACTUAL DATA WITH THE EXPECTED DATA.
3597      : -
3597 015014 011400      MOV      (R4),R0      ;GET THE ACTUAL DATA.
3598 015016 040500      BIC      R5,R0      ;REMOVE THE INACTIVE BITS.
3599 015020 111201      MOV      (R2),R1      ;GET THE EXPECTED DATA.
3600 015022 040501      BIC      R5,R1      ;REMOVE THE INACTIVE BITS.
3601 015024 120001      CMPB    R0,R1      ;COMPARE ACTUAL AND EXPECTED.
3602 015026 001003      BNE      4$          ;CHECK FURTHER IF DATA MISCOMPARE.
3603 015030 004737 024444 JSR      PC,UPDCHR    ;UPDATE PTRS AND COUNTERS FOR THE CHAR.
3604 015034 000446      BR       50$        ;EXIT WITH 'SUCCESS', NO ERROR FOUND.
3605
3606      :+
3607      : ACTUAL AND EXPECTED DATA MISCOMPARE.
3608      : DETERMINE IF IT'S LIKELY WE RECEIVED AN EXTRA CHAR WITHIN THE DATA PATTERN.
3609 015036 004737 014516 4$:      JSR      PC,CHKEXT    ;CHECK FOR EXTRA CHAR RX'ED IN PATTERN.
3610 015042 103010      BCC      6$          ;GO CHECK FOR LOST CHAR IF NO EXTRA CHAR.
3611
3612      :+
3613      : IT IS LIKELY THAT WE RECEIVED AN EXTRA CHARACTER WITHIN THE DATA PATTERN.
3614      : COUNT THE CHAR AS AN EXTRA CHAR, DON'T COUNT AS A STANDARD CHAR.
3615      : REPORT 'EXTRA CHAR RECEIVED WITHIN DATA PATTERN ON LINE NN'
3616 015044 012701 011063      MOV      #EM9027,R1    ;SELECT 'EXTRA CHAR ON LINE' ERROR MSG.
3617 015050 111200      MOV      (R2),R0      ;GET THE EXPECTED RECEIVE DATA.
3618 015052 040500      BIC      R5,R0      ;REMOVE THE INACTIVE BITS FROM EXPECTED DATA.
3619 015054 011402      MOV      (R4),R2      ;GET THE ACTUAL RECEIVE DATA.
3620 015056 040502      BIC      R5,R2      ;REMOVE THE INACTIVE BITS FROM ACTUAL DATA.
3621 015060 010004      MOV      R0,R4      ;PASS EXPECTED DATA TO ERROR REPORT ROUTINE.
3622 015062 000424      BR       12$        ;GO COUNT EXTRA CHAR AND EXIT WITH 'FAILURE'.
3623
3624      :+
3625      : ACTUAL AND EXPECTED DATA MISCOMPARE.
3626      : NOT LIKELY THAT WE RECEIVED AN EXTRA CHARACTER WITHIN THE DATA PATTERN.
3627      : DETERMINE IF IT'S LIKELY WE LOST A CHARACTER FROM THE DATA PATTERN.
3628 015064 004737 014616 6$:      JSR      PC,CHKLOS    ;CHECK FOR A LOST CHAR CONDITION.
3629 015070 103012      BCC      8$          ;GO REPORT BAD RX DATA IF NOT LOST CHAR.
3630
3631      :+
3632      : IT IS LIKELY THAT WE LOST A CHARACTER FROM THE DATA PATTERN.
3633      : COUNT THE CHAR IN THE RX CHAR COUNT AS IF IT HAD BEEN RECEIVED.
3634      : ALSO, COUNT CHRO AS A VALID CHAR, BECAUSE WE HAVE VERIFIED IT ABOVE.
3635      : REPORT 'SINGLE CHAR MISSING FROM RECEIVED DATA ON LINE NN'
3636 015072 012701 011143      MOV      #EM9028,R1    ;SELECT 'LOST CHAR ON LINE' ERROR MSG. ^^^^^
3637 015076 111200      MOV      (R2),R0      ;GET THE EXPECTED RECEIVE DATA.
3638 015100 040500      BIC      R5,R0      ;REMOVE THE INACTIVE BITS FROM EXPECTED DATA.
3639 015102 011402      MOV      (R4),R2      ;GET THE ACTUAL RECEIVE DATA.
3640 015104 040502      BIC      R5,R2      ;REMOVE THE INACTIVE BITS FROM ACTUAL DATA.
3641 015106 010004      MOV      R0,R4      ;PASS EXPECTED DATA TO ERROR REPORT ROUTINE.
3642 015110 004737 024444 JSR      PC,UPDCHR    ;UPDATE PTRS AND COUNTERS FOR THE CHAR.
3643 015114 000404      BR       10$        ;GO EXIT WITH 'FAILURE'.
3644
3645      :+
3646      : DID NOT LOSE OR GAIN A SINGLE CHARACTER FROM/TO THE DATA PATTERN.
3647      : REPORT 'RECEIVED CHAR MISCOMPARE AGAINST TX DATA ON LINE NN'
3648 015116 010002      8$:      MOV      R0,R2      ;PASS ACTUAL DATUM TO ERROR REPORT ROUTINE.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 86
GLOBAL SUBROUTINE

- CKCHR -

```

3649 015120 010104          MOV   R1,R4          ;PASS EXPECTED DATUM TO ERROR REPORT ROUTINE.
3650 015122 012701 010333  MOV   #EM9008,R1     ;SELECT THE 'DATA MISCOMPARE' MESSAGE.
3651
3652      ;+ UPDATE THE CHARACTER COUNTER AND RX DATA PATTERN POINTER FOR THIS LINE.
3653      ;-
3654 015126 004737 024444 10$: JSR   PC,UPDCHR    ;UPDATE RX PTR AND COUNTER FOR THIS LINE.
3655 015132 000405          BR    14$           ;GO EXIT WITH 'FAILURE'.
3656      ;+
3657      ;: COUNT THE CHARACTER AS AN EXTRA CHARACTER.
3658      ;-
3659 015134 005263 003150 12$: INC   EXCNTB(R3)   ;INCREMENT THE EXTRA CHAR COUNT FOR THIS LINE.
3660 015140 001002          BNE   14$           ;EXIT WITH FAILURE IF NO OVERFLOW.
3661 015142 005363 003150          DEC   EXCNTB(R3)   ;DECREMENT BACK TO -1 (MAX VALUE) IF OVERFLOW.
3662      ;+
3663      ;: INDICATE 'FAILURE' AND EXIT.
3664      ;-
3665 015146 000241          14$: CLC                   ;CLEAR THE 'SUCCESS' FLAG.
3666 015150 000401          BR    60$           ;EXIT THE ROUTINE.
3667
3668      ;+
3669      ;: NO ERROR WAS FOUND.
3670      ;: SET 'SUCCESS' FLAG AND EXIT.
3671      ;-
3672 015152 000261          50$: SEC                   ;SET THE 'SUCCESS' FLAG.
3673
3674 015154          60$: PASS   R1,R2,R4      ;RESTORE GPRS, EXCEPT
3675 015154 010166 000004          MOV   R1,R1SLOT(SP)   ;PUT R1 IN STACK SLOT.
3676 015160 010266 000006          MOV   R2,R2SLOT(SP)   ;PUT R2 IN STACK SLOT.
3677 015164 010466 000012          MOV   R4,R4SLOT(SP)   ;PUT R4 IN STACK SLOT.
3678 015170 004736          JSR   PC,@(SP)+      ;RETURN TO PREG05 SUBRT.
3679
3680      ;: R1 - CONTAINS THE ADDRESS OF THE ERROR REPORT.
3681      ;: R2 - CONTAINS THE ACTUAL DATA RECEIVED.
3682 015172 000207          RTS   PC          ;R4 - CONTAINS THE EXPECTED DATA.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 87
GLOBAL SUBROUTINE

- CKFRPR -

3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3710
3711
3712
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

015174
015174 004537 005232
015200 013704 005224
015204 004737 023642

015210 013701 002242
015214 023737 002410 002174 2\$:
015222 001402
015224 062701 000062
015230 052701 170000 4\$:
015234 013702 002204
015240 004737 024740
015244 103033

015246 005337 002404
015252 001011
015254 010437 005224
015260 012701 010743
015264 012737 012230 005230

015272

```
.SBTTL GLOBAL SUBROUTINE - CKFRPR -
*****
- CHECK FRAMING AND PARITY ERROR REPORTING -
THIS SUBROUTINE IS USED IN THE FRAMING ERROR AND PARITY ERROR TESTS.
IT READS THE CHARACTERS FROM THE DUT RECEIVER CHARACTER FIFO,
AND CHECKS FOR THE CORRECT COMBINATION OF PARITY AND FRAMING
ERROR BITS IN THE MSB. IF CHARACTERS STOP APPEARING IN THE FIFO WITH
DATA.VALID SET OR IF MORE THAN THE ALLOWABLE NUMBER OF CHARACTERS
HAS BEEN READ FROM THE DUT THIS ROUTINE EXITS WITH AN RX COMPLETE
INDICATION. EACH READ CHAR IS ANALYSED AND ANY NECESSARY ERRORS ARE
REPORTED.

INPUTS: R5 - TEST FLAG, BIT15 SET = FRAMING ERR, CLEAR = PARITY ERR.
ERRNBR - SET TO ERROR NUMBER OF FIRST ERROR IN THIS ROUTINE.
OSTEND - ADDRESS OF THE END OF THE OUTPUT STORAGE FIFO BUFFER.
OSTPTR - POINTER TO THE NEXT BYTE TO READ FROM OSTORE.

OUTPUTS: RXCNTB - RECEIVE CHARACTER COUNT UPDATED FOR EACH LINE.
RXPNTB - RECEIVE CHARACTER PIONTER IS UPDATED FOR EACH LINE.

CALLING SEQUENCE: JSR PC,CKFRPR

COMMENTS: THIS ROUTINE REPORTS ERRORS WITH NUMBERS INITIAL ERRNBR
THRU INITIAL ERRNBR + 4.
ERRNBR IS RESTORED BEFORE THIS ROUTINE RETURNS.

SUBORDINATE ROUTINES CALLED: PRFRME,PRPARE,WAIBIS.
*****
CKFRPR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV ERRNBR,R4 ;PRESERVE THE INITIAL ERROR NUMBER.
JSR PC,TXIE1 ;ENABLE TX INTERRUPTS.

;+
; WAIT FOR A CHARACTER TO APPEAR IN THE FIFO.
; IF NO CHARACTER APPEARS WITHIN TIME-OUT PERIOD: EXIT ROUTINE, WE'RE DONE.
;-
MOV RXTOUT,R1 ;GET MINIMUM TIME OUT VALUE.
CMP TXDONF,ACTLNS ;CHECK FOR TRANSMISSION DONE ON ACTIVE LINES.
BEQ 4$ ;SKIP ADDING 50 MS DELAY IF TX DONE ALL LINES.
ADD #50,R1 ;ADD 50 MILLI SEC TO DELAY IF NOT LAST CHAR.
BIS #170000,R1 ;INDICATE TO TEST DATA.VALID BIT.
MOV RBUFA,R2 ;INDICATE TO CHECK DUT RECEIVE BUFFER (FIFO).
JSR PC,WAIBIS ;WAIT FOR RECEIVED CHAR OR TIME-OUT.
BCC 60$ ;EXIT ROUTINE IF TIME-OUT, WE'RE DONE.

DEC CHRTOT ;DECREMENT THE TOTAL CHAR COUNTER.
BNE 6$ ;SKIP ERROR IF NOT TOO MANY CHARS RECEIVED.
MOV R4,ERRNBR ;SET ERROR NUMBER TO INITIAL ERRNBR.
MOV #EM9025,R1 ;SELECT THE ERROR MESSAGE TO BE REPORTED.
MOV #ER0503,ERRBLK ;SELECT THE ERROR REPORT ROUTINE.

;+
REPORT ERROR AT INITIAL ERRNBR.
'MORE THAN TWICE THE EXPECTED NUMBER OF CHARACTERS RECEIVED'
;-
ERROR ; >>>> ERROR <<<<.
```


CVDHCA0 DHV-11 FUNC TST PART1
 CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 88
 GLOBAL SUBROUTINE - CKFRPR -

```

3739 015272 104460
3740 015274 000417          BR      60$          ;EXIT THIS ROUTINE WE HAVE GIVEN UP. TRAP C$ERROR
3741
3742
3743      ;+
3744      ; EXTRACT THE LINE NUMBER OF THE NEW CHARACTER.
3745      ; CALCULATE OFFSET FOR ACCESSING TABLES OF LINE VARIABLES.
3746 015276 010203      6$:      MOV      R2,R3          ;COPY THE READ CHARACTER.
3747 015300 000303          SWAB     R3          ;GET THE LINE NUMBER IN THE LSB.
3748 015302 042703 177760      BIC      #177760,R3      ;CLEAR THE UNWANTED BITS.
3749 015306 006303          ASL      R3          ;SHIFT LEFT TO FORM OFFSET INTO TABLES.
3750
3751      ;+
3752      ; PROCESS THE READ CHARACTERS AS DICTATED BY THE TEST FLAG.
3753 015310 010505          ;-
3754 015312 100003          MOV      R5,R5          ;DETERMINE WHICH TEST CALLED THIS ROUTINE.
3755      BPL      8$          ;BRANCH TO PROCESS CHARACTER IN PARITY TEST.
3756 015314 004737 020252      JSR      PC,PRFRME      ;PROCESS FRAMING ERRORS RECEIVED.
3757 015320 000402          BR      10$           ;SKIP PROCESSING CHARACTERS FOR PARITY TEST.
3758 015322 004737 020350      8$:      JSR      PC,PRPARE      ;PROCESS PARITY ERRORS RECEIVED.
3759
3760 015326 004737 024444      10$:     JSR      PC,UPDCHR      ;UPDATE POINTERS AND COUNTERS FOR THIS LINE.
3761 015332 000730          BR      2$          ;LOOP TO READ NEXT CHAR FROM FIFO.
3762
3763 015334 010437 005224      60$:     MOV      R4,ERRNBR      ;RESTORE THE ERROR NUMBER TO ITS INITIAL VALUE.
3764 015340          PASS          ;RESTORE GPRS.
3765 015340 004736          JSR      PC,@(SP)+      ;RETURN TO PREG05 SUBRT.
3766 015342 000207          RTS      PC
    
```

CV
 CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 89
GLOBAL SUBROUTINE - CKINAC -

CV
CV

3767
3758
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796 015344
3797 015344 004537 005232
3798
3799
3800
3801
3802 015350 010203
3803 015352 000303
3804 015354 042703 177760
3805 015360 006303
3806
3807
3808
3809 015362 005702
3810 015364 100021
3811
3812
3813
3814
3815 015366 016301 005142
3816 015372 036137 002332 002174
3817 015400 001013
3818
3819
3820
3821
3822

```
.SBTTL GLOBAL SUBROUTINE - CKINAC -
:++ *****
: - CHECK FOR NEW CHARACTER ON INACTIVE LINE ROUTINE -
: THIS SUBROUTINE CHECKS A CHARACTER TO DETERMINE IF THE CHARACTER
: WAS RECEIVED ON AN ACTIVE LINE. IF THE CHARACTER WAS RECEIVED ON
: AN INACTIVE LINE THIS ROUTINE RECORDS THE FACT THAT THE CHARACTER
: WAS RECEIVED ON AN INACTIVE LINE, PREPARES AN ERROR MESSAGE FOR
: THE CALLING ROUTINE, AND RETURNS A 'FAILURE' STATUS.
:
: INPUTS: R2 - THE RX CHARACTER INCLUDING ERROR FLAGS AND LINE NUMBER.
: ACTLNS - BIT MAP OF ACTIVE DUT LINES.
: BITTBL - TABLE OF WORDS WITH BITS SET FOR FORMING BIT MAPS.
: EM9006 - LABEL AT 'RX ON INACTIVE LINE' ERROR MESSAGE.
: EXCNTB - BASE OF THE EXTRA CHARACTER COUNTERS TABLE.
: TXRXLB - BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
:
: OUTPUTS: CARRY - 'SUCCESS' FLAG (SET IF NO ERROR FOUND).
: R1 - IF ERROR FOUND, ADDRESS OF ERROR MESSAGE.
: R3 - LINE NUMBER OFFSET OF PASSED IN CHARACTER.
: R4 - IF ERROR FOUND, EXPECTED DATA INDICATION FOR ERROR RPT.
: EXCNT - EXTRA CHARACTER COUNT FOR LINE (UPDATED IF ERROR).
:
: CALLING SEQUENCE: JSR PC,CKINAC
:
: COMMENTS:
:
: SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
CKINAC:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;CALL REGISTER SAVE SUBRT.
:++
: EXTRACT THE LINE NUMBER FROM THE PASSED IN CHARACTER AND USE THE LINE
: NUMBER TO FORM AN OFFSET FOR ACCESSING TABLES OF LINE VARIABLES.
:--
MOV R2,R3 ;EXTRACT THE LINE NUMBER
SWAB R3 ; FROM THE CHARACTER WE
BIC #177760,R3 ; ARE COMPARING.
ASL R3 ;FORM OFFSET INTO WORD TABLE FROM LINE NUMBER.
:++
: IF THE CHARACTER IN QUESTION IS NOT A VALID CHARACTER, EXIT WITH 'SUCCESS'.
:--
TST R2 ;CHECK DATA.VALID BIT.
BPL 50$ ;EXIT WITH SUCCESS IF CHAR IS NOT VALID.
:++
: IF THE TX LINE WHICH IS ASSOCIATED WITH THIS RX LINE IS AN ACTIVE LINE,
: EXIT THE ROUTINE WITH 'SUCCESS'.
:--
MOV TXRXLB(R3),R1 ;GET THE TX LINE # OFFSET FOR THIS RX LINE.
BIT BITTBL(R1),ACTLNS ;DETERMINE IF TX LINE IS AN ACTIVE LINE.
BNE 50$ ;EXIT ROUTINE WITH SUCCESS IF LINE IS ACTIVE.
:++
: THE CHARACTER IN QUESTION WAS RECEIVED ON AN INACTIVE LINE.
: COUNT THIS CHARACTER AS AN EXTRA CHAR.
: SET UP ERROR INFORMATION.
: EXIT ROUTINE WITH 'FAILURE' INDICATION.
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 90

GLOBAL SUBROUTINE

- CKINAC -

```

3823      ; -
3824 015402 005263 003150      INC   EXCNTB(R3)      ; INCREMENT THE EXTRA CHAR COUNT FOR THIS LINE.
3825 015406 001002              BNE   2$              ; SKIP SETTING TO MAX VALUE IF NO OVERFLOW.
3826 015410 005363 003150      DEC   EXCNTB(R3)      ; DECREMENT BACK TO -1 (MAX VALUE) IF OVERFLOW.
3827 015414 012701 010175      2$:  MOV   #EM9006,R1    ; SET UP RX ON INACTIVE LINE MESSAGE.
3828 015420 012704 100000      MOV   #BIT15,R4      ; SET UP 'NONE' EXPECTED D' A INDICATION.
3829 015424 000241              CLC                   ; CLEAR THE 'SUCCESS' FLAG.
3830 015426 000401              BR    60$             ; GO REPORT RX CHAR ON INACTIVE LINE.
3831
3832      ; +
3833      ; WE HAVE NOT FOUND A 'CHAR ON INACTIVE LINE' ERROR SITUATION.
3834      ; SET THE 'SUCCESS' FLAG AND EXIT THE ROUTINE.
3835      ; -
3836 015430 000261      50$:  SEC                   ; SET THE 'SUCCESS' FLAG.
3837
3838 015432              60$:  PASS   R1,R3,R4      ; RESTORE GPRS, EXCEPT OUTPUT GPRS.
3839 015432 010166 000004      MOV   R1,R1SLOT(SP)  ; PUT R1 IN STACK SLOT.
3840 015436 010366 000010      MOV   R3,R3SLOT(SP)  ; PUT R3 IN STACK SLOT.
3841 015442 010466 000012      MOV   R4,R4SLOT(SP)  ; PUT R4 IN STACK SLOT.
3842 015446 004736              JSR   PC,@(SP)+      ; RETURN TO PREG05 SUBRT.
3843 015450 000207      RTS   PC            ; CARRY - SUCCESS FLAG (SET IF NO ERROR).

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 91
GLOBAL SUBROUTINE - CKTRAP -

3844			
3845			
3846			
3847			
3848			
3849			
3850			
3851			
3852			
3853			
3854			
3855			
3856			
3857			
3858			
3859			
3860			
3861			
3862			
3863			
3864			
3865			
3866			
3867	015452		
3868	015452	004537	005232
3869	015456	005037	002256
3870	015462	011011	
3871	015464	005737	002256
3872	015470	000261	
3873	015472	001401	
3874	015474	000241	
3875	015476		
3876	015476	004736	
3877	015500	000207	

```

.SBTTL GLOBAL SUBROUTINE - CKTRAP -
*****
* CHECK TRAP ROUTINE -
* THIS SUBROUTINE IS USED TO CHECK FOR A BUS TIME-OUT TRAP (004 TRAP)
* WHICH IS CAUSED BY AN ACCESS TO A NON-EXISTENT MEMORY OR I/O LOCATION.
* IF THE TRAP DOES NOT OCCUR, THIS ROUTINE RETURNS A SUCCESS INDICATION.
*
* INPUTS: R0 - SOURCE ADDRESS FOR MOVE.
* R1 - DESTINATION ADDRESS FOR MOVE.
* (R0) - SOURCE FOR THE MOVE.
*
* OUTPUTS: (R1) - WRITTEN TO THE CONTENTS OF (R0).
* CARRY FLAG - SET ON RETURN IF NO 004 TRAP DETECTED.
* TP4FLG - NONZERO IF TRAP OCCURRED, CLEARED OTHERWISE.
*
* CALLING SEQUENCE: JSR PC,CKTRAP
*
* COMMENTS: IF THIS SUBROUTINE CAUSES A TRAP, EITHER THE ADDRESS WHICH
* IS LABELED ADRPTR WILL BE THE TRAP PC ADDRESS ON THE STACK.
*
* SUBORDINATE ROUTINES CALLED: NONE.
*****
CKTRAP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
CLR TP4FLG JSR ;CLEAR THE 004 TRAP FLAGS.
MOV (R0),(R1) ;PERFORM THE MOVE IN QUESTION.
ADRPTR:: TST TP4FLG ;CHECK FOR OCCURENCE OF TRAP.
SEC ;INDICATE SUCCESS.
BEQ 60$ ;EXIT WITH SUCCESS IF TRAP DID NOT OCCUR.
CLC ;INDICATE FAILURE.
60$: PASS ;RESTORE GPRS.
RTS PC JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 92
GLOBAL SUBROUTINE - CLNRST -

3878
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909
3910
3911
3912
3913
3914
3915
3916
3917
3918
3919
3920
3921
3922
3923

015502
015502 004537 005232

015506 004737 021712
015512 103002

015514 004737 020612

015520
015520
015520 004736
015522 000207

```

.SBTTL GLOBAL SUBROUTINE - CLNRST -
*****
- CLEAN RESET OF THE DEVICE UNDER TEST -
THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
THE DUT'S SELF-TEST IS SKIPPED, AND THE FIFO IS PURGED OF ANY ERROR
CODES, ETC.
IF THE RESET DOES NOT SUCCESSFULLY COMPLETE, THEN THE CARRY BIT IS
PASSED BACK TO THE CALLING ROUTINE (CLEAR).

INPUTS: CSRA - CONTAINS THE ADDRESS OF THE CSR
TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
ERRNBR - ERROR NUMBER FOR POSSIBLE ERROR REPORT.
ERRTBL- ERRTP,ERNBR,AND ERRMSG SET UP CORRECTLY.

OUTPUTS: THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
ERRBLK - VALUE MAY BE DESTROYED.
IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.

CALLING SEQUENCE: JSR PC,CLNRST

COMMENTS: THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS ERRNBR.
THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.

SUBORDINATE ROUTINES CALLED: DELAY,MSLGET,PUFIFO,RESETT.
*****
CLNRST:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.

;+
; RESET THE DUT.
; THIS ROUTINE REPORTS ERRORS WITH NUMBERS FROM ERRNBR THRU ERRNBR+2.
;-
JSR PC,RESETT ;RESET THE DUT TO A KNOWN STATE.
BCC 60$ ;EXIT ROUTINE WITH ABORT TEST INDICATOR.

;+
; PURGE THE FIFO OF ERROR CODES, SAVE ANY BMP CODES FOUND.
;-
JSR PC,PUFIFO ;PURGE THE FIFO.

60$: ;EXIT THE TEST USING RESETT OR PUFIFO STATUS.
PASS ;RESTORE GPRS, PASS THE FOLLOWING INTACT:
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
;CARRY BIT:IF CLEAR, THEN ABORT THE TEST.

RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 93

GLOBAL SUBROUTINE

- CLR16W -

3924
3925
3926
3927
3928
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948

015524
015524 004537 005232
015530 012701 000020
015534 005020
015536 005301
015540 001375
015542
015542 004736
015544 000207

```

.SBTTL GLOBAL SUBROUTINE - CLR16W -
:++ *****
: * - CLEAR SIXTEEN WORDS ROUTINE -
: * THIS SUBROUTINE CLEARS 16 WORDS STARTING WITH THE SPECIFIED WORD.
: * INPUTS: R0 - ADDRESS OF THE FIRST WORD TO CLEAR.
: * OUTPUTS: (R0) TO (R0+15) - 16 WORDS OF MEMORY ARE CLEARED TO 0.
: * CALLING SEQUENCE: JSR PC,CLR16W
: * COMMENTS:
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****

CLR16W:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
R5,PREG05 ;CALL REGISTER SAVE SUBRT.
2$: MOV #16,R1 ;SET THE LOOP COUNTER TO 16.
CLR (R0)+ ;CLEAR A WORD OF MEMORY.
DEC R1 ;COUNT THIS LOOP.
BNE 2$ ;LOOP IF NOT 16 WORD CLEARED.
60$: PASS ;RESTORE GPRS.
JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC

```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 94
GLOBAL SUBROUTINE - CONMAP -

3949
3950
3951
3952
3953
3954
3955
3956
3957
3958
3959
3960
3961
3962
3963
3964
3965
3966
3967
3968
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978
3979
3980
3981
3982
3983
3984
3985
3986
3987
3988

015546
015546 004537 005232
015552 012702 005142
015556 010503
015560 012704 000010
015564 005005
015566 006203
015570 103005
015572 011201
015574 006201
015576 004737 017260
015602 050005
015604 005722
015606 005304
015610 001366
015612
015612 010566 000014
015616 004736
015620 000207

```

.SBTTL GLOBAL SUBROUTINE - CONMAP -
:++ *****
: * - CONVERT LINE BIT MAP.
: * THIS SUBROUTINE IS USED TO CONVERT A BIT MAP PASSED TO IT , INTO
: * ANOTHER LINE BIT MAP THAT IS BASED UPON THE ASSOCIATED TX/RX LINE
: * NUMBER/OFFSET TABLE.
: * INPUTS: R5 - CONTAINS THE LINE BIT MAP TO BE TRANSFORMED.
: * TXRXLB - BASE ADDRESS OF ASSOCIATED TX/RX LINE NUMBER TABLE.
: * OUTPUTS: R5 - CONTAINS AN ASSOCIATED LINE BIT MAP.
: * CALLING SEQUENCE: JSR PC,CONMAP
: * COMMENTS: THE TX/RX ASSOCIATION TABLE MUST BE INITIALISED BEFORE THIS
: * ROUTINE IS CALLED.
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****

CONMAP::SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                MOV #TXRXLB,R2 ;GET THE BASE ADDRESS OF THE LINE ASSOC TABLE.
                MOV R5,R3 ;COPY THE BIT MAP TO BE TRANSFORMED.
                MOV #NUMLNS,R4 ;SET MAX LINE COUNTER.
                CLR R5 ;CLEAR ASSOCIATED LINE BIT MAP.
2$: ASR R3 ;SHIFT ACTLNS BIT MAP INT BOOLEAN REGISTER.
    BCC 4$ ;SKIP SETTING ASSOCIATED LINE NUMBER BIT MAP.
    MOV (R2),R1 ;GET ASSOCIATED LINE NUMBER OFFSET FROM TABLE.
    ASR R1 ;SHIFT RIGHT TO GET LINE NUMB FROM OFFSET.
    JSR PC,LINBIT ;GENERATE AN SINGLE BIT MAP FOR THIS LINE.
    BIS R0,R5 ;SET BIT FOR THIS LINE IN ASSOCIATED BIT MAP.
    TST (R2)+ ;INCREMENT ADDRESS FOR THE NEXT LINE NUMBER.
    DEC R4 ;DECREMENT LINE COUNT.
    BNE 2$ ;LOOP IF NOT DONE.
60$: PASS R5 ;RESTORE GPRS, EXCEPT
                MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
                JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
                ;R5 - CONTAINS THE ASSOCIATED LINE BIT MAP.

                RTS PC

```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.F11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 95

GLOBAL SUBROUTINE - DELAY -

3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
4000
4001
4002
4003
4004
4005
4006
4007
4008
4009
4010
4011
4012
4013
4014
4015
4016
4017
4018
4019
4020

015622
015622 004537 005232
015626 010401
015630 012702 177777
015634 005003
015636 012704 015660
015642 004737 017454
015650 004737 020024
015654
015654 004736
015656 000207
015660 177777

```

.SBTTL GLOBAL SUBROUTINE - DELAY -
*****
- DELAY SUBROUTINE -
THIS SUBROUTINE IS USED TO DELAY A VARIABLE NUMBER OF MILLI-SECONDS.
INPUTS: R4 - CONTAINS THE NUMBER OF MS TO DELAY.
        MSLCNT.
OUTPUTS: NONE.
CALLING SEQUENCE: JSR PC,DELAY
COMMENTS: IF NO HARDWARE CLOCK INTERRUPTS ARE OCCURING, CONTROL-CS WILL
          NOT BE HONORED FOR THE DURATION OF THE DELAY.
SUBORDINATE ROUTINES CALLED: NONE.
*****
DELAY:: SAVE                ;SAVE CONTENTS OF GPRS R0 THRU R5.
                            R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                            ;PASS NUMBER OF MS DELAY AS TIME-OUT VALUE.
                            MOV R4,R1 ;TELL MSLOOP ROUTINE TO CHECK ALL BITS.
                            #-1,R2 ;TELL MSLOOP RTN TO CHECK FOR ALL BITS CLEAR.
                            CLR R3 ;TELL MSLOOP TO CHECK DUMMY NON-ZERO WORD.
                            MOV #62$,R4 ;DELAY THE REQUESTED # OF MS.
                            JSR PC,MSLOOP ;EXIT ROUTINE IF WE TIMED-OUT.]
                            BCC 60$ ;IF NO TIME-OUT, BAD PROGRAM OR HOST MACHINE.
                            JSR PC,OOPS ;RESTORE GPRS.
60$: PASS                    ;RETURN TO PREG05 SUBRT.
                            JSR PC,@(SP)+
                            RTS PC
62$: .WORD -1                ;DUMMY, NON-ZERO WORD.

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 96

GLOBAL SUBROUTINE

- DODMA -

```

4021 .SBTTL GLOBAL SUBROUTINE - DODMA -
4022 :+ *****
4023 :* - INITIATE DMA TRANSMISSION ROUTINE -
4024 :* THIS ROUTINE WRITES THE DMA PARAMETER TO THE SPECIFIED DEVICE AND
4025 :* INITIATES THE DMA TRANSMISSION.
4026 :*
4027 :* INPUTS: R1 - LINE NUMBER ON WHICH TO INITIATE THE DMA.
4028 :* R2 - START ADDRESS OF THE DMA BUFFER (16 BIT VIRTUAL).
4029 :* R3 - CHARACTER COUNT OF THE DMA BUFFER.
4030 :* CSRA - CONTAINS ADDRESS OF THE DUT CSR.
4031 :* IESTAT - STORAGE FOR STATES OF THE INTERRUPT ENABLE BITS.
4032 :* MMENAB - MEMORY MANAGEMENT FLAG (0 IF MEM MGT NOT ENABLED).
4033 :* HOST MEM MGT PAR REGISTERS - IF MEM MGT IS IN USE.
4034 :* TXAD1A - CONTAINS ADDRESS OF DMA TX BUFFER ADDRESS REG #1.
4035 :* TXAD2A - CONTAINS ADDRESS OF DMA TX BUFFER ADDRESS REG #2.
4036 :* TXBFCA - CONTAINS ADDRESS OF DMA CHARACTER COUNT REGISTER.
4037 :*
4038 :* OUTPUTS: CARRY - SUCCESS FLAG (SET IF DMA START FOUND CLEAR).
4039 :* DUT TBUFFAD1 - LS 16 BITS OF DMA BUFFER ADDRESS (INITIALIZED).
4040 :* DUT TBUFFAD2 - MS 6 BITS OF DMA BUFFER ADDRESS (INITIALIZED),
4041 :* DMA START BIT SET.
4042 :* DUT TBUFFCT - DMA BUFFER CHARACTER COUNT (INITIALIZED).
4043 :*
4044 :* CALLING SEQUENCE: JSR PC,DODMA
4045 :*
4046 :* COMMENTS: THIS ROUTINE DETERMINES IF MEMORY MANAGEMENT IS BEING USED
4047 :* AND SETS UP THE FULL 22 BIT PHYSICAL ADDRESS IF NECESSARY.
4048 :*
4049 :* SUBORDINATE ROUTINES CALLED: NONE.
4050 :-- *****
4051
4052 015662 DODMA:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
4053 015662 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4054 015666 012704 000200 MOV #200,R4 ;PREPARE TO CLEAR UPPER 6 BITS OF DMA BUFF ADR.
4055 015672 005737 002310 TST MMENAB ;CHECK FOR MEMORY MANAGEMENT IN USE.
4056 015676 001427 BEQ 6$ ;GOTO SET UP DEVICE IF MEM MGT NOT IN USE.
4057
4058 :+
4059 :* MEMORY MANAGEMENT IS IN USE.
4060 :* CONSTRUCT 22 BIT PHYSICAL ADDRESS FROM THE 16 BIT VIRTUAL ADDRESS.
4061 :--
4061 015700 010205 MOV R2,R5 ;STRIP THE MOST SIGNIFICANT 3 BITS OF THE
4062 015702 012700 000005 MOV #5,R0 ;DMA BUFFER VIRTUAL ADDRESS AND MULTIPLY
4063 015706 006105 2$: ROL R5 ;THEIR VALUE BY TWO TO GET AN OFFSET INTO
4064 015710 005300 DEC R0 ;THE TABLE OF MEMORY MANAGEMENT PAGE
4065 015712 001375 BNE 2$ ;ADDRESS REGISTERS (PAR).
4066 015714 042705 177761 BIC #177761,R5 ;
4067 015720 063705 002312 ADD PAR0A,R5 ;ADD IN THE BASE VALUE OF THE MM PAR REGISTERS.
4068 015724 011505 MOV (R5),R5 ;GET THE 16 BIT PHYSICAL ADDRESS BLOCK COUNT.
4069 015726 012700 000006 MOV #6,R0 ;SHIFT UPPER 6 BITS OF THE PHYSICAL ADDRESS
4070 015732 006305 4$: ASL R5 ;BLOCK COUNT (GOTTEN FROM THE PROPER PAR)
4071 015734 006104 ROL R4 ;INTO THE LS 6 BITS OF THE WORD TO WRITE
4072 015736 005300 DEC R0 ;INTO THE DUT TBUFFAD2 REGISTER.
4073 015740 001374 BNE 4$ ;
4074 015742 042702 160000 BIC #160000,R2 ;ADD THE 13 BIT DISPLACEMENT FIELD FROM VIRTUAL
4075 015746 060502 ADD R5,R2 ;ADR TO THE SHIFTED BLOCK NUMBER FROM THE
4076 015750 005504 ADC R4 ;MEMORY MANAGEMENT PAR.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 97
GLOBAL SUBROUTINE - DODMA -

```

4077 015752 052704 000200          BIS    #200,R4          ;SET THE DMA_START BIT IN WORD FOR TBUFFAD2.
4078
4079      ;+
4080      ;WRITE THE DMA PARAMETERS OUT TO THE DUT DMA REGISTERS.
4081      ;DISABLE INTERRUPTS.
4082      ;SET UP DUT CSR IND.ADR.REG FIELD.
4083      ;WRITE THE DMA TRANSMIT CHARACTER COUNT.
4084      ;WRITE THE LEAST SIGNIFICANT 16 BITS OF THE DMA BUFFER START ADDRESS.
4085      ;WRITE THE MOST SIGNIFICANT 6 BITS OF THE ADDRESS,
4086      ;SETTING THE DMA_START BIT, AND INITIATING THE DMA TRANSMISSION.
4087 015756      6S:  GETPRI  R5          ;GET THE PRESENT PROCESSOR PRIORITY.
4088 015756      TRAP          C$GPRI
4089 015760      MOV          RO,R5
4090      SETPRI  #PRI07          ;DISABLE ALL HARDWARE INTERRUPTS.
4091 015762      MOV          #PRI07,RO
4092 015766      TRAP          C$SPRI
4093 015770      BIS    IESTAT,R1          ;PREPARE FOR SETUP OF LINE NUMBER IN DUT CSR.
4094 015774      MOV    R1,@CSRA          ;SET UP THE DUT CSR IND.ADR.REG FIELD.
4095 016000      TSTB  @TXAD2A          ;TEST THE DUT DMA_START BIT.
4096 016004      CLC                    ;INDICATE FAILURE IN CASE DMA.HO BIT IS SET.
4097 016006      BMI    60S            ;EXIT WITH FAILURE IF DMA.HO BIT IS SET.
4098 016010      MOV    R3,@TXBFCA          ;WRITE THE DMA CHARACTER COUNT.
4099 016014      MOV    R2,@TXAD1A          ;WRITE THE LS 16 BITS OF BUFFER ADDRESS.
4100 016020      MOVB  R4,@TXAD2A          ;WRITE MS 6 BITS OF ADR AND START DMA TX.
4101 016024      SETPRI  R5          ;RESTORE THE PROCESSOR PRIORITY.
4102 016024      MOV    R5,RO
4103 016026      TRAP          C$SPRI
4104 016030      SEC                    ;INDICATE SUCCESS.
4105
4106 016032      60S:  PASS
4107 016032      JSR          PC,@(SP)+          ;RESTORE GPRS,
4108 016034      RTS    PC          ;RETURN TO PREG05 SUBRT.
          ; CARRY - SUCCESS FLAG (SET IF SUCCESS).

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 98
GLOBAL SUBROUTINE

- FINACT -

4109
4110
4111
4112
4113
4114
4115
4116
4117
4118
4119
4120
4121
4122
4123
4124
4125
4126
4127
4128
4129
4130
4131
4132
4133
4134
4135
4136
4137
4138
4139
4140
4141
4142
4143
4144
4145
4146
4147
4148
4149
4150
4151
4152
4153
4154

```

.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 R5,PREG05 ;SAVE CONTENTS OF GPRS R0 THRU R5.
;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.
2$: BIT R5,R0 ;LOOK FOR AN ACTIVE LINE.
BNE 4$ ;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 2$ ;LOOP TO TRY THE NEXT LINE.
CLC ;CLEAR CARRY BIT, NO ACTIVE LINE FOUND.
BR 60$ ;EXIT WITH FAILURE.
4$: 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

```

016036 004537 005232

016042 005001

016044 012703 000010

016050 013700 002174

016054 012705 000001

016102 010166 000004

016106 010566 000014

016112 004736

016114 000207

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 99
GLOBAL SUBROUTINE

- FRPSUP -

4155
4156
4157
4158
4159
4160
4161
4162
4163
4164
4165
4166
4167
4168
4169
4170
4171
4172
4173
4174
4175
4176
4177
4178
4179
4180
4181
4182
4183
4184
4185
4186
4187
4188
4189
4190
4191
4192
4193
4194
4195
4196
4197
4198
4199
4200
4201
4202
4203
4204
4205
4206
4207
4208
4209
4210

```

.SBTTL GLOBAL SUBROUTINE - FRPSUP -
+*****
+ - FRAMING AND PARITY ERROR TRANSMISSION/RECEPTION SET-UP -
+
+ THIS ROUTINE IS USED TO INITIALISE BOTH THE DUT AND THE
+ TRANSMISSION/RECEPTION CONTROL PARAMETERS TO THE CORRECT
+ STATE, PRIOR TO A FRAMING OR PARITY ERROR DETECTION AND
+ REPORTING TEST.
+
+ INPUTS: R0 - LPR CONTENTS FOR LINES IN THE BIT MAP IN GPR4.
+ R1 - LPR CONTENTS FOR LINES NOT IN THE BIT MAP IN GPR4.
+ R2 - START ADDRESS OF DATA PATTERN TO TRANSMIT.
+ R3 - LENGTH OF THE DATA PATTERN TO TX.
+ R4 - LOCAL LINE GROUP BIT MAP.
+ ACTLNS - CONTAINS A BIT MAP OF ALL CURRENTLY ACTIVE LINES.
+ LOPBCK - CONTAINS THE TYPE OF LOOPBACK MODE SELECTED.
+ CBB - LABEL AT BASE OF TX/RX CONTROL BLOCK.
+
+ OUTPUTS: THE CONTENTS OF THE TXRCB ARE DESTROYED.
+ THE INDIRECT ADDRESS FIELD OF THE DUT CSR MAY BE DESTROYED.
+ THE DUT'S LPR'S AND LNC'S MAY BE MODIFIED.
+ THE FOLLOWING POINTERS AND COUNTERS ARE INITIALISED:
+ CHCNT,CHRTOT,DPEND,DPLEN,EXCNT,RXCNT,RXDONF,RXPTR,TXCNT,
+ TXDONF,TXPTR,TXRXL.
+
+ CALLING SEQUENCE: JSR PC,FRPSUP
+
+ COMMENTS: THIS ROUTINE SHOULD BE CALLED TWICE DURING THE TESTING OF
+ THE FRAMING AND PARITY ERROR DETECTION AND REPORTING TEST.
+ SO THAT BOTH LINE GROUPS ARE TESTED ON TRANSMISSION AND
+ RECEPTION.
+ JSR PC,FRPSUP ; DO SET-UP.
+ EXECUTE TEST FOR THE ABOVE SET-UP.
+ COMPLEMENT THE LINE GROUP BIT MAP.
+ JSR PC,FRPSUP ; DO SET UP AGAIN.
+ EXECUTE TEST AGAIN.
+
+ SUBORDINATE ROUTINES CALLED: TXRINI.
+-----*****

```

```

FRPSUP:: SAVE ;SAVE THE CONTENTS OF THE GPR'S.
;CALL REGISTER SAVE SUBRT.
MOV R0,70$ ;SAVE LPR PARAMETER FOR LINE TX.
MOV R1,72$ ;SAVE LPR PARAMETER FOR LINE RX.
+
+ SET UP THE TRANSMISSION/RECEPTION CONTROL BLOCK TO INITIALISE THE
+ ACTIVE LINES IN THE BIT MAP PASSED INTO THIS ROUTINE.
+
MOV R0,CBB ;SET CONTENTS OF LPR PARAMS IN TX/RX C.BLK.
MOV #CBB+2,R0 ;GET ADDRESS OF THE NEXT WORD IN THE CNTRL BLK.
MOV #4,(R0)+ ;LNCTRL PARAMETER, ENABLE RECEIVERS.
MOV R2,(R0)+ ;START ADDRESS OF DATA PATTERN.
MOV R3,(R0)+ ;SET DATA PATTERN LENGTH.
MOV #1,(R0)+ ;NUMBER OF DATA PATTERNS TO TRANSMIT.
MOV ACTLNS,(R0) ;BIT MAP OF LINES TO INITIALISE.
COM R4 ;GENERATE A BIT MAP OF ACTIVE LINES IN GRP1.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 100
GLOBAL SUBROUTINE

- FRPSUP -

```

4211 016164 040420          BIC   R4,(R0)+      ;CLEAR THE UNWANTED LINES.
4212 016166 113720 002176  MOVB  LOPBCK,(R0)+  ;SET LOOPBACK MODE,STAGGARED.
4213 016172 005200          INC   R0            ;INCREMENT ADDRESS TO GET NEXT WORD IN TABLE.
4214 016174 012710 000001  MOV   #1,(R0)      ;SET AMMOUNT OF OFFSET FOR EACH TX START.
4215
4216
4217      ;+
4218      ;: INITIALISE THE DUT AND THE ASSOCIATED POINTERS AND COUNTERS, TO THE STATE
4219      ;: DICTATED BY THE CONTENTS OF THE TX/RX CONTROL BLOCK.
4220 016200 004737 023666      ;-
4221      JSR   PC,TXRINI      ;INITIALISE DUT.
4222      ;+
4223      ;: SET UP CONTROL BLOCK FOR LINES IN GROUP 2.
4224 016204 012700 003030      ;-
4225      MOV   #CBB,R0      ;GET START ADDRESS OF CONTROL BLOCK.
4226 016210 010120          MOV   R1,(R0)+      ;SET LPR PARAMETER FOR RX LINES.
4227 016212 062700 000010      ADD   #10,R0        ;SELECT THE ADDRESS OF THE LINE BIT MAP IN C.B.
4228 016216 013710 002174      MOV   ACTLNS,(R0)   ;BIT MAP OF LINES TO INITIALISE.
4229 016222 005104          COM   R4            ;GENERATE A BIT MAP OF LINES IN GRP 2.
4230 016224 040410          BIC   R4,(R0)      ;CLEAR THE UNWANTED LINES.
4231
4232      ;+
4233      ;: INITIALISE THE DUT AND THE ASSOCIATED POINTERS AND COUNTERS, TO THE STATE
4234      ;: DICTATED BY THE CONTENTS OF THE TX/RX CONTROL BLOCK.
4235 016226 004737 023666      ;-
4236      JSR   PC,TXRINI      ;INITIALISE DUT.
4237
4238      ;+
4239      ;: SET-UP THE REQUIRED LPR PARAMETERS NEEDED FOR THE CORRECT RECEPTION OF DATA
4240      ;: ON ASSOCIATED IN-ACTIVE LINES.
4241
4242      ;+
4243      ;: INITIALISE LPR PARAMETERS FOR INACTIVE LINES IN GROUP 2.
4244 016232 012701 000377      ;-
4245 016236 013702 002174      MOV   #MAPLNS,R1    ;SET UP BIT MAP CORRESPONDING TO ALL LINES.
4246 016242 005101          MOV   ACTLNS,R2    ;GET THE ACTIVE (TX) LINE BIT MAP.
4247 016244 005102          COM   R1            ;GENERATE A BIT MAP OF NONE EXISTANT LINES.
4248 016246 040102          COM   R2            ;GENERATE A BIT MAP OF INACTIVE LINES.
4249 016250 040402          BIC   R1,R2        ;CLEAR ANY 'NONE EXISTANT' INACTIVE LINES.
4250 016252 010237 003042      BIC   R4,R2
4251 016256 005037 003040      MOV   R2,CBMAPA    ;SET UP BIT MAP IN CONTROL BLOCK.
4252 016262 013737 016360 003030  CLR   CBDPNA        ;CLEAR REPEAT TX COUNT IN CONTROL BLOCK.
4253 016270 004737 023666      MOV   72$,CBLPRA   ;SET-UP COMPLEMENTARY LPR PARAM.
4254      JSR   PC,TXRINI      ;INITIALISE INACTIVE LINES.
4255      ;+
4256      ;: INITIALISE LPR PARAMETERS FOR INACTIVE LINES IN GROUP 1.
4257 016274 013702 002174      ;-
4258 016300 005102          MOV   ACTLNS,R2    ;GET THE ACTIVE (TX) LINE BIT MAP.
4259 016302 040102          COM   R2            ;GENERATE A BIT MAP OF INACTIVE LINES.
4260 016304 005104          BIC   R1,R2        ;CLEAR ANY NONE EXISTANT INACTIVE LINES.
4261 016306 040402          COM   R4
4262 016310 010237 003042      BIC   R4,R2        ;ONLY PASS LGRP2 ASSOCIATED LINE BIT MAP.
4263 016314 013737 016356 003030  MOV   R2,CBMAPA    ;SET-UP BIT MAP IN CONTROL BLOCK.
4264 016322 004737 023666      MOV   70$,CBLPRA   ;SET-UP COMPLEMENTARY LPR PARAM FOR LGRP1.
4265      JSR   PC,TXRINI      ;INITIALISE INACTIVE LINES IN LGRP1.
4266      ;+
4267      ;: DISABLE RECEIVERS ON ALL LINES TO ENSURE THAT ONLY THE RECEIVERS OF THE
4268      ;: ASSOCIATED ACTIVE (TX) LINES ARE ENABLED.(STAGGARED LOOPBACK)

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 101
GLOBAL SUBROUTINE - FRPSUP -

```

4267      ; RE-ENABLE RECEPTION ON THE CORRECT ASSOCIATED LINES.
4268      :-
4269      MOV      #MAPLNS,R5      ;SET-UP BIT MAP FOR ALL LINES.
4270      JSR      PC,RXDSBL      ;DISABLE RX ON ALL LINES.
4271      ;+
4272      ; ENABLE RECEIVERS ON ASSOCIATED (RX) LINES.
4273      :-
4274      MOV      ACTLNS,R5      ;GET ACTIVE (TX) LINE BIT MAP.
4275      JSR      PC,CONMAP      ;GENERATE AN ASSOCIATED (RX) LINE BIT MAP.
4276      JSR      PC,RXENBL      ;ENABLE RECEIVERS ON ASSOCIATED LINES.
4277
4278      60$:      PASS          ;RESTORE GRP'S.
4279      JSR      PC,@(SP)+      ;RETURN TO PREG05 SUBRT.
4280
4281      RTS      PC
4282      70$:      .WORD      0      ;LOCAL STORAGE OF LPR PARAMETER TX.
4283      72$:      .WORD      0      ;LOCAL STORAGE OF LPR PARAMETER RX.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 102

GLOBAL SUBROUTINE - GETCHR -

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

```

.SBTTL GLOBAL SUBROUTINE - GETCHR -
+*****
* - GET A CHARACTER FROM THE RX BUFFER ROUTINE -
* THIS SUBROUTINE GETS A CHARACTER FROM THE RX BUFFER WHICH IS IN THE
* HOST SYSTEM MEMORY. IF THE BUFFER IS EMPTY UPON ENTRY OF THIS ROUTINE
* THIS ROUTINE RETURNS A NULL CHARACTER WITH DATA.VALID CLEAR AND A
* BUFFER EMPTY INDICATION.
*
* INPUTS: RXBCNT - RX BUFFER CHARACTER COUNT.
* RXBEND - LABEL AFTER END OF THE RX BUFFER AREA IN MEMORY.
* RXBETX - EQUATED TO RX BUFFER LEVEL AT WHICH TO ENABLE TX.
* RXBOPT - POINTER TO NEXT AVAILABLE INPUT SLOT OF RX BUFFER.
* RXBSTA - LABEL AT START OF RX BUFFER AREA IN MEMORY.
*
* OUTPUTS: R2 - CHARACTER WHICH IS READ FROM THE BUFFER.
* RXBOPT - UPDATED TO POINT TO NEXT INPUT SLOT OF RX BUFFER.
* RXBCNT - RX BUFFER CHARACTER COUNT (UPDATED).
* CARRY - "SUCCESS" FLAG (SET IF BUFFER IS NOT EMPTY ON ENTRY).
*
* CALLING SEQUENCE: JSR PC,GETCHR
*
* COMMENTS:
*
* SUBORDINATE ROUTINES CALLED: NONE.
*--*****

```

```

GETCHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                CLR R0 ;CLEAR THE 'RE-ENABLE' TX FLAG (SUBRTN OUTPUT).
                CLR R2 ;GET NULL CHAR IN CASE BUFFER IS EMPTY.
                TST RXBCNT ;CHECK FOR RX BUFFER EMPTY, CLEAR CARRY.
                BEQ 60$ ;EXIT THE ROUTINE IF BUFFER IS EMPTY.
                MOV RXBOPT,R4 ;GET THE BUFFER OUTPUT POINTER.
                MOV (R4),R2 ;GET A CHARACTER FROM THE BUFFER.
                CLR (R4)+ ;DELETE THE READ CHARACTER FROM THE BUFFER.
                CMP R4,#RXBEND ;CHECK IF POINTER SHOULD WRAP AROUND.
                BLO 2$ ;SKIP WRAPAROUND IF POINTER IS NOT AT END.
                MOV #RXBSTA,R4 ;WRAP INPUT POINTER AROUND.
                MOV R4,RXBOPT ;UPDATE THE OUTPUT POINTER STORAGE.
                DEC RXBCNT ;REMOVE THIS CHAR FROM THE BUFFER COUNT.
                SEC ;SET SUCCESS FLAG, BUFFER WAS NOT EMPTY.
                60$: PASS R2 ;RESTORE GPRS, EXCEPT
                MOV R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
                JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
                ;R2 - CONTAINS THE CHARACTER READ FROM BUFFER.
                ;CARRY-"SUCCESS" FLAG, SET IF BUFFER NOT EMPTY.
                RTS PC

```

```

016362
016362 004537 005232
016366 005000
016370 005002
016372 005737 002624
016376 001416
016400 013704 002620
016404 011402
016406 005024
016410 020427 003026
016414 103402
016416 012704 002626
016422 010437 002620
016426 005337 002624
016432 000261
016434
016434 010266 000006
016440 004736
016442 000207

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 103
GLOBAL SUBROUTINE

- GETLP1 -

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

016444
016444 004537 005232
016450 005701
016452 001010
016454 012701 004750
016460 012702 004760
016464 012703 004766
016470 012704 004772
016474 020427 005000
016500 103425
016502 012704 004772
016506 005723
016510 020327 004772
016514 103417
016516 012703 004766
016522 005722
016524 020227 004766
016530 103411
016532 012702 004760
016536 005721
016540 020127 004760
016544 103403
016546 005000
016550 000241
016552 000405

```
.SBTTL GLOBAL SUBROUTINE - GETLP1 -
** *****
* - GET LINE PARAMETERS ROUTINE NUMBER ONE -
* THIS ROUTINE IS USED TO REPEATEDLY GET COMBINATIONS OF LINE PARAMETER
* CONTENTS FOR THE SINGLE CHARACTER MODE TX/RX TEST (SHORT DATA PATTERN).
* EACH TIME THIS ROUTINE IS CALLED IT GETS ANOTHER COMBINATION OF THE
* PARAMTERS IN THE PARAMTER TABLES UNTIL ALL COMBINATIONS HAVE BEEN
* RETURNED AT WHICH POINT IT RETURNS A 'FAILURE' INDICATION.
*
* INPUTS: SINGLE CHARACTER MODE, SHORT DATA PATTERN TX/RX TABLES:
* SCBCT - NUMBER OF BITS PER CHAR TABLE (4 ENTRIES).
* SCBRT - BAUDRATES TABLE (3 ENTRIES).
* SCNST - NUMBER OF STOP BITS BITS TABLE (2 ENTRIES).
* SCTPT - TYPE OF PARITY TABLE (3 ENTRIES).
* EACH TABLE HAS A BASE AND END LABEL CONSISTING OF THE NAME OF
* THE TABLE WITH A 'B' AND 'E' APPENDED RESPECTIVELY.
* R1 THRU R4 - POINTERS INTO SCBCT THRU SCTPT TABLES RESPECTIVLY.
* R1 IS CLEAR IF THIS IS THE FIRST CALL OF GETLP1.
*
* OUTPUTS: R0 - COMPOSED LPR CONTENTS, CLEAR IF FAILURE (DONE).
* R1 THRU R4 - TABLE POINTERS (UPDATED).
*
* CALLING SEQUENCE: JSR PC,GETLP1
*
* COMMENTS: THIS ROUTINE SHOULD BE USED IN CONJUNCTION WITH A SWAPX
* ROUTINE TO AVOID DESTROYING THE GPR CONTENTS.
*
* SUBORDINATE ROUTINES CALLED: NONE.
*-- *****
```

```
GETLP1:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                TST R1 ;TEST FOR THIS BEING FIRST CALL OF GETLP1.
                BNE 2$ ;SKIP ORIGINAL SET UP IF NOT FIRST CALL.
                MOV #SCBCTB,R1 ;INITIALIZE BITS PER CHAR TABLE POINTER.
                MOV #SCBRTB,R2 ;INITIALIZE BAUDRATE TABLE POINTER.
                MOV #SCNSTB,R3 ;INITIALIZE # OF STOP BITS TABLE POINTER.
                MOV #SCTPTB,R4 ;INITIALIZE TYPE OF PARITY TABLE POINTER.
2$: CMP R4,#SCTPTE ;CHECK FOR POINTER AT END OF TABLE.
    BLO 4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
    MOV #SCTPTB,R4 ;RESET POINTER TO BEGINNING OF TABLE.
    TST (R3)+ ;INC THE # OF STOP BITS TABLE POINTER BY 2.
    CMP R3,#SCNSTE ;CHECK FOR POINTER AT END OF TABLE.
    BLO 4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
    MOV #SCNSTB,R3 ;RESET POINTER TO BEGINNING OF TABLE.
    TST (R2)+ ;INC BAUD RATES TABLE POINTER BY 2.
    CMP R2,#SCBRTE ;CHECK FOR POINTER AT END OF TABLE.
    BLO 4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
    MOV #SCBRTB,R2 ;RESET POINTER TO BEGINNING OF TABLE.
    TST (R1)+ ;INC THE BITS PER CHAR TABLE POINTER BY 2.
    CMP R1,#SCBCTE ;CHECK FOR POINTER AT END OF TABLE.
    BLO 4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
    CLR R0 ;PREPARE TO PASS OUT CLEAR LPR FIELDS.
    CLC ;INDICATE 'FAILURE' FOR EXIT.
    BR 60$ ;EXIT WITH 'FAILURE', WE'RE DONE.
```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 104
GLOBAL SUBROUTINE - GETLP1 -

```

4389
4390 016554 011100      4$:   MOV   (R1),R0      ;GET THE BITS/CHAR FIELD OF NEW LPR CONTENTS.
4391 016556 051200      BIS   (R2),R0      ;INCLUDE THE BAUD RATE FIELDS.
4392 016560 051300      BIS   (R3),R0      ;INCLUDE THE NUMBER OF STOP BITS FIELD.
4393 016562 052400      BIS   (R4)+,R0     ;INCLUDE THE TYPE OF PARITY FIELD.
4394
4395 016564 000261      SEC                               ;INDICATE 'SUCCESS' FOR EXIT.
4396
4397 016566      60$:   PASS   R0,R1,R2,R3,R4 ;RESTORE GPR R5, LEAVE THE FOLLOWING INTACT:
4398 016566 010066 000002      MOV   R0,R0SLOT(SP) ;PUT R0 IN STACK SLOT.
4399 016572 010166 000004      MOV   R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
4400 016576 010266 000006      MOV   R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
4401 016602 010366 000010      MOV   R3,R3SLOT(SP) ;PUT R3 IN STACK SLOT.
4402 016606 010466 000012      MOV   R4,R4SLOT(SP) ;PUT R4 IN STACK SLOT.
4403 016612 004736      JSR   PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
4404 016614 000207      RTS   PC          ; R1 THRU R4 - POINTERS, R0 - NEW LPR FIELDS.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 105
GLOBAL SUBROUTINE - GETLP2 -

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 016616
4436 016616 004537 005232
4437 016622 005701
4438 016624 001006
4439 016626 012701 004750
4440 016632 012702 004766
4441 016636 012703 004772
4442
4443 016642 020327 005000
4444 016646 103417
4445 016650 012703 004772
4446 016654 005722
4447 016656 020227 004772
4448 016662 103411
4449 016664 012702 004766
4450 016670 005721
4451 016672 020127 004760
4452 016676 103403
4453 016700 005000
4454 016702 000241
4455 016704 000406
4456
4457 016706 012700 177400
4458 016712 051100
4459 016714 051200
4460 016716 052300

```

.SBTTL GLOBAL SUBROUTINE - GETLP2 -
;-- *****
;-- * - GET LINE PARAMETERS ROUTINE NUMBER TWO -
;-- * THIS ROUTINE IS USED TO REPEATEDLY GET COMBINATIONS OF LINE PARAMETER
;-- * CONTENTS FROM THE SINGLE CHARACTER MODE TX/RX TEST (LONG DATA PATTERN).
;-- * EACH TIME THIS ROUTINE IS CALLED IT GETS ANOTHER COMBINATION OF THE
;-- * PARAMETERS IN THE PARAMETER TABLES UNTIL ALL COMBINATIONS HAVE BEEN
;-- * RETURNED AT WHICH POINT IT RETURNS A 'FAILURE' INDICATION.
;-- *
;-- * INPUTS: SINGLE CHARACTER MODE, SHORT DATA PATTERN TX/RX TABLES:
;-- *          SCBCT - NUMBER OF BITS PER CHAR TABLE (4 ENTRIES).
;-- *          SCNST - NUMBER OF STOP BITS TABLE (2 ENTRIES).
;-- *          SCTPT - TYPE OF PARITY TABLE (3 ENTRIES).
;-- *          EACH TABLE HAS A BASE AND END LABEL CONSISTING OF THE NAME OF
;-- *          THE TABLE WITH A 'B' AND 'E' APPENDED RESPECTIVELY.
;-- *          R1 THRU R3 - POINTERS INTO SCBCT, SCNST, SCTPT TABLES
;-- *          R1 IS CLEAR IF THIS IS THE FIRST CALL OF GETLP2.
;-- *
;-- * OUTPUTS: R0 - COMPOSED LPR CONTENTS, CLEAR IF FAILURE (DONE),
;-- *          38.4K BAUDRATE IS SELECTED.
;-- *          R1 THRU R3 - TABLE POINTERS (UPDATED).
;-- *
;-- * CALLING SEQUENCE: JSR PC,GETLP2
;-- *
;-- * COMMENTS: THIS ROUTINE SHOULD BE USED IN CONJUNCTION WITH A SWAPX
;-- *            ROUTINE TO AVOID DESTROYING THE GPR CONTENTS.
;-- *
;-- * SUBORDINATE ROUTINES CALLED: NONE.
;-- *****
GETLP2:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR  R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          TST  R1 ;TEST FOR THIS BEING FIRST CALL OF GETLP2.
          BNE  2$ ;SKIP ORIGINAL SET UP IF NOT FIRST CALL.
          MOV  #SCBCTB,R1 ;INITIALIZE BITS PER CHAR TABLE POINTER.
          MOV  #SCNSTB,R2 ;INITIALIZE # OF STOP BITS TABLE POINTER.
          MOV  #SCTPTB,R3 ;INITIALIZE TYPE OF PARITY TABLE POINTER.
2$:      CMP  R3,#SCTPTE ;CHECK FOR POINTER AT END OF TABLE.
          BLO  4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
          MOV  #SCTPTB,R3 ;RESET POINTER TO BEGINNING OF TABLE.
          TST  (R2)+ ;INC THE # OF STOP BITS TABLE POINTER BY 2.
          CMP  R2,#SCNSTE ;CHECK FOR POINTER AT END OF TABLE.
          BLO  4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
          MOV  #SCNSTB,R2 ;RESET POINTER TO BEGINNING OF TABLE.
          TST  (R1)+ ;INC BAUD RATES TABLE POINTER BY 2.
          CMP  R1,#SCBCTE ;CHECK FOR POINTER AT END OF TABLE.
          BLO  4$ ;GO GET LPR CONTENTS IF NOT AT END OF TABLE.
          CLR  R0 ;PREPARE TO PASS OUT CLEAR LPR FIELDS.
          CLC  ;INDICATE 'FAILURE' FOR EXIT.
          BR   60$ ;EXIT WITH 'FAILURE', WE'RE DONE.
4$:      MOV  #177400,R0 ;SET BAUD RATE FIELDS FOR 38.4 K BAUD.
          BIS  (R1),R0 ;GET THE BITS/CHAR FIELD OF NEW LPR CONTENTS.
          BIS  (R2),R0 ;INCLUDE THE NUMBER OF STOP BITS FIELD.
          BIS  (R3)+,R0 ;INCLUDE THE TYPE OF PARITY FIELD.

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 106
GLOBAL SUBROUTINE

- GETLP2 -

4461									
4462	016720	000261			SEC				:INDICATE 'SUCCESS' FOR EXIT.
4463									
4464	016722			60\$:	PASS	R0,R1,R2,R3			:RESTORE GPRS R4 & R5, LEAVE FOLLOWING INTACT:
4465	016722	010066	000002			MOV	R0,R0SLOT(SP)		:PUT R0 IN STACK SLOT.
4466	016726	010166	000004			MOV	R1,R1SLOT(SP)		:PUT R1 IN STACK SLOT.
4467	016732	010266	000005			MOV	R2,R2SLOT(SP)		:PUT R2 IN STACK SLOT.
4468	016736	010366	000010			MOV	R3,R3SLOT(SP)		:PUT R3 IN STACK SLOT.
4469	016742	004736				JSR	PC,@(SP)+		:RETURN TO PREG05 SUBRT.
4470	016744	000207			RTS	PC			: R1 THRU R3 - POINTERS, R0 - NEW LPR FIELDS.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 107
GLOBAL SUBROUTINE - GETTIM -

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
4501
4502
4503
4504
4505
4506
4507
4508
4509

016746
016746 004537 005232
016752 000301
016754 042701 177400
016760 010102
016762 042701 000360
016766 006202
016770 006202
016772 006202
016774 006202
016776 020102
017000 101401
017002 010201
017004 116102 005122
017010 042702 177400
017014 010237 002242
017020
017020 004736
017022 000207

```

.SBTTL GLOBAL SUBROUTINE - GETTIM -
:++ *****
: * - GET TIME-OUT VALUE BASED ON MINIMUM BAUDRATE ROUTINE -
: * THIS SUBROUTINE GETS THE NECESSARY TIME-OUT VALUE TO VERIFY THAT ALL
: * CHARACTERS HAVE BEEN RECEIVED AT THE COMPLETION OF THE TX/RX OF A DATA
: * PATTERN. THIS USES THE SLOWEST BAUDRATE WHICH IS SPECIFIED IN THE
: * PASSED IN DUT LPR CONTENTS TO CALCULATE THIS TIME-OUT VALUE.
: *
: * INPUTS: R1 - DUT LPR CONTENTS.
: *
: * OUTPUTS: RXTOUT - TIME-OUT VALUE FOR WAITING FOR LAST RX CHAR.
: *
: * CALLING SEQUENCE: JSR PC,GETTIM
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****

GETTIM:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                SWAB R1 ;PUT THE BAUD RATE FIELDS IN THE LOW BYTE.
                BIC #177400,R1 ;CLEAR STOP,PARITY,AND CHAR FIELDS.
                MOV R1,R2 ;COPY BAUD RATE FIELDS.
                BIC #360,R1 ;SELECT RX BAUD RATE FIELD ONLY.
                ASR R2 ;SHIFT TX BAUD RATE FIELD
                ASR R2 ; TO OCCUPY THE LOW FOUR BYTES.
                ASR R2
                ASR R2
                CMP R1,R2 ;CHECK IF SAME BAUD RATE IN EACH FIELD.
                BLOS 2$ ;BRANCH IF RX BAUD RATE IS LOWER OR SAME.
                MOV R2,R1 ;RX BAUD RATE IS THE SLOWER OF THE TWO.
                MOVB PROTBL(R1),R2 ;GET PROPORTIONAL DELAY FROM TABLE.
                BIC #177400,R2 ;CLEAR UPPER BYTE BECAUSE OF SIGN EXTENSION.
                MOV R2,RXTOUT ;LOAD THE RX TIME-OUT VARIABLE.

60$: PASS ;RESTORE GPRS.
                JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.

                RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 108
GLOBAL SUBROUTINE - INDATP -

```

4510 .SBTTL GLOBAL SUBROUTINE - INDATP -
4511 :++ *****
4512 :* - INITIALISE DATA PATTERN -
4513 :* THIS SUBROUTINE IS USED TO INITIALISE AN INCREMENTAL BYTE DATA PATTERN
4514 :* IN THE GENERAL BUFFER AREA.
4515 :* THE DATA PATTERN WILL BE SEQUENTIAL FROM 0 TO 255 (DECIMAL).
4516 :*
4517 :* INPUTS: BUFBAS - ADDRESS OF THE START OF THE GENERAL BUFFER AREA.
4518 :*          BUFMID - ADDRESS OF THE 255 TH LOCATION.
4519 :*
4520 :* OUTPUTS: THE FIRST 255 LOCATIONS OF THE GENERAL BUFFER AREA CONTAIN DATA
4521 :*
4522 :* CALLING SEQUENCE: JSR PC,INIDATP
4523 :*
4524 :* COMMENTS:
4525 :*
4526 :* SUBORDINATE ROUTINES CALLED: NONE.
4527 :-- *****
4528
4529 017024 INDATP:: SAVE JSR ;SAVE CONTENTS OF GPRS R0 THRU R5.
4530 017024 004537 005232 R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4531
4532 017030 012702 003510 MOV #BUFBAS,R2 ;INITIALIZE THE DATA PATTERN IN THE GENERAL
4533 017034 005003 CLR R3 ; DATA BUFFER TO A 256 BYTE PATTERN.
4534 017036 110322 2$: MOVB R3,(R2)+ ;
4535 017040 005203 INC R3 ;SELECT THE NEXT CHARACTER.
4536 017042 020227 004110 CMP R2,#BUFMID ;CHECK IF WE HAVE 256 DATA PATTERNS.
4537 017046 103773 BLO 2$ ;
4538
4539 017050 60$: PASS ;RESTORE GPRS.
4540 017050 004736 JSR PC,@(SP)+ ;RETURN TO PR:G05 SUBRT.
4541 017052 000207 RTS PL

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 109
GLOBAL SUBROUTINE - INICHR -

4542
4543
4544
4545
4546
4547
4548
4549
4550
4551
4552
4553
4554
4555
4556
4557
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
4584
4585
4586
4587
4588
4589
4590
4591
4592
4593
4594
4595
4596
4597

```

.SBTTL GLOBAL SUBROUTINE - INICHR -
** *****
* - SEND INITIAL CHARACTERS ROUTINE -
* THIS ROUTINE IS USED TO INITIATE SINGLE CHARACTER TRANSMISSION.
* THIS ROUTINE SENDS THE INITIAL CHARACTERS TO EACH ACTIVE LINES TO
* CAUSE FUTURE TX INTERRUPTS WHICH WILL CONTINUE THE TRANSMISSION IF
* MORE THAN ONE CHARACTER IS TO BE SENT TO EACH ACTIVE LINE.
*
* INPUTS: ACTLNS - BIT MAP OF ACTIVE DUT LINES.
* BITTBL - LABEL OF TABLE OF WORDS EACH WITH A BIT SET.
* CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
* DPENDB - BASE OF THE DATA PATTERN END TABLE (ENTRY PER LINE).
* DPLENB - BASE OF THE DATA PATTERN LENGTH TABLE.
* IBM - BIT MASK OF INACTIVE TX/RX BITS.
* IESTAT - STATES OF DUT INT ENABLE BITS (OTHER BITS CLEAR).
* NUMLNS - EQUATED TO THE NUMBER OF LINES ON THE DUT.
* TXCHRA - CONTAINS THE ADDRESS OF THE DUT TXCHAR REGISTER.
* TXCNTB - LABEL AT BASE OF THE TX CHARACTER COUNTER TABLE.
* TXPTRB - LABEL AT BASE OF THE TX DATA PATTERN POINTERS TABLE.
*
* OUTPUTS: CSR - DUT CSR IND.ADR.REG FIELD IS DESTROYED.
* TXCHAR - DUT TXCHAR HAS WORD WRITTEN TO IT.
* TXCNTX - COUNTERS INCREMENTED FOR LINES ON WHICH CHARS SENT.
* TXPTRB - EACH POINTER IN TABLE POINTS TO NEXT TX CHAR FOR LINE.
*
* CALLING SEQUENCE: JSR PC,INICHR
*
* COMMENTS: THIS ROUTINE ASSUMES THAT AT LEAST ONE CHARACTER SHOULD BE
* TRANSMITTED ON EACH ACTIVE LINE.
* INTERRUPTS MUST BE DISABLED WHEN CALLING THIS ROUTINE.
*
* SUBORDINATE ROUTINES CALLED: NONE.
** *****
INICHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
; R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV IESTAT,R1 ;GET STATE OF TX.IE, RX.IE FOR USE IN SETTING
; UP THE IND.ADR.REG FIELD OF THE DUT CSR.
CLR R2 ;SET LINE NUMBER OFFSET TO LINE 0.
BIT BITTBL(R2),ACTLNS ;TEST THE ACTIVE LINES BIT FOR THIS LINE.
BEQ 6$ ;DON'T TX ON THIS LINE IF IT IS NOT ACTIVE.
MOV R1,@CSRA ;SET UP THE IND.ADR.REG FIELD OF THE CSR.
MOV TXPTRB(R2),R5 ;GET THE TX DATA PATTERN POINTER FOR THIS LINE.
MOVB (R5)+,R4 ;GET THE CHAR TO TX ON THIS LINE, INC POINTER.
CMP R5,DPENDB(R2) ;COMPARE POINTER WITH DATA PATTERN END ADR.
BLO 4$ ;SKIP POINTER WRAPAROUND IF NOT AT PATTERN END.
SUB DPLENR(R2),R5 ;WRAP TX POINTER AROUND TO BEGINNING OF PAT'N.
MOV R5,TXPTRB(R2) ;UPDATE THE TX POINTER STORAGE TABLE FOR LINE.
BIC IBM,R4 ;CLEAR INACTIVE BITS OF TX CHARACTER WORD.
BIS #BIT15,R4 ;SET THE TX.DATA.VALID BIT IN THE WORD.
MOV R4,@TXCHA ;TX THE FIRST CHARACTER FOR THIS LINE.
INC TXCNTB(R2) ;INCREMENT TX CHARACTER COUNTER FOR THIS LINE.
INC R1 ;INCREMENT WORD FOR IND.ADR.REG FIELD SET UP.
ADD #2,R2 ;SET LINE NUMBER OFFSET TO NEXT LINE.
CMP R2,#NUMLNS*2 ;COMPARE LINE OFFSET WITH TWICE THE # OF LINES.
BLT 2$ ;LOOP TO SEND CHAR TO ANOTHER LINE IF NOT DONE.

```

017054			
017054	004537	005232	
017060	013701	002234	
017064	005002		
017066	036237	002332	002174 2\$:
017074	001424		
017076	010177	163100	
017102	016205	003250	
017106	112504		
017110	020562	003050	
017114	103402		
017116	166205	003110	
017122	010562	003250	4\$:
017126	043704	002226	
017132	052704	100000	
017136	010477	163042	
017142	005262	003410	
017146	005201		6\$:
017150	062702	000002	
017154	020227	000020	
017160	002742		

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 110
GLOBAL SUBROUTINE

- INICHR -

4598 017162
4599 017162 004736
4600 017164 000207

608: PASS
RTS PC

JSR ;RESTORE GPRS.
PC,@(SP)+

;RETURN TO PREG05 SUBRT.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 112
GLOBAL SUBROUTINE - INIDMA -

```

4657
4658 017224 004737 015662      :- JSR    PC,DODMA
4659 017230 103403              BCS     6$      ;SKIP ERROR IF DODMA WAS SUCCESSFUL.
4660
4661      :-+ SET THE PROPER BIT OF THE TX INTERRUPT FLAGS TO INDICATE THE LINE ERROR.
4662      :-
4663 017232 050537 002252      BIS     R5,TXINTF ;INDICATE THE ERROR.
4664 017236 000402              BR      10$     ;SKIP UPDATING POINTERS AND COUNTERS.
4665
4666      :-+ UPDATE THE TX CHARACTER COUNT FOR THIS LINE.
4667      :-
4668 017240 060364 003410      6$:    ADD     R3,TXCNTB(R4) ;ADD THE DATA PATTERN LENGTH TO TX CHAR COUNT.
4669      :-+
4670      :- INCREMENT LINE COUNTER,GOTO NEXT LINE IF NOT DONE.
4671      :-
4672 017244 005201              10$:   INC     R1      ;INCREMENT THE LINE COUNTER.
4673 017246 020127 000010      CMP     R1,#NUMLNS ;COMPARE THE LINE COUNTER WITH NUMBER OF LINES.
4674 017252 002752              BLT     2$      ;LOOP TO SEND CHAR TO ANOTHER LINE IF NOT DONE.
4675
4676 017254              60$:   PASS
4677 017254 004736              JSR     PC      ;RESTORE GPRS.
4678 017256 000207              RTS     PC      P,@(SP)+ ;RETURN TO PREG05 SUBRT.
    
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 113
GLOBAL SUBROUTINE

- LINBIT -

4679
4680
4681
4682
4683
4684
4685
4686
4687
4688
4689
4690
4691
4692
4693
4694
4695
4696
4697
4698
4699
4700
4701
4702
4703
4704
4705
4706
4707
4708
4709
4710
4711

017260
017260 004537 005232
017264 042701 177760
017270 006301
017272 016100 002332
017276
017276 010066 000002
017302 004736
017304 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.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
;MASK OUT ALL BUT 4 LSBITS OF THE LINE #.
;MULTIPLY LINE # BY 2 TO GET WORD TABLE OFFSET.
;GET THE SINGLE BIT BIT MAP.
;RESTORE GPRS, EXCEPT THE FOLLOWING.
R0,ROSLOT(SP) ;PUT R0 IN STACK SLOT.
PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
;R0 - BIT MAP WITH LINE # BIT SET.
```

```
BIC #177760,R1
ASL R1
MOV BITTBL(R1),R0
60$: PASS R0
RTS PC
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 114
GLOBAL SUBROUTINE - MAPCNT -

```

4712 .SBTTL GLOBAL SUBROUTINE - MAPCNT -
4713 :+*****
4714 :* - COUNT BITS IN BIT MAP ROUTINE -
4715 :* THIS SUBROUTINE COUNTS THE NUMBER OF BITS WHICH ARE SET IN A BIT MAP.
4716 :*
4717 :* INPUTS: R2 - THE BIT MAP FOR WHICH TO COUNT THE BITS.
4718 :*
4719 :* OUTPUTS: R2 - COUNT OF THE NUMBER OF BITS THAT WERE SET.
4720 :*
4721 :* CALLING SEQUENCE: JSR PC,MAPCNT
4722 :*
4723 :* COMMENTS:
4724 :*
4725 :* SUBORDINATE ROUTINES CALLED: NONE.
4726 :--*****
4727
4728 017306 MAPCNT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
4729 017306 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4730 017312 010201 MOV R2,R1
4731 017314 001405 BEQ 60$ ;EXIT WITH ZERO IF NO BITS ARE SET IN MAP.
4732
4733 017316 005002 CLR R2 ;CLEAR THE BIT COUNT.
4734 017320 000261 SEC ;COUNT THE LAST BIT TO BE SHIFTED OUT.
4735
4736 017322 005502 2$: ADC R2 ;COUNT THE BIT IF IT WAS SET.
4737 017324 006301 ASL R1 ;SHIFT ANOTHER BIT OUT OF THE MAP.
4738 017326 001375 BNE 2$ ;LOOP IF ALL BITS NOT SHIFTED OUT OF MAP.
4739
4740 017330 60$: PASS R2 ;RESTORE GPRS, EXCEPT THE FOLLOWING:
4741 017330 010266 000006 MOV R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
4742 017334 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
4743 017336 000207 RTS PC ; R2 - COUNT OF BITS SET IN BIT MAP.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 115
GLOBAL SUBROUTINE

- MSLGET -

4744
4745
4746
4747
4748
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758
4759
4760
4761
4762
4763
4764
4765
4766
4767
4768
4769
4770
4771
4772
4773
4774
4775
4776
4777
4778
4779
4780
4781
4782
4783
4784
4785
4786
4787
4788
4789
4790
4791
4792
4793
4794
4795
4796
4797
4798
4799

```

.SBTTL GLOBAL SUBROUTINE - MSLGET -
*****
- MILLI SECONDS LOOP WHICH RETURNS READ WORD AND REMAINING TIME -
THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE. IT IS USED
TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD. THE
CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
ROUTINE AND THEN ONCE EACH MILLI-SECOND THERE AFTER.
UPON RETURN, THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION
IS RETURNED BY THIS SUBROUTINE.

INPUTS:      R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
              R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
              R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
              R4 - ADDRESS OF THE WORD TO TEST.
              MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.

OUTPUTS:     R0 - THE LAST WORD WHICH WAS READ TO CHECK FOR THE CONDITION.
              R1 - REMAINING NUMBER OF MS IN TIME-OUT TIME.
              CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).

CALLING SEQUENCE:  JSR      PC,MSLGET

COMMENTS:     THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
              CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
              ON THE SYSTEM.
              THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
              DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
              LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
              IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
              THE DESIRED CONDITION BEFORE RETURNING. IT INDICATES SUCCESS
              IF THE CONDITION IS MET, FAILURE OTHERWISE.

SUBORDINATE ROUTINES CALLED: NONE.
*****
MSLGET:: SAVE                ;SAVE CONTENTS OF GPRS R0 THRU R5.
                        JSR      R5,PREG05      ;CALL REGISTER SAVE SUBRT.
;+
; SET UP MASK FOR REMOVING UNUSED BITS IN THE TEST WORD, AND CLEAR UNUSED
; BITS IN THE DESIRED STATE WORD TO ALLOW DIRECT COMPARISON.
;-
      COM      R2                ;GET MASK OF UNUSED BITS.
      BIC      R2,R3             ;MASK OUT UNUSED BITS IN DESIRED STATE WORD.
;+
; HANDLE THE TEST AND EXIT IF WE HAVE A 0 TIME-OUT VALUE.
;-
      TST      R1                ;TEST THE TIME-OUT VALUE FOR ZERO.
      BNE     2$                ;IF NON-ZERO TIME-OUT, GO LOOP AND TEST.
      MOV     (R4),R0           ;GET THE WORD TO TEST BEFORE EXITING.
      MOV     R0,62$           ;SAVE VALUE SO WE CAN RETURN IT.
      BIC     R2,R0            ;MASK OUT UNTESTED BITS OF WORD.
      CMP     R0,R3            ;COMPARE AGAINST DESIRED STATE WORD.
      SEC     ;INDICATE SUCCESS IN CASE WORDS ARE EQUAL.

```

017340
017340 004537 005232

017344 005102
017346 040203

017350 005701
017352 001011
017354 011400
017356 010037 017452
017362 040200
017364 020003
017366 000261

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 116
GLOBAL SUBROUTINE - MSLGET -

```

4800 017370 001420          BEQ      6$          ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.
4801 017372 000241          CLC              ;INDICATE FAILURE (TIME-OUT).
4802 017374 000416          BR       6$          ;EXIT WITH FAILURE, WORDS AREN'T EQUAL.
4803
4804          ;+
4805          ;: NON-ZERO TIME-OUT VALUE. LOOP, WAITING FOR CONDITION OR TIME-OUT.
4806 017376 013705 002302 2$:      MOV      MSLCNT,R5      ;LOAD MS LOOP COUNT.
4807 017402 011400          4$:      MOV      (R4),R0      ;GET THE WORD TO TEST.
4808 017404 010037 017452      MOV      R0,62$      ;SAVE WORD IN CASE THIS IS THE LAST.
4809 017410 040200          BIC      R2,R0      ;MASK OUT UNTESTED BITS OF WORD.
4810 017412 020003          CMP      R0,R3      ;COMPARE AGAINST DESIRED STATE WORD.
4811 017414 000261          SEC              ;SET CARRY IN CASE OF SUCCESS.
4812 017416 001405          BEQ      6$          ;EXIT WITH SUCCESS IF WORDS ARE EQUAL.
4813 017420 005305          DEC      R5      ;COUNT DOWN THE INSIDE MS LOOP COUNT.
4814 017422 001367          BNE      4$          ;LOOP IF MS NOT UP.
4815 017424 005301          DEC      R1      ;DECREMENT THE MS TIME COUNT.
4816 017426 001363          BNE      2$          ;IF TIME NOT UP, LOOP TO COUNT ANOTHER MS.
4817 017430 000241          CLC              ;CLEAR CARRY, WE TIMED-OUT.
4818
4819          ;+
4820          ;: HAVE EITHER FOUND CONDITION, OR TIMED-OUT (POSSIBLY FROM 0 TIME-OUT VALUE).
4821          ;: RESTORE THE LAST CONTENTS READ FROM THE TEST WORD. EXIT ROUTINE.
4822 017432 013700 017452 6$:      MOV      62$,R0      ;PASS OUT THE LAST READ WORD.
4823 017436          60$:     PASS      R0,R1      ;RESTORE GPRS, EXCEPT THE FOLLOWING:
4824 017436 010066 000002          MOV      R0,R0SLOT(SP) ;PUT R0 IN STACK SLOT.
4825 017442 010166 000004          MOV      R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
4826 017446 004736          JSR      PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
4827          ;R0 - LAST READ WORD CHECKED FOR CONDITION.
4828          ;R1 - REMAINING TIME (0 IF TIME-OUT OCCURED).
4829 017450 000207          RTS      PC          ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.
4830          ;+
4831          ;: LOCAL STORAGE.
4832          ;-
4833 017452 000000 62$:     .WORD  0      ;STORAGE FOR THE LAST READ WORD.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 117
GLOBAL SUBROUTINE - MSLOOP -

```

4834 .SBTTL GLOBAL SUBROUTINE - MSLOOP -
4835 *****
4836 * - TEST LOOP SUBROUTINE -
4837 * THIS SUBROUTINE IS A GENERAL PURPOSE TEST LOOP SUBROUTINE. IT IS USED
4838 * TO VERIFY THAT A CERTAIN ACTION OCCURS BEFORE A TIME-OUT PERIOD. THE
4839 * CALLING ROUTINE PASSES IN WHICH BITS SHOULD BE SET AND CLEARED FOR THE
4840 * DESIRED CONDITION AND THE TIME-OUT VALUE IN MILLI-SECONDS.
4841 * THIS ROUTINE CHECKS FOR THE DESIRED CONDITION UPON ENTRANCE INTO THE
4842 * ROUTINE AND THEN ONCE EACH MILLI-SECOND THEREAFTER.
4843 *
4844 * INPUTS: R1 - TIME-OUT VALUE IN MILLI-SECONDS (UP TO 64K MS).
4845 * R2 - BIT MAP OF BITS TO TEST (1 INDICATES TO TEST THE BIT).
4846 * R3 - DESIRED STATES OF THE INDICATED FIELDS IN R2.
4847 * R4 - ADDRESS OF THE WORD TO TEST.
4848 * MSLCNT - MILLI SECOND SOFTWARE LOOP COUNT.
4849 *
4850 * OUTPUTS: CARRY - SUCCESS FLAG (SET IF CONDITION IS MET BEFORE TIME-OUT).
4851 *
4852 * CALLING SEQUENCE: JSR PC,MSLOOP
4853 *
4854 * COMMENTS: THIS ROUTINE WORKS WITH OR WITHOUT A HARDWARE CLOCK, BUT THE
4855 * CALIBRATION IS ONLY GUARENTEED WHEN A LINE CLOCK IS AVAILABLE
4856 * ON THE SYSTEM.
4857 * THIS ROUTINE CAN BE USED AS A DELAY ROUTINE, BY SPECIFYING THE
4858 * DESIRED DELAY AS THE TIME-OUT AND SPECIFYING A CONDITION TO
4859 * LOOK FOR WHICH WILL NOT BE MET DURING THE DELAY.
4860 * IF A TIME-OUT VALUE OF 0 IS SPECIFIED, THIS ROUTINE CHECKS FOR
4861 * THE DESIRED CONDITION BEFORE RETURNING. IT INDICATES SUCCESS
4862 * IF THE CONDITION IS MET, FAILURE OTHERWISE.
4863 *
4864 * SUBORDINATE ROUTINES CALLED: MSLGET.
4865 *****
4866
4867 017454 MSLOOP:: SAVE ;SAVE CONIENTS OF GPRS R0 THRU R5.
4868 017454 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4869
4870
4871 *+
4872 * CALLING THE MSLGET ROUTINE FROM THE MSLOOP ROUTINE ISOLATES THE CALLER OF
4873 * MSLOOP FROM THE RETURNED TEST WORD AND REMAINING TIME-OUT VALUES.
4874
4875 JSR PC,MSLGET ;CALL THE MULTI-PURPOSE MS LOOP AND SEARCH RTN.
4876 017464 60$: PASS ;RESTORE GPRS,
4877 017464 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
4878 017466 000207 RTS PC ;CARRY - SET IF SUCCESS, CLEAR IF TIME-OUT.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 118
GLOBAL SUBROUTINE - MUL16U -

```

4879 .SBTTL GLOBAL SUBROUTINE - MUL16U -
4880 :-- *****
4881 :          - 16 BIT UNSIGNED MULTIPLY ROUTINE -
4882 :          THIS ROUTINE MULTIPLIES 2 16 BIT UNSIGNED NUMBERS AND RETURNS A 16 BIT
4883 :          UNSIGNED RESULT. THE MULTIPLICATION IS PERFORMED BY ITERATIVE
4884 :          ADDITION OF ONE NUMBER TO A SUM WHILE DECREMENTING THE OTHER NUMBER
4885 :          TO ZERO. IF OVERFLOW OCCURS (177777 TO 0) THE PRODUCT IS INVALID.
4886 :
4887 :          INPUTS:      R1 - MULTIPLICAND (16 BIT UNSIGNED).
4888 :                      R2 - MULTIPLIER (16 BIT UNSIGNED).
4889 :
4890 :          OUTPUTS:    R1 - PRODUCT (16 BIT UNSIGNED), -1 IF OVERFLOW.
4891 :                      CARRY - SET IF SUCCESS (NO OVERFLOW), CLEAR OTHERWISE.
4892 :
4893 :          CALLING SEQUENCE:  JSR    PC,MUL16U
4894 :
4895 :          COMMENTS:    NOTE: FOR MINIMUM EXECUTION TIME R2 SHOULD CONTAIN THE
4896 :                      SMALLER OF THE 2 ARGUMENTS.
4897 :
4898 :          SUBORDINATE ROUTINES CALLED: NONE.
4899 :-- *****
4900
4901 017470          MUL16U:: SAVE
4902 017470 004537 005232          JSR    ;SAVE CONTENTS OF GPRS R0 THRU R5.
4903 017474 005003          CLR    R3          ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
4904 017476 005702          TST    R2          ;CLEAR THE PRODUCT.
4905 017500 001003          BNE    2$          ;CHECK THE MULTIPLIER.
4906 017502 005001          CLR    R1          ;GO TO DO MULTIPLICATION IF NOT ZERO.
4907 017504 000261          SEC          ;RETURN A PRODUCT OF ZERO.
4908 017506 000412          BR     60$          ;INDICATE SUCCESS.
4909
4910 017510          2$: ADD    R1,R3          ;EXIT THE ROUTINE.
4911 017512 103405          BCS    50$          ;ADD THE MULTIPLICAND TO THE PRODUCT.
4912 017514 005302          DEC    R2          ;EXIT WITH OVERFLOW IF ONE OCCURRED.
4913 017516 001374          BNE    2$          ;DECREMENT THE MULTIPLIER.
4914 017520 010301          MOV    R3,R1          ;LOOP IF MULTIPLIER NOT ZERO.
4915 017522 000261          SEC          ;PREPARE TO PASS OUT THE PRODUCT.
4916 017524 000403          BR     60$          ;INDICATE SUCCESS.
4917
4918 017526 012701 177777          50$: MOV    #-1,R1          ;EXIT WITH SUCCESS.
4919 017532 000241          CLC          ;FORCE PRODUCT TO MAX VALUE, WE OVERFLOWED.
4920
4921 017534          60$: PASS    R1          ;INDICATE FAILURE.
4922 017534 010166 000004          MOV    R1,R1SLOT(SP) ;RESTORE GPRS, EXCEPT THE FOLLOWING:
4923 017540 004736          JSR    PC,@(SP)+ ;PUT R1 IN STACK SLOT.
4924
4925 017542 000207          RTS    PC          ;RETURN TO PREG05 SUBRT.
; R1 - PRODUCT (16 BIT UNSIGNED),
; CARRY - SET IF SUCCESS (NO OVERFLOW).

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 119

GLOBAL SUBROUTINE - NEWCHR -

4926
4927
4928
4929
4930
4931
4932
4933
4934
4935
4936
4937
4938
4939
4940
4941
4942
4943
4944
4945
4946
4947
4948
4949
4950
4951
4952
4953
4954
4955
4956
4957
4958
4959
4960
4961
4962
4963
4964
4965
4966
4967
4968
4969
4970
4971
4972
4973
4974
4975
4976
4977
4978
4979
4980
4981

```

.SBTTL GLOBAL SUBROUTINE - NEWCHR -
:++ *****
:  - NEW CHARACTER HANDLING ROUTINE -
:  THIS SUBROUTINE HANDLES A NEW CHARACTER WHICH HAS BEEN READ FROM
:  THE DUT. THE COUNTERS AND POINTERS WHICH ARE INVOLVED WITH THE
:  CHARACTER ARE UPDATED. THE CHARACTER IS CHECKED FOR ERRORS AND
:  ANY ERRORS WHICH ARE FOUND ARE REPORTED.
:
: INPUTS:  R2 - THE READ CHARACTER INCLUDING ERROR FLAGS AND LINE NUMBER.
:          R3 - MASK OF THE INACTIVES BITS IN A TX OR RX CHAR BYTE.
:          ACTLNS - BIT MAP OF ACTIVE DUT LINES.
:          DPRSQB - LABEL AT DATA PATTERN RESYNC QUEUES TABLE BASE.
:          TXRXLB - BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
:          BITTBL - TABLE OF WORDS WITH BITS SET FOR USE IN FORMING MAPS.
:          ERSMPF - 'PRINT ERROR SUMMARY FOR LINE' FLAGS.
:          ERRRTBL - ERROR INFORMATION (ERRNBR, ERRMSG, ERRTP).
:          ERCNTB - BASE OF THE RX CHARACTER ERROR COUNTERS TABLE.
:          NDERPT - CONTAINS NUMBER OF CHAR ERRORS TO REPORT ON A LINE.
:
: INPUTS TO SUBROUTINES:  CHCNTB, DPENDB, DPLEN, DPRSQE, EXCNTB, RXCNTB,
:                          RXPTRB, ERRNBR, ERRMSG, ERRTP.
:
: OUTPUTS:  ERRBLK - CONTENTS DESTROYED.
:           FOLLOWING VARIABLES UPDATED FOR LINE ON WHICH CHAR WAS RECEIVED:
:           DPRSQ - DATA PATTERN RESYNC QUE OF RECEIVED CHARACTERS.
:           ERCNT - COUNT OF THE NUMBER OF CHARACTER ERRORS ON LINE.
:           ERSMPF - UPDATED 'PRINT ERROR SUMMARY FOR LINE' FLAGS.
:           EXCNT - COUNT OF THE NUMBER OF EXTRA CHARS RECEIVED ON LINE.
:           RXCNT - COUNT OF THE NUMBER OF CHARACTERS RECEIVED ON LINE.
:           RXPTR - UPDATED TO POINT TO THE NEXT EXPECTED CHAR ON LINE.
:
: CALLING SEQUENCE:  JSR  PC,NEWCHR
:
: COMMENTS:  THIS ROUTINE CAN REPORT ERRORS WITH NUMBERS INITIAL ERRNBR
:            AND INITIAL ERRNBR + 1. ERRNBR IS RESTORED TO ITS INITIAL
:            VALUE BEFORE THIS ROUTINE RETURNS.
:
: SUBROUTINES CALLED:  CKCHR,CKINAC,TXROFF,TXRON.
: INDIRECT SUBROUTINES:  CHKEXT,CHKLOS,ER9002,ER9003,UPDCHR.
:-- *****
NEWCHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
; R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV R3,R5 ;GET THE BIT MAP OF INACTIVE DATA BYTE BITS.
BIS #177400,R5 ;ALL UPPER BITS OF EXPECTED DATA ARE INACTIVE.
CLR 70$ ;CLEAR THE 'ERROR FOUND' FLAG.
:
: IF THE NEW CHARACTER IS VALID ON AN INACTIVE LINE, GO REPORT ERROR.
: ROUTINE USED ALSO EXTRACTS LINE NUMBER FROM THE NEW CHARACTER.
:
: JSR PC,CKINAC ;CHECK FOR CHAR ON INACTIVE LINE.
BCC 4$ ;GO REPORT ERROR IF ON INACTIVE LINE.
:
: PUSH THE NEW CHARACTER ON THE RESYNC QUE FOR THIS LINE.
:
: MOV R3,R4 ;CALCULATE BASE ADDRESS OF THE
ASL R4 ; DATA PATTERN RESYNCH QUEUE

```

017544		
017544	004537	005232
017550	010305	
017552	052705	177400
017556	005037	020022
017562	004737	015344
017566	103043	
017570	010304	
017572	006304	

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 120
GLOBAL SUBROUTINE

- NEWCHR -

```

4982 017574 006304          ASL    R4          ; (QUEUE IS 4 WORDS LONG) FOR
4983 017576 062704 004550  ADD    #DPRSQB,R4 ; THIS LINE.
4984 017602 010401          MOV    R4,R1      ; GET THE BASE OF THE QUEUE.
4985 017604 016121 000002  MOV    2(R1),(R1)+ ; MOVE FROM CHR1 SLOT TO CHR0 SLOT.
4986 017610 016121 000002  MOV    2(R1),(R1)+ ; MOVE FROM CHR2 SLOT TO CHR1 SLOT.
4987 017614 010211          MOV    R2,(R1)    ; PUT NEW CHAR INTO CHR2 SLOT.
4988
4989
4990
4991
4992 017616 011402          ;+
4993 017620 100076          ; CHECK THE DATA VALID FOR THE CHARACTER AT THE BOTTON OF THE QUEUE.
4994
4995
4996
4997 017622 032702 070000  MOV    (R4),R2    ; GET CHR0 VALUE, SET FLAGS.
4998 017626 001420          BPL    60$        ; EXIT ROUTINE IF DATA.VALID IS CLEAR.
4999
5000
5001
5002
5003 017630 005337 020022          ;+
5004 017634 016300 005142          ; TEST FOR ANY OF THE ERROR BITS SET IN CHR0.
5005 017640 036037 002332 002406  BIT    #70000,R2 ; TEST FOR ANY CHR0 ERROR BITS SET.
5006 017646 001010          BEQ    2$         ; SKIP THIS ERROR IF NO ERROR BITS SET.
5007 017650 012737 013144 005230  ;+
5008 017656 004737 024142          ; WE HAVE AT LEAST ONE ERROR FLAG SET ON THE RECEIVED CHAR.
5009 017662          ; REPORT DATA ERROR FLAG ERROR IF NOT IN SUMMARY MODE.
5010 017662 104460          ;+
5011 017664 004737 024202          ; SET THE 'ERROR FOUND' FLAG.
5012
5013
5014
5015 017670 004737 014756          MOV    TXRXLB(R3),R0 ; GET THE TX LINE OFFSET FOR THIS RX LINE.
5016 017674 103424          BIT    BITTBL(R0),ERSMRF ; CHECK THE ERROR SUMMARY FLAG FOR TX LINE.
5017
5018
5019
5020 017676 005337 020022          BNE    2$         ; IF ERROR SUMMARY FLAG SET, SKIP NEXT REPORT.
5021 017702 016300 005142          MOV    #ER9003,ERRBLK ; SELECT THE ER9003 ERROR REPORT ROUTINE.
5022 017706 036037 002332 002406  JSR    PC,TXROFF   ; TURN OFF TX AND RX DURING ERROR REPORTING.
5023 017714 001014          ERROR          ; >>>> ERROR <<<<<.
5024 017716 012737 012776 005230  JSR    PC,TXRON    ; TURN TX AND RX BACK ON.
5025 017724 005237 005224          ;+
5026 017730 004737 024142          ; CHECK THE CHARACTER AT THE BOTTOM OF THE RESYNC QUE FOR DATA ERRORS.
5027 017734          ;+
5028 017734 104460          ; WE HAVE SOME SORT OF DATA ERROR SO REPORT IT (UNLESS IN SUMMARY REPORT MODE).
5029 017736 004737 024202          ;+
5030 017742 005337 005224          ; SET THE 'ERROR FOUND' FLAG.
5031
5032
5033
5034
5035 017746 005737 020022          MOV    TXRXLB(R3),R0 ; GET THE TX LINE OFFSET FOR THIS RX LINE.
5036 017752 001421          BIT    BITTBL(R0),ERSMRF ; CHECK THE ERROR SUMMARY FLAG FOR THIS LINE.
5037 017754 005263 003210          BNE    6$         ; SKIP ERROR REPORT IF ERROR SUMMARY FLAG SET.
                    MOV    #ER9002,ERRBLK ; SELECT THE ER9002 ERROR REPORT ROUTINE.
                    INC    ERNBR        ; SELECT INITIAL ERNBR + 1.
                    JSR    PC,TXROFF   ; TURN OFF TX AND RX DURING ERROR REPORTING.
                    ERROR          ; >>>> ERROR <<<<<.
                    TRAP    C$ERROR
                    JSR    PC,TXRON    ; TURN TX AND RX BACK ON.
                    DEC    ERNBR        ; RESTORE INITIAL ERNBR.
                    ;+
                    ; COUNT A CHARACTER ERROR IF ONE OCCURRED.
                    ; UPDATE THE 'REPORT ERROR SUMMARY' FLAG FOR LINE BASED ON ERROR COUNT.
                    ;+
                    ; CHECK THE 'ERROR FOUND' FLAG.
                    ; SKIP COUNTING AN ERROR IF FLAG IS CLEAR.
                    ; INCREMENT THE ERROR COUNTER FOR THIS LINE.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 121
 CVDHCA.P11 12-JUL-83 11:44 GLOBAL SUBROUTINE - NEWCHR -

```

5038 017760 001002          BNE      8$          ;SKIP SETTING COUNTER TO MAX IF NO OVERFLOW.
5039 017762 005363 003210    DEC      ERCNTB(R3)    ;RESET THE ERROR COUNTER TO -1 (MAX VALUE).
5040 017766 005737 002166    8$:     TST      NDERPT    ;DISABLE ERROR SUMMARY FUNCTION IF
5041 017772 001411          BEQ      60$          ;NUMBER OF DATA ERRORS TO REPORT IS 0.
5042 017774 026337 003210 002166    CMP      ERCNTB(R3),NDERPT ;COMPARE ERROR COUNT WITH # OF ERR'S TO RPT.
5043 020002 103405          BLO      60$          ;SKIP SETTING OF SUMMARY FLAG IF NOT TOO MANY.
5044 020004 016300 005142    MOV      TXRXLB(R3),RO  ;GET THE TX LINE OFFSET FOR THIS RX LINE.
5045 020010 056037 002332 002406    BIS      BITTBL(R0),ERSMRF ;SET 'PRINT ERROR SUMMARY' FLAG FOR LINE.
5046
5047 020016          60$:     PASS          ;RESTORE GPRS.
5048 020016 004736          JSR      PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
5049 020020 000207          RTS      PC
5050
5051 020022 000000          70$:     .WORD      0          ;LOCAL STORAGE FOR ERROR OCCURRED FLAG.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 122
GLOBAL SUBROUTINE - OOPS -

```

5052 .SBTTL GLOBAL SUBROUTINE - OOPS -
5053 :+ *****
5054 :* - PROGRAM ABORT SUBROUTINE -
5055 :* THIS SUBROUTINE IS USED TO ABORT THE PROGRAM WHEN A FATAL ERROR IS
5056 :* DETECTED IN THE PROGRAM OR THE HOST SYSTEM HARDWARE. AN ERROR MESSAGE
5057 :* IS PRINTED GIVING SOME INFORMATION ABOUT THE NATURE OF THE ABORT.
5058 :*
5059 :* INPUTS: R1 - ERROR CODE GIVING REASON FOR ABORT.
5060 :*
5061 :* OUTPUTS: AN ERROR MESSAGE IS PRINTED.
5062 :* A LIST OF RETURN PC VALUES FOR ALL SUBROUTINE CALLS IS PRINTED.
5063 :*
5064 :* CALLING SEQUENCE: JSR PC,OOPS
5065 :*
5066 :* COMMENTS:
5067 :*
5068 :* SUBORDINATE ROUTINES CALLED: NONE.
5069 :-- *****
5070
5071 020024 OOPS:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
5072 020024 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
5073 ; REPORT 'HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED.' ERROR.
5074 020030 ERRSF 101,EM0101
5075 020030 104454 TRAP C$ERSF
5076 020032 000145 .WORD 101
5077 020034 020070 .WORD EM0101
5078 020036 000000 .WORD 0
5079 ; REPORT 'PROGRAM HUNG, WAITING FOR A CONTROL-C.'
5080 020040 PRINTF #EM0102
5081 020040 012746 020154 MOV #EM0102,-(SP)
5082 020044 012746 000001 MOV #1,-(SP)
5083 020050 010600 MOV SP,R0
5084 020052 104417 TRAP C$PNTF
5085 020054 062706 000004 ADD #4,SP
5086 020060 2$: BREAK ;LOOK FOR OPERATOR CONTROL-C INPUT.
5087 020060 104422 TRAP C$BRK
5088 020062 000776 BR 2$ ;INFINITE LOOP.
5089 020064 60$: PASS ;DON'T NEED THIS, BUT SOMEBODY MAY CHANGE THIS
5090 020064 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
5091 020066 000207 RTS PC ; ROUTINE IN THE FUTURE, SO BE CONSISTANT.
5092
5093 020070 047510 052123 041440 EM0101:: .ASCIZ /HOST COMPUTER HARDWARE OR SOFTWARE BUG ENCOUNTERED./
5094 020076 046517 052520 042524
5095 020104 020122 040510 042122
5096 020112 040527 042522 047440
5097 020120 020122 047523 052106
5098 020126 040527 042522 041040
5099 020134 043525 042440 041516
5100 020142 052517 052116 051105
5101 020150 042105 000056
5102 020154 047045 040445 051120 EM0102:: .ASCIZ /%N%APROGRAM HUNG, WAITING FOR A CONTROL-C. <*****%N%N/
5103 020162 043517 040522 020115
5104 020170 052510 043516 020054
5105 020176 040527 052111 047111
5106 020204 020107 047506 020122
5107 020212 020101 047503 052116

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 123
CVDHCA.P11 12-JUL-83 11:44 GLOBAL SUBROUTINE - OOPS -

5108	020220	047522	026514	027103
5109	020226	036040	025052	025052
5110	020234	025052	025052	025052
5111	020242	025052	022452	022516
5112	020250	000116		
5113				

.EVEN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 124
GLOBAL SUBROUTINE - PRFRME -

```

5114 .SBTTL GLOBAL SUBROUTINE - PRFRME -
5115 :+ *****
5116 :* - PROCESS FRAMING ERRORS -
5117 :* THIS SUBROUTINE IS USED IN THE FRAMING ERROR BIT TEST, TO VERIFY THAT
5118 :* ALL RECEIVED CHARACTERS HAVE THEIR FRAMING ERROR BIT SET AND PARITY
5119 :* ERROR BIT CLEAR.
5120 :*
5121 :* INPUTS: R2 - CONTAINS THE CHARACTER READ FROM THE FIFO.
5122 :* ERRNBR - ERROR NUMBER OF ERRORS IN THIS ROUTINE.
5123 :* ERSMRF - 'REPORT ERROR SUMMARY FOR LINE' FLAGS
5124 :*
5125 :* OUTPUTS: ERBLK - THE CONTENTS OF THIS WORD ARE DESTROYED.
5126 :* ERCNTB - THE ERROR COUNT FOR THIS LINE IS UPDATED.
5127 :* MESSAGES MAY BE PRINTED AT THE OPERATORS CONSOLE.
5128 :*
5129 :*
5130 :* CALLING SEQUENCE: JSR PC,PRFRME
5131 :*
5132 :* COMMENTS: THIS ROUTINE REPORTS ERRORS WITH INITIAL NUMBER.
5133 :* ERRNBR IS RESTORED TO ITS INITIAL VALUE BEFORE THIS SUBROUTINE
5134 :* RETURNS.
5135 :*
5136 :* SUBORDINATE ROUTINES CALLED: ER6201.
5137 :-- *****
5138
5139 020252 PRFRME::SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
5140 020252 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
5141 020256 013704 005224 MOV ERRNBR,R4 ;SAVE THE CONTENTS OF THE INITIAL ERROR NUMBER.
5142 020262 005005 CLR R5 ;CLEAR ERROR/MESSAGE FLAGS.
5143
5144
5145 :+ TEST FRAMING AND PARITY ERROR BITS IN TURN. REPORT ANY ERRORS FOUND, IE.
5146 :+ FRAMING ERROR BIT CLEAR, OR PARITY ERROR BIT SET.
5147 :--
5148 020264 012737 012334 005230 MOV #ER6201,ERRBLK ;SET UP THE ADDRESS OF THE ERROR ROUTINE.
5149 020272 032702 020000 BIT #BIT13,R2 ;CHECK ON STATE OF THE FRAMING ERROR BIT.
5150 020276 001002 BNE 6$ ;BRANCH IF FRAMING ERROR BIT SET.
5151 020300 052705 000002 BIS #BIT1,R5 ;SET REPORT FRAMING ERROR FLAG.
5152
5153 020304 032702 010000 6$: BIT #BIT12,R2 ;CHECK ON THE STATE OF THE PARITY ERROR BIT.
5154 020310 001402 BEQ 8$ ;BRANCH IF PARITY ERROR BIT CLEAR.
5155 020312 052705 000014 BIS #14,R5 ;SET REPORT 'PARITY ERROR SET' FLAGS.
5156 020316 005705 8$: TST R5 ;CHECK IF ANY ERROR FLAGS SET.
5157 020320 001407 BEQ 60$ ;EXIT IF ALL FLAGS CLEAR.
5158 020322 036337 002332 002406 BIT BITTBL(R3),ERSMRF ;CHECK THE ERROR SUMMARY FLAG FOR THIS LINE.
5159 020330 001001 BNE 10$ ;SKIP ERROR REPORT IF ERROR SUMMARY FLAG SET.
5160
5161 ;REPORT ERROR 'CHARACTER RECEIVED WITH PARITY/FRAMING ERROR BIT SET'.
5162 020332 ERROR ; >>>> ERROR <<<<. TRAP C$ERROR
5163 020332 104460 10$: INC ERCNTB(R3) ;INCREMENT ERROR COUNT FOR THIS LINE.
5164 020334 005263 003210 60$: MOV R4,ERRNBR ;RESTORE ERROR NUMBER.
5165 020340 010437 005224 PASS ;RESTORE GPRS.
5166 020344 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
5167 020344 000207 RTS PC
5168

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 125
GLOBAL SUBROUTINE - PRPARE -

5169
5170
5171
5172
5173
5174
5175
5176
5177
5178
5179
5180
5181
5182
5183
5184
5185
5186
5187
5188
5189
5190
5191
5192
5193
5194
5195
5196
5197
5198
5199
5200
5201
5202
5203
5204
5205
5206
5207
5208
5209
5210
5211
5212
5213
5214
5215
5216
5217
5218
5219
5220
5221
5222
5223
5224

020350
020350 004537 005232
020354 013746 005224
020360 005005

020362 012737 012334 005230
020370 032702 010000
020374 001002
020376 052705 000010
020402 032702 020000
020406 001402
020410 052705 000003
020414 005705
020416 001405
020420 036337 002332 002406
020426 001024

020430
020430 104460

020432 005237 005224
020436 016304 003310

```
.SBTTL GLOBAL SUBROUTINE - PRPARE -
:++ *****
:
: - PROCESS PARITY ERRORS -
: THIS SUBROUTINE IS USED IN THE PARITY ERROR TEST, TO VERIFY THAT
: ALL RECEIVED CHARACTERS HAVE THEIR PARITY ERROR BIT SET AND FRAMMING
: ERROR BIT CLEAR.
:
: INPUTS: R2 - CONTAINS THE CHARACTER READ FROM THE FIFO.
: R3 - CONTAINS 2 * LINE NUMBER OF THE READ CHAR.
: ERRNBR - ERROR NUMBER OF ERRORS IN THIS ROUTINE.
: ERSMRF - 'REPORT ERROR SUMMARY FOR LINE' FLAGS
:
: OUTPUTS: ERRBLK - THE CONTENTS OF THIS WORD ARE DESTROYED.
: ERCNTB - THE ERROR COUNT FOR THIS LINE IS UPDATED.
: MESSAGES MAY BE PRINTED AT THE OPERATORS CONSOLE.
:
: CALLING SEQUENCE: JSR PC,PRPARE
:
: COMMENTS: THIS ROUTINE REPORTS ERRORS WITH INITIAL ERRNBR THRU ERRNBR+1.
: ERRNBR IS RESTORED TO ITS INITIAL VALUE BEFORE THIS SUBROUTINE
: RETURNS.
: THE CONTENTS OF THE ERRBLK ARE DESTROYED.
:
: SUBORDINATE ROUTINES CALLED: ER9002,ER6201.
:-- *****
PRPARE::SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV ERRNBR,-(SP) ;SAVE THE CONTENTS OF THE INITIAL ERROR NUMBER.
CLR R5 ;CLEAR ERROR/MESSAGE FLAGS.
:
:++ TEST FRAMMING AND PARITY ERROR BITS IN TURN. REPORT ANY ERRORS FOUND, IE.
: PARITY ERROR BIT CLEAR, OR FRAMMING ERROR BIT SET.
:--
MOV #ER6201,ERRBLK ;SET UP THE ADDRESS OF THE ERROR ROUTINE.
BIT #BIT12,R2 ;CHECK ON STATE OF THE PARITY ERROR BIT.
BNE 6$ ;BRANCH IF PARITY ERROR BIT SET.
BIS #BIT3,R5 ;SET REPORT PARITY ERROR FLAG.
6$: BIT #BIT13,R2 ;CHECK ON THE STATE OF THE FRAMMING ERROR BIT.
BEQ 8$ ;BRANCH IF FRAMMING ERROR BIT CLEAR.
BIS #3,R5 ;SET REPORT 'FRAMMING ERROR SET' FLAGS.
8$: TST R5 ;CHECK IF ANY ERROR FLAGS SET.
BEQ 12$ ;BRANCH TO MAKE DATA CHECK IF ALL FLAGS CLEAR.
BIT BITTBL(R3),ERSMRF ;CHECK THE ERROR SUMMARY FLAG FOR THIS LINE.
BNE 14$ ;SKIP ALL ERROR REP IF IN ERROR SUMMARY MODE.
;REPORT ERROR 'CHAR RECEIVED WITH PARITY/FRAMMING ERROR BIT SET/CLEAR'.
ERROR ; >>>> ERROR <<<<.
TRAP C$ERROR
:
:++ COMPARE ACTUAL DATA WITH EXPECTED DATA TO CHECK FOR MULTIPLE ERRORS.
:--
12$: INC ERRNBR ;INCREMENT ERROR NUMBER.
MOV RXPTRB(R3),R4 ;GET THE POINTER TO THE EXPECTED DATA.
```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-93 13:56 PAGE 126
 CVDHCA.P11 12-JUL-83 11:44 GLOBAL SUBROUTINE - PRPARE -

```

5225 020442 111404          MOVB   (R4),R4          ;GET THE EXPECTED DATA.
5226 020444 120204          CMPB   R2,R4          ;COMPARE ACTUAL AND EXPECTED DATA.
5227 020446 001424          BEQ    18$           ;SKIP ERROR REPORT IF DATA CORRECT.
5228 020450 042704 100000    BIC    #BIT15,R4      ;CLEAR 'NONE' EXPECTED MESSAGE FLAG.
5229 020454 036337 002332 002406 BIT    BITTBL(R3),ERSMRF ;CHECK THE ERROR SUMMARY FLAG FOR THIS LINE.
5230 020462 001014          BNE    16$           ;SKIP ERROR REPORT IF ERROR SUMMARY FLAG SET.
5231 020464 036337 002332 002412 BIT    BITTBL(R3),RXDNF ;CHECK FOR RECEPTION COMPLETE ON THIS LINE.
5232 020472 001402          BEQ    14$           ;SKIP SETTING NONE EXPECTED FLAG.
5233 020474 052704 100000    BIS    #BIT15,R4      ;SET 'NONE' EXPECTED MESSAGE FLAG.
5234 020500 012701 010333 14$:  MOV    #EM9008,R1     ;SELECT ERROR MESSAGE TO BE REPORTED.
5235 020504 012737 012776 005230 MOV    #ER9002,ERRBLK ;SELECT ERROR REPORTING ROUTINE.
5236                                ;REPORT ERROR'RECEIVE CHARACTER MISCOMPARE'
5237                                ERROR
5238 020512 104460                                TRAP   C$ERROR
5239
5240 020514 005263 003210 16$:  INC    ERCNTB(R3)     ;INCREMENT ERROR COUNT FOR THIS LINE.
5241 020520 012637 005224 18$:  MOV    (SP)+,ERRNBR  ;RESTORE ERROR NUMBER.
5242
5243 020524 60$:  PASS                                ;RESTORE GPRS.
5244 020524 004736                                JSR    PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
5245 020526 000207                                RTS    PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 127
GLOBAL SUBROUTINE - PRTLPR -

5246
5247
5248
5249
5250
5251
5252
5253
5254
5255
5256
5257
5258
5259
5260
5261
5262
5263
5264
5265
5266
5267
5268
5269
5270
5271
5272
5273
5274
5275
5276
5277
5278
5279
5280
5281
5282
5283
5284
5285
5286
5287
5288

020530
020530 004537 005232
020534 013701 002202
020540 013702 002206
020544 042703 177760
020550 053703 002234
020554 010311
020556 011204

020560
020560 010446
020562 012746 011037
020566 012746 007016
020572 012746 000003
020576 010600
020600 104415
020602 062706 000010

020606
020606 004736
020610 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 JSR R5,PREG05 ;SAVE CONTENTS OF GPRS R0 THRU R5.
;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:NNNNNN'
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

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

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 128
GLOBAL SUBROUTINE - PUFIFO -

5289
5290
5291
5292
5293
5294
5295
5296
5297
5298
5299
5300
5301
5302
5303
5304
5305
5306
5307
5308
5309
5310
5311
5312
5313
5314
5315
5316
5317
5318
5319
5320
5321
5322
5323
5324
5325
5326
5327
5328
5329
5330
5331
5332
5333
5334
5335
5336
5337
5338
5339

020612
020612 004537 005232
020616 012701 001000
020622 013704 002204

020626 011402
020630 100016

020632 012700 070000
020636 040200
020640 001006

020642 012700 000300
020646 040200
020650 001002
020652 004737 022400

020656 005301
020660 001362
020662 000241
020664 000401
020666 000261

020670
020670 004736

020672 000207

```

.SBTTL GLOBAL SUBROUTINE - PUFIFO -
*****
* - PURGE THE FIFO
* THIS ROUTINE TRIES TO REMOVE ALL THE CHARACTERS FROM THE FIFO.
* ANY BMP CODES THAT ARE FOUND ARE SAVED ON THE BMP CODE QUEUE.
*
* INPUTS: RBUFA- CONTAINS THE ADDRESS OF THE RECEIVER.
*
* OUTPUTS: CARRY BIT - INDICATES THE STATE OF THE FIFO, SET:= PURGED.
*          BMPQ - THE CONTENTS OF THE BMP CODE QUEUE MAY BE UPDATED.
*
* CALLING SEQUENCE: JSR PC,PUFIFO
*
* COMMENTS:
*
* SUBORDINATE ROUTINES CALLED: SAVBMP.
*****
PUFIFO::SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
              R5,PREG05 ;CALL REGISTER SAVE SUBRT.
              JSR ;SEI MAXIMUM TRY COUNT OF 512.
              MOV #512.,R1 ;GET ADDRESS OF THE RECEIVER BUFFER REGISTER.
              MOV RBUFA,R4
2$:           MOV (R4),R2 ;GET THE CONTENTS OF THE RECEIVER BUFFER REG.
              BPL 6$ ;EXIT IF THE FIFO IS EMPTY, DATA_VALID CLR.
:
: CHECK IF THE READ CHARACTER IS ACTUALLY A BMP CODE.
: IF IT IS, THEN SAVE IT ON THE BMP CODE QUEUE TO BE REPORTED LATER.
:-
              MOV #70000,R0 ;GENERATE A BIT MAP OF CHAR ERROR BITS
              BIC R2,R0 ;WHICH ARE NOT SET FOR CHAR.
              BNE 4$ ;THROW CHAR AWAY IF NOT BMP OR SELFTEST CODE.
:
: CHECK IF THE READ DATA IS MODEM STATUS , BMP OR SELFTEST?.
:-
              MOV #300,R0 ; CHECK IF BMP OR SELFTEST?.
              BIC R2,R0 ;TRY TO CLEAR BMP FLAGS IN THE READ DATA.
              BNE 4$ ;IF IT IS MODEM OR SELFTEST CODE THROW IT AWAY.
              JSR PC,SAVBMP ;SAVE BMP CODE ON THE QUEUE.
4$:           DEC R1 ;DECREMENT THE TRY COUNT.
              BNE 2$ ;LOOP TO TRY AGAIN.
              CLC ;CLEAR CARRY,TO INDICATE FIFO NOT PURGED.
              BR 60$ ;EXIT WITH CARRY CLEAR.
6$:           SEC ;SET CARRY, TO INDICATE FIFO PURGED.
60$:          PASS
              JSR ;RESTORE GPRS,
              PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
              ;CARRY BIT, SET INDICATES FIFO PURGED.
              RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 129
GLOBAL SUBROUTINE

- PUFIFR -

5340
5341
5342
5343
5344
5345
5346
5347
5348
5349
5350
5351
5352
5353
5354
5355
5356
5357
5358
5359
5360
5361
5362
5363
5364
5365
5366
5367
5368
5369
5370
5371
5372
5373
5374
5375
5376
5377
5378
5379
5380
5381
5382
5383
5384
5385
5386
5387
5388
5389
5390
5391
5392
5393
5394
5395

```

.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 NONE 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: ERRTBL - 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.
* BMPQOP - 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 REGISTER 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.
;+
2$: MOV @RBUFA,R2 ;GET THE CONTENTS OF THE RECEIVER BUFFER REG.
BPL 8$ ;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.

```

020674				
020674	004537	005232		
020700	013746	005224		
020704	012705	001000		
020710	017702	161270		
020714	100056			
020716	012700	070000		
020722	040200			
020724	001012			
020726	012737	012706	005230	
020734	012700	000300		
020740	040200			
020742	001003			
020744	004737	022400		
020750	000423			

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 130
GLOBAL SUBROUTINE - PUFIFR -

```

5396
5397 020752 032702 000001      4$: BIT #BIT0,R2      ;TEST THE MODEM STATUS INDICATION BIT.
5398 020756 001420      BEQ 6$      ;DO NOT REPORT ANY ERROR IF MODEM STATUS.
5399 020760 012701 011422      MOV #EM9104,R1    ;PASS THE CORRECT ERROR MESSAGE TO REPORT.
5400 020764 010203      MOV R2,R3        ;EXTRACT THE LINE NUMBER FROM
5401 020766 000303      SWAB R3         ; THE READ DATA.
5402 020770 042703 177760      BIC #177760,R3   ;
5403 020774 052704 100000      BIS #BIT15,R4   ;SET THE 'NONE' EXPECTED MESSAGE FLAG.
5404 021000 005237 005224      INC ERRNBR      ;SET ERROR NUMBER TO INTIAL ERRBR+1.
5405 021004 012737 012776 005230  MOV #ER9002,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
5406      ;REPORT ERROR 'UNEXPECTED DATA FOUND IN FIFO'.
5407 021012      ERROR      ; >>>>> ERROR <<<<<.
5408 021012 104460      TRAP C$ERROR
5409 021014 005337 005224      DEC ERRNBR      ;RESTORE ERROR NUMBER TO INTIAL ERRNBR.
5410
5411 021020 005305      6$: DEC R5      ;DECREMENT READ COUNTER.
5412 021022 001332      BNE 2$         ;LOOP TO READ NEXT CHAR FROM FIFO IF COUNT > 0.
5413
5414      ;+
5415      ; THE FIFO WILL NOT CLEAR, REPORT THE ERROR AND INDICATE THAT THE TEST IS TO
5416      ; BE ABORTED.
5417 021024 062737 000002 005224  ADD #2,ERRNBR    ;SET ERROR NUMBER TO INTIAL ERRNBR+2.
5418 021032 012737 012254 005230  MOV #ER1603,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
5419 021040 012701 010632      MOV #EM9017,R1  ;PASS THE MESSAGE TO BE REPORTED.
5420      ;REPORT THE ERROR 'FIFO WILL NOT PURGE, (DATA VALID STUCK SET)'
5421      ;'?????? TEST ABORTED'.
5422      ; >>>>> ERROR <<<<<.
5423 021044      TRAP C$ERROR
5424 021046 104460      CLC          ;INDICATE THE TEST IS TO BE ABORTED.
5425 021050 000401      BR 10$       ;EXIT THIS ROUTINE AND ABORT THE CURRENT TEST.
5426
5427 021052 000261      8$: SEC      ;SET THE CARRY, DO NOT ABORT THE TEST.
5428
5429 021054 012637 005224      10$: MOV (SP)+,ERRNBR ;RESTORE INITIAL ERROR NUMBER.
5430 021060      60$: PASS      ;RESTORE GPRS,
5431 021060 004736      JSR          PC,@(SP)+ ;RETURN TO PREGOS SUBRT.
5432      ;CARRY BIT, SET INDICATES FIFO PURGED, DO NOT
5433      ; ABORT THE TEST.
5434 021062 000207      RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 131
GLOBAL SUBROUTINE

- PURRXB -

5435
5436
5437
5438
5439
5440
5441
5442
5443
5444
5445
5446
5447
5448
5449
5450
5451
5452
5453
5454
5455
5456
5457
5458
5459
5460
5461
5462
5463
5464
5465
5466
5467
5468

021064
021064 004537 005232
021070 012701 002620
021074 012721 002626
021100 012721 002626
021104 005021
021106 020127 003026
021112 101774
021114
021114 004736
021116 000207

```

.SBTTL GLOBAL SUBROUTINE - PURRXB -
:++ *****
: * - PURGE THE RX BUFFER IN MEMORY ROUTINE -
: * THIS SUBROUTINE IS USED BEFORE THE BEGINNING OF A TX/RX OF DATA
: * PATTERNS TO CLEAR OUT THE RX BUFFER AND TO INITIALIZE THE VARIOUS
: * COUNTERS AND POINTERS RELATED TO THAT BUFFER.
: *
: * INPUTS: RXBSTA - LABEL AT THE BEGINNING OF THE RX BUFFER.
: *
: * OUTPUTS: RXBCNT - COUNT OF # OF CHARS IN RX BUFFER (CLEARED).
: *          RXBIPT - INPUT POINTER TO RX BUFFER (INITIALIZED).
: *          RXBOPT - OUTPUT POINTER TO RX BUFFER (INITIALIZED).
: *          THE CONTENTS OF THE RX BUFFER ARE CLEARED.
: *
: * CALLING SEQUENCE: JSR PC,PURRXB
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
PURRXB:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
                JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
                MOV #RXBOPT,R1 ;GET THE ADDRESS OF THE RX OUTPUT POINTER.
                MOV #RXBSTA,(R1)+ ;INITIALIZE THE RX BUFFER OUTPUT POINTER.
                MOV #RXBSTA,(R1)+ ;INITIALIZE THE RX BUFFER INPUT POINTER.
2$: CLR (R1)+ ;CLEAR CHAR COUNT AND THE BUFFER AREA.
        CMP R1,#RXBEND ;CHECK IF LAST LOCATION HAS BEEN CLEARED.
        BLOS 2$ ;LOOP IF NOT DONE.
60$: PASS ;RESTORE GPRS.
                JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
                RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 132
GLOBAL SUBROUTINE

- RDCHRS -

5469
5470
5471
5472
5473
5474
5475
5476
5477
5478
5479
5480
5481
5482
5483
5484
5485
5486
5487
5488
5489
5490
5491
5492
5493
5494
5495
5496
5497
5498
5499
5500
5501
5502
5503
5504
5505
5506
5507
5508
5509
5510
5511
5512
5513
5514
5515
5516
5517
5518
5519
5520
5521
5522
5523
5524

```

.SBTTL GLOBAL SUBROUTINE - RDCHRS -
++ *****
- READ AND COMPARE INPUT CHARACTERS ROUTINE -
THIS SUBROUTINE READS THE CHARACTERS FROM THE RX BUFFER IN MEMORY.
IF CHARACTERS STOP APPEARING IN THE BUFFER WITH DATA.VALID SET
OR IF MORE THAN THE ALLOWABLE NUMBER OF CHARACTERS HAS BEEN READ FROM
THE BUFFER THIS ROUTINE EXITS WITH AN RX COMPLETE INDICATION.
EACH READ CHAR IS ANALYZED AND ANY NECESSARY ERRORS ARE REPORTED.

INPUTS:  ACTLNS - BIT MAP OF THE ACTIVE DUT LINES.
         ERRNBR - SET TO ERROR NUMBER OF FIRST ERROR IN THIS ROUTINE.
         IBM - MASK OF THE INACTIVE BITS IN A TX JR RX CHAR BYTE.
         OSTEND - ADDRESS OF THE END OF THE OUTPUT STORAGE FIFO BUFFER.
         OSTPTR - POINTER TO THE NEXT BYTE TO READ FROM OSTORE.
         RXBOPT - POINTER INTO THE RX CHAR BUFFER IN MEMORY.
         RXTOUT - TIME-OUT VALUE FOR RX OF LAST CHAR.

OUTPUTS:  ERROR MESSAGES MAY BE PRINTED AT THE OPERATOR'S CONSOLE.
         TXDBLF - TX/RX DISABLED FLAG (CLEARED).
         TXENBM - TX.ENABLE STATE MASK (DESTROYED).
         SAVPRI - STORAGE FOR PROCESSOR PRIORITY (DESTROYED).
         SAVTEN - STORAGE FOR TX.ENABLE STATES (DESTROYED).

CALLING SEQUENCE:  JSR  PC,RDCHRS

COMMENTS:  THIS ROUTINE REPORTS ERRORS WITH NUMBERS INITIAL ERRNBR
           THRU INITIAL ERRNBR + 4.
           ERRNBR IS RESTORED BEFORE THIS ROUTINE RETURNS.

SUBROUTINES CALLED:  CKCHR,NEWCHR,REPCOD,RXIE0,RXIE1,TXENBL,TXIE0,TXIE1,
                   WAIBIS.
-- *****

```

```

RDCHRS:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
              R5,PREG05 ;CALL REGISTER SAVE SUBRT.
              JSR
              MOV ERRNBR,R4 ;PRESERVE THE INITIAL ERROR NUMBER.
              MOV IBM,R3 ;GET THE INACTIVE BIT MASK.
              CLR TXDBLF ;CLEAR THE TX DISABLED FLAG.
              JSR PC,RXIE1 ;TURN ON DUT RECEPTION INTERRUPTS.
              JSR PC,TXIE1 ;TURN ON DUT TRANSMISSION INTERRUPTS.

;+
; CLEAR ALL RESYNC QUEUES FOR ALL LINES.
;-
              MOV #DPRSQB,R1 ;GET BASE ADDRESS OF RESYNC QUEUES TABLE.
              MOV #DPRSQE,R2 ;GET END ADDRESS OF RESYNC QUEUES TABLE.
2$:          CLR (R1)+ ;CLEAR A WORD OF THE TABLE.
              CMP R1,R2 ;CHECK IF POINTER AT END OF TABLE.
              BLO 2$ ;LOOP UNTIL TABLE IS CLEAR.

;+
; WAIT FOR A CHARACTER TO APPEAR IN THE FIFO.
; IF NO CHARACTER APPEARS WITHIN TIME-OUT PERIOD: EXIT ROUTINE, WE'RE DONE.
;-
4$:          MOV RXTOUT,R1 ;GET TIME-OUT FOR SLOWEST BAUD RATE IN USE.
              CMP TXDNF,ACTLNS ;CHECK FOR TRANSMISSION DONE ON ACTIVE LINES.
              BEQ 6$ ;SKIP ADDING 50 MS DELAY IF TX DONE ALL LINES.

```

```

021120
021120 004537 005232
021124 013704 005224
021130 013703 002226
021134 005037 002414
021140 004737 022354
021144 004737 023642

021150 012701 004550
021154 012702 004750
021160 005021
021162 020102
021164 103775

021166 013701 002242
021172 023737 002410 002174
021200 001402

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 133
 CVDHCA.P11 12-JUL-83 11:44 GLOBAL SUBROUTINE - RDCHRS -

```

5525 021202 062701 000062          ADD    #50,R1          ;ADD 50 MILLI SEC TO DELAY IF NOT LAST CHAR.
5526 021206 052701 170000      6$:   BIS    #170000,R1    ;INDICATE TO TEST DATA VALID BIT.
5527 021212 013702 002620          MOV    RXBOPT,R2      ;INDICATE TO CHECK MEMORY RECEIVE BUFFER.
5528 021216 004737 024740          JSR    PC,WAIBIS      ;WAIT FOR RECEIVED CHAR OR TIME-OUT.
5529 021222 103076          BCC    18$           ;EXIT ROUTINE IF TIME-OUT, WE'RE DONE.
5530
5531 021224 004737 016362          JSR    PC,GETCHR      ;READ A CHARACTER FROM THE MEMORY BUFFER.
5532
5533      ;+
5534      ; CHECK IF THE TX ISR IS DISABLED.
5535      ; RE-ENABLE RX ISR IF THE SPACE FOR NEW CHARS IS LOW ENOUGH.
5536      ; IF THE BUFFER CAN ACCOMODATE MORE CHARS THEN RE-ENABLE TRANSMISSION.
5537
5537 021230 005737 002414      8$:   TST    TXDBLF          ;CHECK IF TX IS DISABLED.
5538 021234 100027          BPL    10$           ;SKIP RX/TX CHECK IF TX NOT DISABLED.
5539 021236 023727 002624 000020    CMP    RXBCNT,#RXBETX ;COMPARE BUFFER COUNT WITH LEVEL TO ENABLE RX.
5540 021244 101023          BHI    10$           ;SKIP ENABLE RX IF BUFFER TOO FULL.
5541 021246 004737 022354          JSR    PC,RXIE1      ;ENABLE RECEPTION INTERRUPTS.
5542 021252 013705 002250          MOV    TXENBM,R5     ;GET THE PRESERVED TX.ENABLE STATES.
5543 021256 023727 002624 000020    CMP    RXBCNT,#RXBETX ;COMPARE BUFFER COUNT WITH LEVEL TO ENABLE TX.
5544 021264 101013          BHI    10$           ;SKIP ENABLING TX IF BUFFER TOO FULL.
5545 021266          GETPRI R1          ;SAVE THE CURRENT PROCESSOR PRIORITY.
5546 021266 104440          TRAP    CS$PRI
5547 021270 010001          MOV    R0,R1
5548 021272          SETPRI #PRI07        ;DISABLE INTERRUPTS.
5549 021272 012700 000340          MOV    #PRI07,R0
5550 021276 104441          TRAP    C$SPRI
5551 021300 004737 023412          JSR    PC,TXENBL     ;ENABLE TRANSMISSION.
5552 021304 005037 002414          CLR    TXDBLF        ;CLEAR THE TX DISABLE FLAG.
5553 021310          SETPRI R1          ;RE-ENABLE INTERUPTS.
5554 021310 010100          MOV    R1,R0
5555 021312 104441          TRAP    C$SPRI
5556 021314
5557
5558 021314 005337 002404          DEC    CHRTOT        ;DECREMENT THE TOTAL CHAR COUNTER.
5559 021320 001011          BNE    12$           ;SKIP ERROR IF NOT TOO MANY RECEIVED.
5560 021322 010437 005224          MOV    R4,ERRNBR     ;SET ERROR NUMBER TO INITIAL ERRNBR.
5561 021326 012701 010743          MOV    #EM9025,R1    ;SELECT THE PROPER ERROR MESSAGE.
5562 021332 012737 012230 005230    MOV    #ER0503,ERRBLK ;SELECT THE PROPER ERROR REPORT ROUTINE.
5563
5564      ;+
5565      ; REPORT ERROR AT INITIAL ERRNBR.
5566      ; "MORE THAN TWICE THE EXPECTED NUMBER OF CHARACTERS RECEIVED."
5567
5567 021340          ERROR          ; >>>> ERROR <<<<<.
5568 021340 104460          TRAP    C$ERROR
5569 021342 000452          BR    60$           ;EXIT THE ROUTINE, WE'RE GIVING UP.
5570
5571      ;+
5572      ; DETERMINE IF THE CHARACTER IS DATA OR A STATUS CODE.
5573
5573 021344 012700 070000      12$:  MOV    #70000,R0     ;GENERATE A BIT MAP OF CHARACTER ERROR BITS
5574 021350 040200          BIC    R2,R0         ; WHICH ARE NOT SET FOR THE CHARACTER.
5575 021352 001007          BNE    14$           ;SKIP REPORTING OF ERROR CODE IF WE HAVE CHAR.
5576
5577      ;+
5578      ; THE DATA IS EITHER A BMP CODE OR A MODEM STATUS CODE.
5579      ; REPORT THAT THE CODE WAS FOUND.
5580      ; ERRORS REPORTED WITH ERROR NUMBERS >>>> ERRNBR+1 AND ERRNBR+2 <<<<<.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 134
GLOBAL SUBROUTINE

- RDCHRS -

```

5581 021354 010437 005224      MOV    R4,ERRNBR      ;GET THE ERROR NUMBER PASSED INTO THIS ROUTINE.
5582 021360 005237 005224      INC    ERRNBR        ;SET ERROR NUMBER TO INITIAL ERRNBR+1.
5583 021364 004737 021532      JSR    PC,REPCOD     ;REPORT THE BMP OR MODEM STATUS CHANGE CODE.
5584 021370 000407              BR     16$          ;BRANCH TO GET THE NEXT CHARACTER.
5585
5586      ;+ THE DATA IS A VALID CHARACTER:
5587      ; COMPARE THE READ DATA WITH THE EXPECTED DATA.
5588      ; UPDATE EXPECTED DATA POINTER.
5589      ; ERRORS REPORTED WITH ERROR NUMBERS >>>> ERRNBR+3 AND ERRNBR+4 <<<<.
5590
5591 021372 010437 005224      14$:  MOV    R4,ERRNBR      ;CALCULATE THE STARTING ERROR NUMBER FOR THE
5592 021376 062737 000003      ADD    #3,ERRNBR     ; NEXT ROUTINE CALL (INITIAL ERRNBR+3).
5593 021404 004737 017544      JSR    PC,NEWCHR     ;HANDLE THE NEW DATA CHARACTER.
5594
5595      ;+ DONE PROCESSING THIS CHARACTER.
5596      ; READ ANOTHER CHAR FROM THE DUT FIFO.
5597      ; IF DATA.VALID IS SET, LOOP TO CHECK THE RECEIVED CHARACTER.
5598      ; IF DATA.VALID IS CLEAR LOOP TO WAIT FOR IT SET OR TIME-OUT.
5599
5600 021410 004737 016362      16$:  JSR    PC,GETCHR     ;READ A CHARACTER FROM THE RX BUFFER.
5601 021414 103705              BCS   8$            ;IF DATA.VALID SET, GO TO CHECK THE RX CHAR.
5602 021416 000663              BR    4$            ;LOOP TO WAIT CHAR OR TIME-OUT IF BUFFER EMPTY.
5603
5604      ;+ USE DUMMY CHARACTERS TO FORCE ANALYSIS OF CHARACTERS IN RESYNC QUEUES.
5605
5606 021420 004737 022314      18$:  JSR    PC,RXIEO     ;TURN OFF DUT RX INTERRUPTS.
5607 021424 004737 023206      JSR    PC,TXDONE    ;CHECK IF TX DONE, TURN OFF DUT TX INTERRUPTS.
5608 021430 005002              CLR    R2           ;CLEAR THE DUMMY CHARACTER.
5609 021432 005001              CLR    R1           ;CLEAR THE LOOP COUNTER.
5610 021434 004737 017544      20$:  JSR    PC,NEWCHR     ;FORCE ONE RESYNC QUE CHAR TO BE ANALYZED.
5611 021440 062702 000400      ADD    #400,R2      ;INCREMENT THE LINE NUMBER IN THE DUMMY CHAR.
5612 021444 005201              INC    R1           ;INCREMENT THE LOOP COUNTER.
5613 021446 120127 000010      CMPB  R1,#NUMLNS    ;TEST FOR LOOP COUNTER EQUAL TO # OF DUT LINES.
5614 021452 002770              BLT   20$          ;LOOP IF LOOP COUNT IS NOT ALL LINES DONE.
5615 021454 005701              TST   R1           ;CHECK FOR SECOND TIME AROUND OUTER LOOP.
5616 021456 100404              BMI  60$          ;EXIT IF OUTER LOOP DONE TWICE.
5617 021460 005002              CLR    R2           ;CLEAR THE DUMMY CHAR FOR 2ND TIME AROUND LOOP.
5618 021462 012701 100000      MOV    #100000,R1   ;CLEAR LOOP COUNT, SET OUTER LOOP FLAG.
5619 021466 000762              BR    20$          ;LOOP THE SECOND TIME AROUND OUTER LOOP.
5620
5621 021470 010437 005224      60$:  MOV    R4,ERRNBR     ;RESTORE THE ERROR NUMBER TO ITS INITIAL VALUE.
5622 021474              PASS              ;RESTORE GPRS.
5623 021474 004736              JSR    PC,@(SP)+   ;RETURN TO PREG05 SUBRT.
5624 021476 000207              RTS    PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 135
GLOBAL SUBROUTINE

- RDMAST -

5625
5626
5627
5628
5629
5630
5631
5632
5633
5634
5635
5636
5637
5638
5639
5640
5641
5642
5643
5644
5645
5646
5647
5648
5649
5650
5651
5652
5653
5654
5655
5656
5657
5658
5659
5660
5661
5662
5663

```

.SBTTL GLOBAL SUBROUTINE - RDMAST -
:++ *****
: * - REPORT DMA START BIT ERRORS ROUTINE -
: * THIS SUBROUTINE CHECKS FOR LINES WHICH HAVE DMA_START BIT ERRORS
: * DURING THE JUST COMPLETED DMA TRANSMISSION. IF ANY ARE FOUND,
: * THEY ARE REPORTED.
: * INPUTS: ERRMSG - ADDRESS OF PRIMARY ERROR MESSAGE FOR THIS ROUTINE.
: * ERRNBR - ERROR NUMBER OF ERROR REPORTED IN THIS ROUTINE.
: * TXINTF - CONTAINS BIT MAP OF LINES WITH DMA_START BIT ERRORS.
: * OUTPUTS: ERRBLK - ADDRESS OF THE ERROR REPORTING ROUTINE (DESTROYED).
: * MESSAGES MAY BE PRINTED AT THE OPERATOR CONSOLE.
: * CALLING SEQUENCE: JSR PC,RDMAST
: * COMMENTS: IF NO LINES HAVE DMA_START BIT ERRORS, NO MESSAGES ARE PRINTED.
: * SUBORDINATE ROUTINES CALLED: ER9102.
:-- *****

```

```

RDMAST:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;CALL REGISTER SAVE SUBRT.
JSR R5,PREG05 ;GET COPY OF THE DMA_START ERRORS BIT MAP.
MOV TXINTF,R2 ;EXIT IF NO DMA_START ERROR BITS ARE SET.
BEQ 60$

```

```

: + WE HAVE SOME DMA_START BIT ERRORS TO REPORT.
: -
MOV #ER9102,ERRBLK ;SELECT THE ERROR REPORTING ROUTINE.
MOV #EM9102,R1 ;INDICATE THAT WE HAVE DMA_START BIT ERROR.

```

```

: + REPORT 'DMA_START BIT SET AFTER RESET OR TX.ACTION ... ON LINES(S):'
: -
ERROR ; >>>> ERROR <<<<.
; TRAP C$ERROR

```

```

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

```

```

021500
021500 004537 005232
021504 013702 002252
021510 001406
021512 012737 013656 005230
021520 012701 011333
021524 104460
021526
021526 004736
021530 000207

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 136
GLOBAL SUBROUTINE

- REPCOD -

5664
5665
5666
5667
5668
5669
5670
5671
5672
5673
5674
5675
5676
5677
5678
5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5693
5694
5695
5696
5697
5698
5699
5700
5701
5702
5703
5704
5705
5706
5707
5708
5709
5710
5711
5712
5713
5714
5715
5716
5717
5718
5719

021532
021532 004537 005232
021536 012737 012706 005230
021544 013703 005224
021550 010204
021552 000304
021554 042704 177760

021560 012701 010135
021564 032702 000001
021570 001422
021572 005237 005224
021576 012701 010157
021602 012700 000300
021606 040200
021610 001003
021612 004737 022400
021616 000420
021620 122702 000201
021624 001413
021626 122702 000203
021632 001410
021634 000400

021636 042702 177400
021642 004737 024142
021646
021646 104460
021650 004737 024202

```
.SBTTL GLOBAL SUBROUTINE - REPCOD -
:++ *****
: - ROUTINE TO REPORT ERROR CODE FROM DUT -
: THIS ROUTINE REPORTS AN ERROR CODE WHICH HAS BEEN READ FROM THE DUT
: FIFO. THE CODE IS CHECKED TO DETERMINE WHETHER IT IS A SELFTEST CODE
: AN MODEM STATUS CHANGE CODE OR A BMP CODE. THIS ROUTINE ASSUMES THAT
: THE CODE INDICATES AN ERROR. IF A BMP CODE IS FOUND IT IS NOT REPORTED
: IMMEDIATELY, BUT IS SAVED ON THE BMP CODE QUEUE TO BE REPORTED LATER.
:
: INPUTS: R2 - CONTAINS THE ERROR CODE COMPLETE WITH FLAGS AND LINE #.
: ERRIBL - ERRIBL,ERRNBR,AND ERRMSG SET UP CORRECTLY.
:
: OUTPUTS: ERRBLK - VALUE MAY BE DESTROYED.
: BMPCPQ - MAYBE UPDATED IF A BMP CODE IS ADDED TO THE QUEUE.
:
: CALLING SEQUENCE: JSR PC,REPCOD
:
: COMMENTS: ERRNBR IS RESTORED TO ITS ENTERING VALUE BY THIS ROUTINE.
: THIS ROUTINE REPORTS ERRORS WITH NUMBERS ERRNBR THRU ERRNBR+1.
:
: SUBORDINATE ROUTINES CALLED: ER9001,SAVBMP.
:-- *****
REPCOD:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
JSR #ER9001,ERRBLK ;SELECT THE ERROR REPORTING ROUTINE.
MOV ERRNBR,R3 ;PRESERVE THE ERROR NUMBER.
MOV R2,R4 ;EXTRACT THE LINE NUMBER FIELD
SWAB R4 ; FROM THE ERROR CODE WHICH WAS
BIC #177760,R4 ; PASSED INTO THIS ROUTINE.
:
:++ DETERMINE THE TYPE OF CODE WHICH IS TO BE REPORTED.
:--
MOV #EM9003,R1 ;SELECT MODEM STATUS CODE MESSAGE.
BIT #BIT0,R2 ;TEST THE MODEM STATUS INDICATION BIT.
BEQ 4$ ;GOTO REPORT ERROR IF MODEM STATUS CODE.
INC ERRNBR ;SELECT THE SELFTEST CODE ERROR NUMBER.
MOV #EM9004,R1 ;SELECT SELFTEST CODE MESSAGE.
MOV #300,R0 ;CHECK IF SELF-TEST OR BMP CODE.
BIC R2,R0 ;TRY TO CLEAR BMP BITS.
BNE 2$ ;GO CHECK FOR SELFTEST CODE IF NOT BMP.
JSR PC,SAVBMP ;SAVE THE BMP CODE ON THE QUEUE.
BR 60$ ;EXIT THIS ROUTINE.
2$: CMPB #201,R2 ;CHECK FOR SELF TEST NULL CODE.
BEQ 6$ ;EXIT ROUTINE IF NULL CODE FOUND.
CMPB #203,R2 ;CHECK FOR SKIP SELF TEST CODE.
BEQ 6$ ;EXIT ROUTINE IF SKIP SELF TEST CODE FOUND.
BR 4$ ;GO REPORT SELF TEST ERROR.
:
:++ REPORT 'UNEXPECTED XXXXX CODE FOUND IN RECEIVE CHAR FIFO.'
:--
4$: BIC #177400,R2 ;REMOVE UPPER BYTE OF CODE TO BE REPORTED.
JSR PC,TXROFF ;TURN OFF TX AND RX DURING ERROR REPORTING.
; >>>> ERROR <<<<<.
; TRAP C$ERROR
JSR PC,TXRON ;TURN TX AND RX BACK ON.
:++
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 137
GLOBAL SUBROUTINE - REPCOD -

```

5720 ; RESTORE THE INITIAL ERROR NUMBER.
5721 ;
5722 021654 010337 005224 60: MOV R3,ERRNBR
5723 ;
5724 021660 60$: PASS ;RESTORE GPRS.
5725 021660 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
5726 021662 000207 RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 138
GLOBAL SUBROUTINE

- REPSMR -

5727
5728
5729
5730
5731
5732
5733
5734
5735
5736
5737
5738
5739
5740
5741
5742
5743
5744
5745
5746
5747
5748
5749
5750
5751
5752
5753
5754
5755
5756
5757
5758
5759
5760
5761
5762
5763
5764
5765
5766
5767
5768
5769
5770

```

.SBTTL GLOBAL SUBROUTINE - REPSMR -
+*****
+ - REPORT ERROR SUMMARY ROUTINE -
+ THIS SUBROUTINE REPORTS AN ERROR SUMMARY FOR THOSE LINES WHICH HAVE
+ EXCEEDED THE NUMBER OF INDIVIDUAL ERRORS TO REPORT FOR A SINGLE LINE
+ IN A SINGLE TEST. THIS PARAMETER CAN BE SPECIFIED BY THE OPERATOR IF
+ HE/SHE ANSWERS THE SOFTWARE PARAMETER QUESTIONS.
+
+ INPUTS: ERCNTB - LABEL AT BASE OF LINE ERROR COUNTERS TABLE.
+          ERRMSG - ADDRESS OF PRIMARY ERROR MESSAGE.
+          ERRNBR - ERROR NUMBER OF ERRORS IN THIS ROUTINE.
+          ERSMRF - 'REPORT ERROR SUMMARY FOR LINE' FLAGS.
+
+ OUTPUTS: ERRBLK - ADDRESS OF ERROR REPORTING ROUTINE (DESTROYED).
+          SUMMARY MESSAGES MAY BE PRINTED AT THE OPERATOR CONSOLE.
+
+ CALLING SEQUENCE: JSR PC,REPSMR
+
+ COMMENTS: IF NO LINES HAVE EXCEEDED THE MAXIMUM NUMBER OF INDIVIDUAL
+           ERRORS TO REPORT, NO MESSAGES ARE PRINTED BY THIS ROUTINE.
+           ERROR SUMMARIES IN THIS ROUTINE ARE REPORTED AS ERRORS.
+           THE CONTENTS OF ERRBLK ARE DESTROYED.
+
+ SUBORDINATE ROUTINES CALLED:
+-----*****
REPSMR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          TST ERSMRF ;CHECK THE 'PRINT LINE ERROR SUMMARY' FLAGS.
          BEQ 60$ ;EXIT WITHOUT ACTION IF NO SUMMARY FLAGS SET.
+
+ WE HAVE SOME ERROR SUMMARIES TO REPORT.
+-----
          MOV #ER9004,ERRBLK ;SELECT ERROR REPORTING ROUTINE.
+
+ REPORT
+ 'ERROR SUMMARY REPORT FOR LINES WITH EXCESSIVE NUMBERS OF ERRORS:'
+-----
          ERROR
          TRAP C$ERROR
+
60$: PASS ;RESTORE GPRS.
          JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
          RTS PC

```

```

021664
021664 004537 005232
021670 005737 002406
021674 001404
021676 012737 013326 005230
021704
021704 104460
021706
021706 004736
021710 000207

```

CVDHCAO DHV-11 FUPC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 139
GLOBAL SUBROUTINE

- RESETT -

5771
5772
5773
5774
5775
5776
5777
5778
5779
5780
5781
5782
5783
5784
5785
5786
5787
5788
5789
5790
5791
5792
5793
5794
5795
5796
5797
5798
5799
5800
5801
5802
5803
5804
5805
5806
5807
5808
5809
5810
5811
5812
5813
5814
5815
5816
5817
5818
5819
5820
5821
5822
5823
5824
5825
5826

021712
021712 004537 005232
021716 012702 000040

021722 013704 002202
021726 030214
021730 001406
021732 005003
021734 012701 004704
021740 004737 017340
021744 103012

021746 010277 160230
021752 004737 022446

021756 005003
021760 012701 004704
C21764 004737 017340

```

.SBTTL GLOBAL SUBROUTINE - RESETT -
*****
- RESET DEVICE UNDER TEST -
THIS SUBROUTINE IS USED TO RESET THE DUT TO A KNOWN STATE.
IF RESET DOES NOT SUCCESSFULLY COMPLETE, IE. TIME-OUT OCCURS, THEN
AN ABORT TEST ERROR MESSAGE IS REPORTED.

INPUTS: CSRA - CONTAINS THE ADDRESS OF THE CSR
TXBFCA - CONTAINS ADDRESS OF DUT DMA BUFFER COUNT REGISTER.
ERRTBL- ERRTP,ERNBR,AND ERRMSG SET UP CORRECTLY.

OUTPUTS: THE DUT PERFORMS ITS RESET FUNCTION INTO A KNOWN STATE.
CARRY - CLEAR INDICATES THE TEST IS TO BE ABORTED.
ERRBLK - VALUE MAY BE DESTROYED.
IESTAT - TX AND RX INTERRUPT FLAGS ARE CLEARED.
TX AND RX INTERRUPT ENABLE BITS IN THE DUT'S CSR ARE CLEARED.

CALLING SEQUENCE: JSR PC,RESETT

COMMENTS: THIS SUBROUTINE CAN REPORT ERRORS WITH NUMBERS INITIAL ERRNBR
THIS ROUTINE DOES NOT DESTROY THE VALUE OF ERRNBR.

SUBORDINATE ROUTINES CALLED: DELAY,MSLGET.
*****
RESETT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV #BIT05,R2 ;SET BIT MASK OF MASTER RESET BIT.

;+
; TEST THE STATE OF THE MASTER RESET BIT IN THE CSR.
; IF MR IS SET THEN WAIT FOR SELF-TEST TO COMPLETE.
; IF TIME-OUT OCCURS, REPORT THE ERROR AND PASS-OUT ABORT TEST INDICATOR.
;-
MOV CSRA,R4 ;GET THE ADDRESS OF THE DUT'S CSR.
BIT R2,(R4) ;CHECK STATE OF MASTER RESET BIT.
BEQ 2$ ;DON'T DELAY IF MR IS ALREADY CLEAR.
CLR R3 ;SET UP DESIRED STATE OF MASTER RESET BIT.
MOV #2500.,R1 ;PASS TIME-OUT VALUE OF 2.5 SECONDS.
JSR PC,MSLGET ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.
BCC 4$ ;GO REPORT ERROR IF TIMEOUT OCCURRED.

;+
; SET MASTER RESET BIT IN CSR. CLEAR TX AND RX ENABLE BITS, ETC.
; SKIP THE SELFTEST.
; TIME-OUT OF 2.5 SECS, JUST IN CASE THE SELF-TEST EXECUTES.
;-
2$: MOV R2,@CSRA ;SET MASTER RESET BIT, DISABLE TX AND RX INTS.
JSR PC,SKPSTS ;TRY TO SKIP THE SELFTEST.

;+
; SET SELF-TEST TIME-OUT OF 2.5 SECONDS, AND WAIT FOR M.R TO CLEAR.
; IF TIME-OUT OCCURS, THEN REPORT THE FATAL ERROR AND PASS-OUT THE ABORT
; TEST INDICATOR.
;-
CLR R3 ;SET UP DESIRED STATE OF MASTER RESET BIT.
MOV #2500.,R1 ;PASS TIME-OUT VALUE OF 2.5 SECONDS.
JSR PC,MSLGET ;WAIT FOR SELF-TEST TO COMPLETE, MR CLEAR.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 140
GLOBAL SUBROUTINE

- RESETT -

```

5827 021770 103410          BCS      6$          ;SKIP ERROR REPORT IF MR CLEARED IN TIME.
5828
5829                      ;+
5830                      ; SET UP ERROR MESSAGE TO REPORT 'FATAL ERROR FOUND DURING RESET,TEST ABORTED'.
5831                      ; INDICATE TEST IS TO BE ABORTED BY CLEARING THE CARRY BIT.
5832 021772 012701 007377 4$:      MOV      #EM1601,R1      ;PASS ERROR MESSAGE TO REPORT.
5833 021776 012737 012254 005230  MOV      #ER1603,ERRBLK ;PASS ADDRESS OF ERROR HANDLING ROUTINE.
5834                      ;REPORT ERROR 'TIME-OUT OCCURRED WAITING FOR MASTER RESET TO CLEAR'
5835                      ; 'TEST ABORTED'
5836 022004                      ;          >>>>> ERROR <<<<<
5837 022004 104460                      ;          TRAP      C$ERROR
5838 022006 000241          CLC
5839 022010 000403          BR       60$          ;INDICATE TEST IS TO BE ABORTED.
5840                      ;EXIT THIS SUBROUTINE, ABORT TEST INDICATOR.
5841                      ;+
5842                      ; CLEAR TX AND RX INTERRUPT ENABLE STATUS FLAGS IN IESTAT.
5843                      ; EXIT WITH CONTINUE TEST INDICATOR SET (IE,CARRY SET).
5844 022012 005037 002234 6$:      CLR      IESTAT      ;CLEAR TX AND RX INTERRUPT STATUS FLAGS.
5845 022016 000261          SEC
5846                      ;INDICATE SUCCESS, CONTINUE TEST.
5847 022020          60$:      PASS
5848 022020 004736          JSR
5849                      ;RESTORE GPRS, PASS THE FOLLOWING INTACT:
5850 022022 000207          RTS      PC          ;PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
5851                      ;CARRY BIT:IF CLEAR,INDICATES ABORT TEST.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 141
GLOBAL SUBROUTINE

- RRXNDN -

5852
5853
5854
5855
5856
5857
5858
5859
5860
5861
5862
5863
5864
5865
5866
5867
5868
5869
5870
5871
5872
5873
5874
5875
5876
5877
5878
5879
5880
5881
5882
5883
5884
5885
5886
5887
5888
5889
5890
5891
5892
5893
5894
5895
5896
5897
5898

```

.SBTTL GLOBAL SUBROUTINE - RRXNDN -
*****
- REPORT RECEPTION NOT COMPLETED ROUTINE -
THIS SUBROUTINE CHECKS FOR LINES WHICH DID NOT RECEIVE THE COMPLETE
DATA PATTERN. IF ANY ARE FOUND, THEY ARE REPORTED.

INPUTS:  R5 - LOCAL ACTIVE LINES BIT MAP.
          DPLENB - BASE OF TABLE OF DATA PATTERN LENGTHS.
          ERRMSG - ADDRESS OF PRIMARY ERROR MESSAGE FOR THIS ROUTINE.
          ERRNBR - ERROR NUMBER OF ERROR REPORTED IN THIS ROUTINE.
          RXCNTB - LABEL AT BASE OF THE RX CHARACTER COUNTERS TABLE.
          RXDNF - RECEPTION DONE FLAGS.

OUTPUTS: ERRBLK - ADDRESS OF THE ERROR REPORTING ROUTINE (DESTROYED).
          MESSAGES MAY BE PRINTED AT THE OPERATOR CONSOLE.

CALLING SEQUENCE: JSR PC,RRXNDN

COMMENTS: IF NO LINES FAILED TO COMPLETE THEIR RECEPTION, NO MESSAGES
          ARE PRINTED.

SUBORDINATE ROUTINES CALLED: ER9005.
*****

```

```

RRXNDN:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          MOV R5,R2 ;GET COPY OF THE LOCAL ACTIVE LINES BIT MAP.
          BIC RXDNF,R2 ;GET MAP OF ACTIVE LINES WITH RX DONE FLAG CLR.
          BEQ 60$ ;EXIT IF NO ACTIVE LINES HAVE RX DONE FLAG CLR.

```

```

+ WE HAVE SOME 'RX NOT COMPLETED' ERRORS TO REPORT.
-
MOV #ER9005,ERRBLK ;SELECT THE ERROR REPORTING ROUTINE.
MOV #EM9016,R1 ;INDICATE THAT WE ARE DEALING WITH RECEPTION.
MOV #RXCNTB,R4 ;PASS BASE OF RX CHAR COUNTERS TABLE TO ER9005.

```

```

+ REPORT 'SINGLE CHARACTER MODE TEST ERROR:'
+ 'DATA PATTERN NOT COMPLETELY RECEIVED ON ALL LINES:'
+ ...
-
ERROR TRAP C$ERROR

```

```

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

```

```

022024
022024 004537 005232
022030 010502
022032 043702 002412
022036 001410

022040 012737 013430 005230
022046 012701 010623
022052 012704 003450

022056
022056 104460

022060
022060 004736
022062 000207

```

CVDHCAJ DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 142
GLOBAL SUBROUTINE - RTXNDN -

```

5899 .SBTTL GLOBAL SUBROUTINE - RTXNDN -
5900 :+ *****
5901 : - REPORT TRANSMISSION NOT COMPLETED ROUTINE -
5902 : THIS SUBROUTINE CHECKS FOR LINES WHICH DID NOT TRANSMIT THE COMPLETE
5903 : DATA PATTERN. IF ANY ARE FOUND, THEY ARE REPORTED.
5904 :
5905 : INPUTS: R5 - LOCAL ACTIVE LINES BIT MAP.
5906 : DPLENB - LABEL AT BASE OF DATA PATTERN LENGTH TABLE.
5907 : ERRMSG - ADDRESS OF PRIMARY ERROR MESSAGE FOR THIS ROUTINE.
5908 : ERRNBR - ERROR NUMBER OF ERROR REPORTED IN THIS ROUTINE.
5909 : TXCNTB - LABEL AT BASE OF THE TX CHARACTER COUNTERS TABLE.
5910 : TXDNF - TRANSMISSION DONE FLAGS.
5911 :
5912 : OUTPUTS: ERRBLK - ADDRESS OF THE ERROR REPORTING ROUTINE (DESTROYED).
5913 : MESSAGES MAY BE PRINTED AT THE OPERATOR CONSOLE.
5914 :
5915 : CALLING SEQUENCE: JSR PC,RTXNDN
5916 :
5917 : COMMENTS: IF NO LINES FAILED TO COMPLETE THEIR TRANSMISSION, NO MESSAGES
5918 : ARE PRINTED.
5919 :
5920 : SUBORDINATE ROUTINES CALLED: ER9005.
5921 :-- *****
5922 :
5923 022064 RTXNDN:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
5924 022064 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
5925 022070 010502 MOV R5,R2 ;GET COPY OF THE LOCAL ACTIVE LINES BIT MAP.
5926 022072 043702 002410 BIC TXDNF,R2 ;GET MAP OF ACTIVE LINES WITH TX DONE FLAG CLR.
5927 022076 001410 BEQ 60$ ;EXIT IF NO ACTIVE LINES HAVE TX DONE FLAG CLR.
5928 :
5929 :+ WE HAVE SOME 'TX NOT COMPLETED' ERRORS TO REPORT.
5930 :-
5931 022100 012737 013430 005230 MOV #ER9005,ERRBLK ;SELECT THE ERROR REPORTING ROUTINE.
5932 022106 012701 010607 MOV #EM9015,R1 ;INDICATE WE ARE DEALING WITH TRANSMISSION.
5933 022112 012704 003410 MOV #TXCNTB,R4 ;PASS BASE OF TX CHAR COUNTERS TO TABLE ER0805.
5934 :
5935 :+ REPORT 'SINGLE CHARACTER MODE TEST ERROR:'
5936 : 'DATA PATTERN NOT COMPLETELY TRANSMITTED ON ALL LINES:'
5937 : ...
5938 :-
5939 :
5940 022116 ERROR ; >>>> ERROR <<<<.
5941 022116 104460 TRAP C$ERROR
5942 :
5943 :
5944 022120 60$: PASS ;RESTORE GPRS.
5945 022120 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
5946 022122 000207 RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 143
GLOBAL SUBROUTINE - RXDSBL -

```

5947 .SBTTL GLOBAL SUBROUTINE - RXDSBL -
5948 :+ *****
5949 :* - DISABLE RECEIVERS -
5950 :* THIS SUBROUTINE IS USED TO DISABLE RECEPTION ON SELECTED LINES BY,
5951 :* CLEARING THE ASSOCIATED RX_ENABLE BIT ON THE DUT.
5952 :*
5953 :* INPUTS: R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO CLEAR RX_ENABLE.
5954 :* CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
5955 :* IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
5956 :* NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
5957 :* LNCTRA - CONTAINS THE ADDRESS OF THE LNCTRL REGISTER.
5958 :*
5959 :* OUTPUTS: R5 - BIT'S SET INDICATE INITIAL STATES OF ALL RX_ENABLE BITS.
5960 :* LNCTRA - THE STATE OF THE RX_ENABLE BIT MAY BE ALTERED.
5961 :* THE CONTENTS OF THE IND_ADD_REG FIELD IN THE CSR ARE DESTROYED.
5962 :*
5963 :* CALLING SEQUENCE: JSR PC,RXDSBL
5964 :*
5965 :* COMMENTS:
5966 :*
5967 :* SUBORDINATE ROUTINES CALLED: NONE.
5968 :-- *****
5969
5970 RXDSBL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
5971 022124 004537 005232 JSR R5,PREGOS ;CALL REGISTER SAVE SUBRT.
5972 022130 010500 MOV R5,R0 ;COPY BIT MAP OF LINES TO DISABLE RECEPTION.
5973 022132 012701 000001 MOV #BIT0,R1 ;INITIALIZE THE SELECTED LINE BIT MASK.
5974 022136 013702 002212 MOV LNCTRA,R2 ;GET THE ADDRESS OF THE LNCTRL REGISTER.
5975 022142 012703 000010 MOV #NUMLNS,R3 ;GET MAXIMUM LINE NUMBER PLUS ONE.
5976 022146 013704 002234 MOV IESTAT,R4 ;GET THE STATES OF THE INT ENABLE BITS.
5977 022152 005005 CLR R5 ;LOG POSSIBLE RX DISABLED ON ALL LINES.
5978
5979 :+ SELECT EVERY LINE IN TURN, AND LOG THE STATE OF EACH RX_ENABLE BIT.
5980 :-
5981 022154 010477 160022 2$: MOV R4,@CSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
5982 022160 032712 000004 BIT #BIT2,(R2) ;CHECK STATE OF RX_ENABLE BIT ON SELECTED LINE.
5983 022164 001401 BEQ 4$ ;SKIP NEXT INSTRUCTION IF RX_ENABLE CLEAR.
5984 022166 050105 BIS R1,R5 ;LOG RX_ENABLE BIT SET FOR SELECTED LINE.
5985
5986 :+ CLEAR RX_ENABLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE RX_DISABLE
5987 :+ LINE BIT_MAP.
5988 :-
5989 022170 030100 4$: BIT R1,R0 ;CHECK STATE OF DISABLE LINE BIT MAP.
5990 022172 001402 BEQ 6$ ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
5991 022174 042712 000004 BIC #BIT2,(R2) ;CLEAR RX_ENABLE BIT ON SELECTED LINE.
5992 022200 005204 6$: INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
5993 022202 006301 ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
5994 022204 005303 DEC R3 ;DECREMENT LINE NUMBER.
5995 022206 001362 BNE 2$ ;LOOP TO CHECK NEXT LINE.
5996
5997 022210 60$: PASS R5 ;RESTORE GPRS,EXCEPT
5998 022210 010506 000014 MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
5999 022214 004736 JSR PC,@(SP)+ ;RETURN TO PREGOS SUBRT.
6000
6001 022216 000207 RTS PC ;R5 - PREVIOUS STATES OF ALL RX_ENABLE BITS.

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 144
GLOBAL SUBROUTINE - RXENBL -

6002
6003
6004
6005
6006
6007
6008
6009
6010
6011
6012
6013
6014
6015
6016
6017
6018
6019
6020
6021
6022
6023
6024
6025
6026
6027
6028
6029
6030
6031
6032
6033
6034
6035
6036
6037
6038
6039
6040
6041
6042
6043
6044
6045
6046
6047
6048
6049
6050
6051
6052
6053
6054
6055
6056
6057

```
.SBTTL GLOBAL SUBROUTINE - RXENBL -
:++ *****
: * - ENABLE RECEIVER -
: * THIS SUBROUTINE IS USED TO ENABLE RECEPTION ON SELECTED LINES BY
: * SETTING THE ASSOCIATED RX.ENABLE BIT ON THE DUT.
: *
: * INPUTS: R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO SET RX.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.
: * LNCTRA - CONTAINS THE ADDRESS OF THE LNCTRL REGISTER.
: *
: * OUTPUTS: R5 - BIT'S SET INDICATE PREVIOUSLY DISABLED LINES.
: * LNCTRA - THE STATE OF THE RX.ENABLE BIT MAY BE ALTERED.
: * THE CONTENTS OF THE IND.ADD.REG FIELD IN THE CSR ARE DESTROYED.
: *
: * CALLING SEQUENCE: JSR PC,RXENBL
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
```

```
RXENBL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
JSR
MOV R5,R0 ;COPY BIT MAP OF LINES TO ENABLE.
MOV #BIT0,R1 ;INITIALIZE THE SELECTED LINE BIT MASK.
MOV LNCTRA,R2 ;GET THE ADDRESS OF THE LNCTRL REGISTER.
MOV #NUMLNS,R3 ;GET MAXIMUM LINE NUMBER.
MOV IESTAT,R4 ;GET THE STATES OF THE INT ENABLE BITS.
CLR R5 ;CLEAR RX.ENABLE BIT LOG OF DISABLED LINES.
:
: SELECT EVERY LINE IN TURN,AND LOG ANY RX.ENABLE BIT THAT IS CLEAR.
:
2$: MOV R4,@CSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
BIT #BIT2,(R2) ;CHECK STATE OF RX.ENABLE BIT ON SELECTED LINE.
BNE 4$ ;SKIP NEXT INSTRUCTION IF RX.ENABLE SET.
BIS R1,R5 ;LOG RX ENABLE BIT CLEAR FOR SELECTED LINE.
:
: SFT RX.ENABLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE RX ENABLE
: LINE BIT MAP.
:
4$: BIT R1,R0 ;CHECK STATE OF RX.ENABLE LINE BIT MAP.
BEQ 6$ ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
BIS #BIT2,(R2) ;ENABLE RECEPTION ON SELECTED LINE.
6$: INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
DEC R3 ;DECREMENT LINE NUMBER.
BNE 2$ ;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 - LINE BIT MAP CORRESPONDING TO THE
; PREVIOUS LINES THAT WERE DISABLED.
RTS PC
```

022220
022220 004537 005232
022224 010500
022226 012701 000001
022232 013702 002212
022236 012703 000010
022242 013704 002234
022246 005005

022250 010477 157726
022254 032712 000004
022260 001001
022262 050105

022264 030100
022266 001402
022270 052712 000004
022274 005204
022276 006301
022300 005303
022302 001362

022304
022304 010566 000014
022310 004736

022312 000207

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 145
GLOBAL SUBROUTINE - RXIEO -

6058
6059
6060
6061
6062
6063
6064
6065
6066
6067
6068
6069
6070
6071
6072
6073
6074
6075
6076
6077
6078
6079
6080
6081
6082
6083
6084
6085
6086
6087
6088
6089

022314 010046
022316 104440
022316 104440
022320 010046
022322 012700 000340
022322 104441
022326 104441
022330 042737 137777 002234
022336 013777 002234 157636
022344 012600
022344 104441
022346 104441
022350 012600
022352 000207

```

.SBTTL GLOBAL SUBROUTINE - RXIEO -
+* *****:*****
* - RECEIVER INTERRUPT DISABLE -
* THIS ROUTINE IS USED TO DISABLE RECEIVER INTERRUPTS IN THE DHV11.
*
* INPUTS: NONE.
*
* OUTPUTS: THE RX.INT.ENBL BIT IS CLEARED IN THE DUT CSR.
* IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
* ENABLE BITS.
*
* CALLING SEQUENCE: JSR PC,RXIEO
*
* COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
* THE DUT CSR ARE DESTROYED.
*
* SUBORDINATE ROUTINES CALLED: NONE.
- - *****:*****
RXIEO:: MOV R0,-(SP) ;SAVE CONTENTS OF R0 ON THE STACK.
GETPRI -(SP) ;SAVE PROCESSOR PRIORITY ON STACK.
;
; TRAP C$GPRI
; MOV R0,-(SP)
; SETPRI #PRI07 ;IGNORE ANY INTERRUPT THAT MAY BE GENERATED.
; MOV #PRI07,R0
; TRAP C$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)+,R0
; TRAP C$SPRI
;
; MOV (SP)+,R0 ;RESTORE R0.
RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 146
GLOBAL SUBROUTINE - RXIE1 -

```

6090 .SBTTL GLOBAL SUBROUTINE - RXIE1 -
6091 :++ *****
6092 :* - RECEIVER INTERRUPT ENABLE -
6093 :* THIS ROUTINE IS USED TO ENABLE RECEIVER INTERRUPTS IN THE DHV11.
6094 :*
6095 :* INPUTS: NONE.
6096 :*
6097 :* OUTPUTS: THE RX.INT.ENBL BIT IS SET IN THE DUT CSR.
6098 :* IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
6099 :* ENABLE BITS.
6100 :*
6101 :* CALLING SEQUENCE: JSR PC,RXIE1
6102 :*
6103 :* COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
6104 :* THE DUT CSR ARE DESTROYED.
6105 :*
6106 :* SUBORDINATE ROUTINES CALLED: NONE.
6107 :-- *****
6108
6109 022354 052737 000100 002234 RXIE1:: BIS #BIT06,IESTAT ;SET RX.INT.ENBL BIT IN IESTAT.
6110 022362 042737 137677 002234 BIC #137677,IESTAT ;CLEAR ALL OTHER BITS, EXCEPT TX AND RX I.E.
6111 022370 013777 002234 157604 MOV IESTAT,@CSRA ;ENABLE RX INTERRUPTS.
6112 022376 000207 RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 147
GLOBAL SUBROUTINE - SAVBMP -

```

6113 .SBTTL GLOBAL SUBROUTINE - SAVBMP -
6114 :++ *****
6115 :* - SAVE BMP CODES ROUTINE -
6116 :* THIS ROUTINE SAVES THE PARAMETER PASSED IN, ONTO THE BMP CODE QUEUE
6117 :* TOGETHER WITH THE NUMBER OF THE CURRENTLY EXECUTING TEST.
6118 :*
6119 :* INPUTS: R2 - CONTAINS THE BMP CODE THAT IS TO BE PLACED ON THE QUEUE.
6120 :* BMPCQP - CONTAINS ADDRESS OF NEXT LOCATION IN THE BMP QUEUE.
6121 :* BMPCQB - LABEL AT BASE OF THE BMP CODE QUEUE.
6122 :* BMPCQE - LABEL OF NEXT LOCATION AFTER THE END OF THE BMP QUEUE.
6123 :* TSTNUM - CONTAINS THE NUMBER OF THE CURRENT TEST.
6124 :*
6125 :* OUTPUTS: BMPCQP - INCREMENTED BY 4.
6126 :* THE CONTENTS OF THE BMP CODE QUEUE ARE UPDATED.
6127 :*
6128 :* CALLING SEQUENCE: JSR PC,SAVBMP
6129 :*
6130 :* COMMENTS: IF THE OVERFLOW OCCURS THEN THE LAST LOCATION WILL BE
6131 :* OVERWRITTEN BY ANY SUBSEQUENT ATTEMPTS TO UPDATE THE QUEUE.
6132 :*
6133 :* SUBORDINATE ROUTINES CALLED: NONE.
6134 :-- *****
6135
6136 022400 SAVBMP:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
6137 022400 004537 005232 ;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
6138 022404 013704 002416 ;MOV BMPCQP,R4 ;GET THE POINTER TO THE NEXT LOCATION IN QUEUE.
6139 022410 113724 002224 ;MOV TSTNUM,(R4)+ ;SAVE THE CURRENT TEST NUMBER ON THE QUEUE.
6140 022414 005204 ;INC R4 ;INCREMENT THE POINTER TO GIVE AN EVEN ADDRESS.
6141 022416 042702 177400 ;BIC #177400,R2 ;CLEAR THE UNWANTED BITS FROM THE BMP CODE.
6142 022422 010224 ;MOV R2,(R4)+ ;SAVE THE BMP CODE ON THE QUEUE.
6143 022424 020427 002620 ;CMP R4,#BMPCQE ;CHECK IF OVERFLOW WILL OCCUR THE NEXT TIME.
6144 022430 103402 ;BLO 2$ ;GO SAVE THE POINTER IF WE WILL NOT OVERFLOW.
6145 022432 162704 000004 ;SUB #4,R4 ;RESET THE POINTER TO THE LAST LOCATION IN QUE.
6146 022436 010437 002416 2$: MOV R4,BMPCQP ;SAVE THE POINTER.
6147
6148 022442 60$: PASS ;RESTORE GPRS.
6149 022442 004736 ;PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
6150 022444 000207 RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 148
GLOBAL SUBROUTINE - SKPSTS -

6151			
6152			
6153			
6154			
6155			
6156			
6157			
6158			
6159			
6160			
6161			
6162			
6163			
6164			
6165			
6166			
6167			
6168			
6169			
6170			
6171	022446		
6172	022446	004537	005232
6173	022452	012704	000012
6174	022456	004737	015622
6175			
6176			
6177			
6178	022462	012701	000050
6179			
6180			
6181	022466	012703	052525
6182	022472	005301	
6183	022474	013704	002202
6184	022500	010124	
6185	022502	010324	
6186	022504	020437	002220
6187	022510	103774	
6188	022512	032701	000017
6189	022516	001365	
6190			
6191	022520		
6192	022520	004736	
6193	022522	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.
;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 DHV-11.
MOV #52525,R3 ;INITIALISE THE SKIP SELF-TEST CODE.
4$: DEC R1 ;SELECT THE NEXT SET OF DEVICE REGISTERS.
MOV CSRA,R4 ;GET THE ADDRESS OF THE CSR OF THE DUT.
MOV R1,(R4)+ ;SELECT A BANK OF DUT REGISTERS.
6$: MOV R3,(R4)+ ;WRITE THE CODE TO A DUT REGISTER.
CMP R4,TXBFCA ;COMPARE POINTER WITH LAST REGISTER ADDRESS.
BLO 6$ ;LOOP IF NOT ALL REGS DONE IN THIS BANK.
BIT #17,R1 ;TEST FOR IND.ADR.REG FIELD DECREMENTED TO 0.
BNE 4$ ;LOOP UNTIL ALL REGISTERS CONTAIN THE CODE.

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

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 149
GLOBAL SUBROUTINE - SPLSUP -

6194
6195
6196
6197
6198
6199
6200
6201
6202
6203
6204
6205
6206
6207
6208
6209
6210
6211
6212
6213
6214
6215
6216
6217
6218
6219
6220
6221
6222
6223
6224
6225
6226
6227
6228
6229
6230
6231
6232
6233
6234
6235
6236
6237
6238
6239
6240
6241
6242
6243
6244
6245
6246
6247
6248
6249

```

.SBTTL GLOBAL SUBROUTINE - SPLSUP -
*****
- SPLIT SPEED TRANSMISSION/RECEPTION SET-UP -
*****
THIS ROUTINE IS USED TO INITIALISE BOTH THE DUT AND THE
TRANSMISSION/RECEPTION CONTROL PARAMETERS TO THE CORRECT
STATE, PRIOR TO SPLIT SPEED TRANSMISSION/RECEPTION.
INPUTS:
R0 - TX,RX LPR CONTENTS FOR LINES IN GROUP II.
R1 - TX,RX LPR CONTENTS FOR LINES IN GROUP I.
R2 - START ADDRESS OF DATA PATTERN TO TRANSMIT.
R3 - NUMBER OF TIME DATA PATTERN TO BE TX ON LINES IN LINGRP1.
R4 - NUMBER OF TIME DATA PATTERN TO BE TX ON LINES IN LINGRP2.
ACTLNS - CONTAINS A BIT MAP OF ALL CURRENTLY ACTIVE LINES.
LGRP1M - CONTAINS THE BIT MAP OF LINE GROUP I LINES.
LOPBCK - CONTAINS THE TYPE OF LOOPBACK MODE SELECTED.
CBB - LABEL AT BASE OF TX/RX CONTROL BLOCK.
OUTPUTS:
THE CONTENTS OF THE CONTROL BLOCK ARE DESTROYED.
THE INDIRECT ADDRESS FIELD OF THE DUT CSR MAY BE DESTROYED.
THE DUT'S LPR'S AND LNC'S MAY BE MODIFIED.
THE FOLLOWING POINTERS AND COUNTERS ARE INITIALISED:
CHCNT,CHRTOT,DPEND,DPLEN,EXCNT,RXCNT,RXDONF,RXPTR,TXCNT,
TXDONF,TXPTR,TXRXL.
CALLING SEQUENCE: JSR PC,SPLSUP
COMMENTS: THIS ROUTINE SHOULD BE CALLED TWICE DURING THE TESTING OF
THE SPLIT SPEED CAPABILITIES OF THE DUT.
SO THAT BOTH LINE GROUPS ARE TESTED ON TRANSMISSION AND
RECEPTION.
EG, R1 - LPR CONTENTS FOR LINES IN LGRP2M,TX=Y,RX=Z BAUD.
R2 - LPR CONTENTS FOR LINES IN LGRP1M,TX=Z,RX=Y BAUD.
R3 - REPEAT TX ON LINES IN LINE GROUP 1 = X TIMES.
R4 - REPEAT TX ON LINES IN LINE GROUP 2 = W TIMES.
JSR PC,SPLSUP ; DO SET-UP.
EXECUTE TEST FOR THE ABOVE SET-UP.
SWAP THE CONTENTS OF R1 AND R2.
SWAP THE CONTENTS OF R3 AND R4.
R1 - LPR CONTENTS FOR LINES IN LGRP2M,TX=Z,RX=Y BAUD.
R2 - LPR CONTENTS FOR LINES IN LGRP1M,TX=Y,RX=Z BAUD.
R3 - REPEAT TX ON LINES IN LINE GROUP 1 = W TIMES.
R4 - REPEAT TX ON LINES IN LINE GROUP 2 = X TIMES.
JSR PC,SPLSUP ;DO SET UP AGAIN.
EXECUTE TEST AGAIN.
SUBORDINATE ROUTINES CALLED: CONMAP,RXDSBL,RXENBL,TXRINI.
*****
SPLSUP:: SAVE ;SAVE CONTENTS OF THE GPR'S R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
JCR
MOV R0,70$ ;SAVE LPR PARAMETER FOR LINE GRP2.
MOV R1,72$ ;SAVE LPR PARAMETER FOR LINE GRP1.
CLR TXDONF ;CLEAR THE TX DONE FLAGS FOR ALL LINES.
CLR RXDONF ;CLEAR THE RX DONE FLAGS FOR ALL LINES.
;+

```

022524		
022524	004537	005232
022530	010037	023020
022534	010137	023022
022540	005037	002410
022544	005037	002412

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 150
GLOBAL SUBROUTINE - SPLSUP -

```

6250      : SET UP THE TRANSMISSION/RECEPTION CONTROL BLOCK TO INITIALISE THE LINES
6251      : IN GROUP II.
6252      :
6253 022550 010037 003030      MOV      R0,CBB      ;SET CONTENTS OF LPR PARAMS IN TX/RX C.BLK.
6254 022554 012700 003032      MOV      #CBB+2,R0  ;GET BASE ADDRESS OF CONTROL BLOCK.
6255 022560 012720 000004      MOV      #4,(R0)+   ;LNCTRL PARAMETER, ENABLE RECEIVERS.
6256 022564 010220             MOV      R2,(R0)+   ;START ADDRESS OF DATA PATTERN.
6257 022566 012720 000020      MOV      #16,(R0)+ ;DATA PATTERN LENGTH SET TO 16.
6258 022572 010420             MOV      R4,(R0)+   ;NUMBER OF DATA PATTNS TO TRANSMIT ON LINGRP2.
6259 022574 013710 002174      MOV      ACTLNS,(R0);BIT MAP OF LINES TO INITIALISE.
6260 022600 043720 002230      BIC      LGRP1M,(R0)+;CLEAR THE UNWANTED LINES FROM BIT MAP.
6261 022604 113720 002176      MOV      LOPBCK,(R0)+;SET LOOPBACK MODE.
6262 022610 005200             INC      R0          ;INCREMENT ADDRESS TO ACCESS NEXT WORD.
6263 022612 012710 000002      MOV      #2,(R0)   ;SET OFFSET FOR EACH TRANSMISSION START TO 2.
6264
6265      :
6266      : INITIALISE THE DUT AND THE ASSOCIATED POINTERS AND COUNTERS, TO THE STATE
6267      : DICTATED BY THE CONTENTS OF THE TX/RX CONTROL BLOCK.
6268 022616 004737 023666      JSR      PC,TXRINI  ;INITIALISE DUT.
6269
6270      :
6271      : SET UP CONTROL BLOCK FOR LINES IN GROUP I.
6272 022622 012700 003030      MOV      #CBB,R0   ;GET START ADDRESS OF CONTROL BLOCK.
6273 022626 010120             MOV      R1,(R0)+   ;SET LPR PARAMETER FOR LINES TO RECEIVE DATA.
6274 022630 012720 000004      MOV      #4,(R0)+   ;LNCTRL PARAMETER, ENABLE RECEIVERS.
6275 022634 010220             MOV      R2,(R0)+   ;START ADDRESS OF DATA PATTERN.
6276 022636 012720 000020      MOV      #16,(R0)+ ;DATA PATTERN LENGTH SET TO 16.
6277 022642 010320             MOV      R3,(R0)+   ;NUMBER OF DATA PATTNS TO TRANSMIT ON LINGRP1.
6278 022644 013710 002174      MOV      ACTLNS,(R0);BIT MAP OF LINES TO INITIALISE.
6279 022650 043720 002232      BIC      LGRP2M,(R0)+;CLEAR THE UNWANTED LINES FROM BIT MAP.
6280 022654 113720 002176      MOV      LOPBCK,(R0)+;SET LOOPBACK MODE.
6281 022660 005200             INC      R0          ;INCREMENT ADDRESS TO ACCESS NEXT WORD.
6282 022662 012710 000002      MOV      #2,(R0)   ;SET OFFSET FOR EACH TRANSMISSION START TO 2.
6283
6284      :
6285      : INITIALISE THE DUT AND THE ASSOCIATED POINTERS AND COUNTERS, TO THE STATE
6286      : DICTATED BY THE CONTENTS OF THE TX/RX CONTROL BLOCK.
6287 022666 004737 023666      JSR      PC,TXRINI  ;INITIALISE DUT.
6288
6289      :
6290      : SET-UP THE REQUIRED LPR PARAMETERS NEEDED FOR THE CORRECT RECEPTION OF DATA
6291      : ON ASSOCIATED IN-ACTIVE LINES.
6292
6293      :
6294      : INITIALISE LPR PARAMETERS FOR LINE GROUP 1.
6295
6296 022672 012701 000377      MOV      #MAPLNS,R1 ;SET UP BIT MAP CORRESPONDING TO ALL LINES.
6297 022676 013702 002174      MOV      ACTLNS,R2  ;GET THE ACTIVE (TX) LINE BIT MAP.
6298 022702 005101             COM      R1          ;GENERATE A BIT MAP OF NONE EXISTANT LINES.
6299 022704 005102             COM      R2          ;GENERATE A BIT MAP OF INACTIVE LINES.
6300 022706 040102             BIC      R1,R2       ;CLEAR ANY 'NONE EXISTANT' INACTIVE LINES.
6301 022710 043702 002232      BIC      LGRP2M,R2  ;ONLY PASS LGRP1 ASSOCIATED LINE BIT MAP.
6302 022714 010237 003042      MOV      R2,CBMAPA  ;SET UP BIT MAP IN CONTROL BLOCK.
6303 022720 005037 003040      CLR      CBDPNA     ;CLEAR REPEAT TX COUNT IN CONTROL BLOCK.
6304 022724 013737 023022 003030 MOV      72$,CBLPRA  ;SET-UP COMPLEMENTARY LPR PARM FOR LGRP2.
6305 022732 004737 023666      JSR      PC,TXRINI  ;INITIALISE INACTIVE LINES IN LGRP2.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12 JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 151
GLOBAL SUBROUTINE - SPLSUP -

```

6306
6307      :+ INITIALISE LPR PARAMETERS FOR LINE GROUP 2.
6308      :-
6309      022736 013702 002174      MOV    ACTLNS,R2      ;GET THE ACTIVE (TX) LINE BIT MAP.
6310      022742 005102              COM    R2              ;GENERATE A BIT MAP OF INACTIVE LINES.
6311      022744 040102              BIC    R1,R2          ;CLEAR ANY NONE EXISTANT INACTIVE LINES.
6312      022746 043702 002230      BIC    LGRP1M,R2     ;ONLY PASS LGRP2 ASSOCIATED LINE BIT MAP.
6313      022752 010237 003042      MOV    R2,CBMAPA     ;SET-UP BIT MAP IN CONTROL BLOCK.
6314      022756 013737 023020 003030  MOV    70$,CBLPRA    ;SET-UP COMPLAMETARY LPR PARAM FOR LGRP1.
6315      022764 004737 023666      JSR    PC,TXRINI     ;INITIALISE INACTIVE LINES IN LGRP1.
6316
6317      :+ DISABLE RECEIVERS ON ALL LINES TO ENSURE THAT ONLY THE RECEIVERS OF THE
6318      : ASSOCIATED ACTIVE (TX) LINES ARE ENABLED.(STAGGARED LOOPBACK)
6319      : RE-ENABLE RECEPTION ON THE CORRECT ASSOCIATED LINES.
6320      :-
6321      022770 012705 000377      MOV    #MAPLNS,R5    ;SET-UP BIT MAP FOR ALL LINES.
6322      022774 004737 022124      JSR    PC,RXDSBL     ;DISABLE RX ON ALL LINES.
6323
6324      :+ ENABLE RECEIVERS ON ASSOCIATED (RX) LINES.
6325      :-
6326      023000 013705 002174      MOV    ACTLNS,R5    ;GET ACTIVE (TX) LINE BIT MAP.
6327      023004 004737 015546      JSR    PC,CONMAP     ;GENERATE AN ASSOCIATED (RX) LINE BIT MAP.
6328      023010 004737 022220      JSR    PC,RXENBL     ;ENABLE RECEIVERS ON ASSOCIATED LINES.
6329
6330      023014          60$: PASS          ;RESTORE GRP'S.
6331      023014 004736          JSR    PC,@(SP)+      ;RETURN TO PREG05 SUBRT.
6332      023016 000207
6333      023020 000000      70$: .WORD 0          ;LOCAL STORAGE OF LPR PARAMETER LGRP2.
6334      023022 000000      72$: .WORD 0          ;LOCAL STORAGE OF LPR PARAMETER LGRP1.

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 152
GLOBAL SUBROUTINE - SWAPO -

6335
6336
6337
6338
6339
6340
6341
6342
6343
6344
6345
6346
6347
6348
6349
6350
6351
6352
6353
6354
6355
6356
6357
6358
6359
6360
6361
6362
6363
6364
6365
6366
6367
6368
6369
6370
6371
6372
6373
6374
6375
6376
6377
6378
6379
6380
6381
6382
6383
6384
6385
6386

023024 010046

023026 010146
023030 010246
023032 010346
023034 010446
023036 010546

023040 012700 002372
023044 012001
023046 012002
023050 012003
023052 012004
023054 012005

023056 012640
023060 012640
023062 012640
023064 012640
023066 012640

023070 012600
023072 000207

```
.SBTTL GLOBAL SUBROUTINE - SWAPO -
+*****
+ - SWAP GPRS WITH GPR SET 0 ROUTINE -
+ THIS SUBROUTINE SWAPS THE PRESENT CONTENTS OF GPRS R1 THRU R5 WITH
+ THE CONTENTS OF THE NUMBER ZERO GPR SAVE AREA. THE CONTENTS OF R0
+ ARE NOT ALTERED BY THIS SUBROUTINE.
+
+ INPUTS: GPR CONTENTS R1 THRU R5.
+ GPRS0B - LABEL AT BASE OF GPR SAVE AREA NUMBER ZERO.
+
+ OUTPUTS: R1 THRU R5 CONTAIN THE PREVIOUS CONTENTS OF GPR SAVE AREA
+ ZERO WORDS 1 THRU 5 RESPECTIVELY.
+ GPRS0 - GPR SAVE AREA 0 WORDS 1 THRU 5, CONTAIN PREVIOUS
+ CONTENTS OF GPRS R1 THRU R5 RESPECTIVELY.
+
+ CALLING SEQUENCE: JSR PC,SWAPO
+
+ COMMENTS: THE STATE OF THE CARRY FLAG IS NOT ALTERED BY THIS ROUTINE.
+
+ SUBORDINATE ROUTINES CALLED: NONE.
+-----
SWAPO:: MOV R0,-(SP) ;SAVE THE CONTENTS OF R0.
+
+ LOAD THE STACK FROM THE GPRS.
+-----
MOV R1,-(SP) ;SAVE THE CONTENTS OF R1.
MOV R2,-(SP) ;SAVE THE CONTENTS OF R2.
MOV R3,-(SP) ;SAVE THE CONTENTS OF R3.
MOV R4,-(SP) ;SAVE THE CONTENTS OF R4.
MOV R5,-(SP) ;SAVE THE CONTENTS OF R5.
+
+ LOAD THE GPRS FROM THE GPR SAVE AREA 0.
+-----
MOV #GPRS0B,R0 ;GET THE BASE ADDRESS OF GPR SAVE AREA 0.
MOV (R0)+,R1 ;LOAD R1 WITH GPR SAVE AREA 0 WORD 1.
MOV (R0)+,R2 ;LOAD R1 WITH GPR SAVE AREA 0 WORD 2.
MOV (R0)+,R3 ;LOAD R1 WITH GPR SAVE AREA 0 WORD 3.
MOV (R0)+,R4 ;LOAD R1 WITH GPR SAVE AREA 0 WORD 4.
MOV (R0)+,R5 ;LOAD R1 WITH GPR SAVE AREA 0 WORD 5.
+
+ LOAD THE GPR SAVE AREA 0 FROM THE STACK.
+-----
MOV (SP)+,-(R0) ;LOAD GPR SAVE AREA 0 WORD 5 WITH SAVED R5.
MOV (SP)+,-(R0) ;LOAD GPR SAVE AREA 0 WORD 4 WITH SAVED R4.
MOV (SP)+,-(R0) ;LOAD GPR SAVE AREA 0 WORD 3 WITH SAVED R3.
MOV (SP)+,-(R0) ;LOAD GPR SAVE AREA 0 WORD 2 WITH SAVED R2.
MOV (SP)+,-(R0) ;LOAD GPR SAVE AREA 0 WORD 1 WITH SAVED R1.
MOV (SP)+,R0 ;RESTORE THE INITIAL VALUE OF R0.
RTS PC
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 153
GLOBAL SUBROUTINE - TSABRT -

6387
6388
6389
6390
6391
6392
6393
6394
6395
6396
6397
6398
6399
6400
6401
6402
6403
6404
6405
6406
6407
6408
6409
6410
6411
6412
6413
6414
6415
6416
6417
6418
6419
6420
6421
6422
6423
6424
6425
6426
6427

023074
023074 004537 005232
023100 012701 023116
023104 012737 012254 005230
023112
023112 104460
023114 000432
023116 047040 047117 051055
023124 046105 052101 042105
023132 052040 051505 020124
023140 051105 047522 020122
023146 047506 047125 020104
023154 052504 044522 043516
023162 052040 051505 020124
023170 054105 041505 052125
023176 047511 000116
023202
023202 004736
023204 000207

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

TSABRT:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV #2$,R1 ;PASS ADDRESS OF FIRST MESSAGE TO BE REPORTED.
MOV #ER1603,ERRBLK ;SET-UP THE ERROR REPORTING ROUTINE.
ERROR ; >>>> ERROR <<<<.
; TRAP C$ERROR
BR 60$ ;
2$: .ASCIZ / NON-RELATED TEST ERROR FOUND DURING TEST EXECUTION/

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

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 154
GLOBAL SUBROUTINE - TXDONE -

```

6428 .SBTTL GLOBAL SUBROUTINE - TXDONE -
6429 :+ *****
6430 :* - TRANSMISSION DONE -
6431 :* THIS SUBROUTINE IS USED IN THE TRANSMISSION/RECEPTION TESTS TO ALLOW
6432 :* TIME FOR TRANSMISSION TO COMPLETE ON OUTSTANDING LINES.
6433 :*
6434 :* INPUTS: ACTLNS - CONTAINS BIT MAP OF ALL ACTIVE LINES.
6435 :* TXDNF - TX DONE FLAGS, SET FOR LINES THAT HAVE SENT ALL CHARS.
6436 :* CHCNT - TABLE CONTAINING THE NUMBER OF CHARS TO BE TX'D.
6437 :*
6438 :*
6439 :* OUTPUTS: TRANSMISSION INTERRUPTS ARE DISABLED.
6440 :*
6441 :* CALLING SEQUENCE: JSR PC, TXDONE
6442 :*
6443 :* COMMENTS:
6444 :*
6445 :* SUBORDINATE ROUTINES CALLED: MSLOOP, MUL16U.
6446 :-- *****
6447
6448 023206 TXDONE:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
6449 023206 004537 005232 JSR R5, PREG05 ;CALL REGISTER SAVE SUBRT.
6450
6451 :+ CHECK IF ALL ACTIVE LINES HAVE COMPLETED TRANSMISSION.
6452 :+ IF ANY HAVE NOT YET COMPLETED, DETERMINE THE TX CHAR COUNT FOR A
6453 :+ LINE THAT HAS OUTSTANDING CHARACTERS TO TRANSMIT. USING THIS VALUE,
6454 :+ CALCULATE THE TIME-OUT VALUE NEEDED AT THE CURRENTLY SELECTED BAUD RATE.
6455 :--
6456 023212 013703 002174 MOV ACTLNS, R3 ;GET THE ACTIVE LINE BIT MAP.
6457 023216 013702 002410 MOV TXDNF, R2 ;GET THE BIT MAP OF LINES THAT HAVE COMPLETED.
6458 023222 040203 BIC R2, R3 ;GENERATE A BIT MAP OF LINES THAT ARE STILL TX.
6459 023224 005703 TST R3 ;CHECK IF ALL LINES HAVE COMPLETED TX.
6460 023226 001427 BEQ 6$ ;GO DISABLE TX INTERRUPTS IF ALL DONE.
6461
6462 :+ FIND A LINE THAT HAS NOT COMPLETED TRANSMISSION.
6463 :+ OBTAIN THE EXPECTED CHARACTER COUNT FOR THAT LINE (WHICH IS THE SAME FOR
6464 :+ ALL OTHER LINES WITH OUTSTANDING TX'S).
6465 :+ CALCULATE TIME-OUT VALUE.
6466 :--
6467 023230 005004 CLR R4 ;CLEAR LINE NUMBER COUNTER.
6468 023232 012702 000001 MOV #1, R2 ;SELECT BIT MAP FOR THE FIRST LINE.
6469 023236 030203 2$: BIT R2, R3 ;SEE IF THIS LINE HAS COMPLETED.
6470 023240 001003 BNE 4$ ;BRANCH IF THIS LINE HAS NOT COMPLETED TX.
6471 023242 006102 ROL R2 ;SHIFT THE LINE BIT MAP FOR THE NEXT LINE.
6472 023244 005204 INC R4 ;INCREMENT THE LINE NUMBER COUNTER.
6473 023246 000773 BR 2$ ;LOOP TO CHECK THE NEXT LINE.
6474 023250 006304 4$: ASL R4 ;LINE NUMBER X 2 TO OBTAIN OFFSET INTO TABLE.
6475 023252 016401 003350 MOV CHCNT(R4), R1 ;GET THE EXPECTED NUMBER OF CHARS TO BE TX'D.
6476 023256 013702 002242 MOV RXTOUT, R2 ;GET THE CURRENT TIME-OUT VALUE FOR ONE CHAR.
6477 023262 004737 017470 JSR PC, MUL16U ;(NUMBER OF CHARS TO TX) X (TIME-OUT OF 1 CHAR)
6478 023266 006301 ASL R1 ;MULTIPLY DELAY TIME BY 2 TO GIVE A SAFE VALUE.
6479
6480 :+ WAIT FOR ALL OUSTANDING TRANSMISSIONS TO COMPLETE OR TIME-OUT.
6481 :+ DISABLE ALL TRANSMISSION INTERRUPTS.
6482 :--
6483 023270 013702 002174 MOV ACTLNS, R2 ;PASS A BIT MAP OF THE BITS TO TEST.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 155
GLOBAL SUBROUTINE

- TXDONE -

6484	023274	010203		MOV	R2,R3			:PASS THE EXPECTED STATE OF THE TXDONF.
6485	023276	012704	002410	MOV	#TXDONF,R4			:PASS THE ADDRESS OF THE WORD TO TEST.
6486	023302	004737	017454	JSR	PC,MSLOOP			:WAIT FOR TIME-OUT OF TX COMPLETION.
6487	023306	004737	023602	6\$: JSR	PC,TXIE0			:DISABLE ALL TX INTERRUPTS.
6488								
6489	023312			60\$: PASS				:RESTORE GPRS.
6490	023312	004736				JSR		PC,@(SP)+
6491	023314	000207		RTS	PC			:RETURN TO PREG05 SUBRT.

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 156
GLOBAL SUBROUTINE - TXDONE -

6492
6493
6494
6495
6496
6497
6498
6499
6500
6501
6502
6503
6504
6505
6506
6507
6508
6509
6510
6511
6512
6513
6514
6515
6516
6517
6518
6519
6520
6521
6522
6523
6524
6525
6526
6527
6528
6529
6530
6531
6532
6533
6534
6535
6536
6537
6538
6539
6540
6541
6542
6543
6544
6545
6546
6547

023316
023316 004537 005232
023322 010500
023324 012701 000001
023330 013702 002216
023334 005202
023336 012703 000010
023342 013704 002234
023346 005005

023350 010477 156626
023354 105712
023356 100001
023360 050105

023362 030100
023364 001402
023366 142712 000200
023372 005204
023374 006301
023376 005303
023400 001363

023402
023402 010566 000014
023406 004736

```
.SBTTL GLOBAL SUBROUTINE - TXDSBL -
:++ *****
:      - TRANSMITTER DISABLE -
:      THIS SUBROUTINE IS USED TO DISABLE TRANSMISSION ON SELECTED LINES BY,
:      CLEARING THE ASSOCIATED TX.ENABLE BIT ON THE DUT.
:
: INPUTS:      R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO CLEAR TX.ENABLE.
:              CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
:              IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
:              NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
:              TXAD2A - CONTAINS THE ADDRESS OF THE TBUFFAD2 REGISTER.
:
: OUTPUTS:     R5 - BIT'S SET INDICATE THE INITIAL STATES OF ALL TX.ENABLE BITS.
:              TBUFFAD2 - 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 TBUFFAD2 REGISTER.
:INC R2 ;GET THE ADDRESS OF THE MSBYTE OF TBUFFAD2 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.
:
:2$: MOV R4,@CSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
: TSTB (R2) ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
: BPL 4$ ;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.
:
:4$: BIT R1,R0 ;CHECK STATE OF DISABLE LINE BIT MAP.
: BEQ 6$ ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
: BICB #BIT7,(R2) ;CLEAR TX.ENABLE BIT ON SELECTED LINE.
:6$: INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
: ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
: DEC R3 ;DECREMENT LINE NUMBER.
: BNE 2$ ;LOOP TO CHECK NEXT LINE.
:
:60$: PASS R5
:MOV R5,R5SLOT(SP) ;PUT R5 IN STACK SLOT.
: JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
: ;R5 - PREVIOUS STATES OF ALL TX.ENABLE BITS.
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 157
GLOBAL SUBROUTINE - TXDSBL -

6548 023410 000207

RTS PC

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 158
GLOBAL SUBROUTINE

- TXENBL -

6549
6550
6551
6552
6553
6554
6555
6556
6557
6558
6559
6560
6561
6562
6563
6564
6565
6566
6567
6568
6569
6570
6571
6572 023412
6573 023412 004537 005232
6574 023416 010500
6575 023420 012701 000001
6576 023424 013702 002216
6577 023430 005202
6578 023432 012703 000010
6579 023436 013704 002234
6580 023442 005005
6581
6582
6583
6584 023444 010477 156532
6585 023450 105712
6586 023452 100401
6587 023454 050105
6588
6589
6590
6591
6592 023456 030100
6593 023460 001402
6594 023462 152712 000200
6595 023466 005204
6596 023470 006301
6597 023472 005303
6598 023474 001363
6599
6600 023476
6601 023476 010566 000014
6602 023502 004736
6603
6604

```

.SBTTL GLOBAL SUBROUTINE - TXENBL -
** *****
* - TRANSMITTER ENABLE -
* THIS SUBROUTINE IS USED TO ENABLE TRANSMISSION ON SELECTED LINES BY
* SETTING THE ASSOCIATED TX.ENABLE BIT ON THE DUT.
* INPUTS: R5 - BIT'S SET CORRESPOND TO LINES ON WHICH TO SET TX.ENABLE.
* CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
* IESTAT - CONTAINS THE STATE OF TXIE AND RXIE BITS IN THE CSR.
* NUMLNS - EQUATED TO BE THE MAXIMUM NUMBER OF LINES AVAILABLE.
* TXAD2A - CONTAINS THE ADDRESS OF THE TBUFAD2 REGISTER.
* OUTPUTS: R5 - BIT'S SET INDICATE PREVIOUSLY DISABLED LINES.
* 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,TXENBL
* COMMENTS:
* SUBORDINATE ROUTINES CALLED: NONE.
-- *****
TXENBL:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
JSR
MOV R5,R0 ;COPY BIT MAP OF LINES TO ENABLE.
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.
MOV IESTAT,R4 ;GET THE STATES OF THE INT ENABLE BITS.
CLR R5 ;CLEAR TX.ENABLE BIT LOG OF DISABLED LINES.
;+
; SELECT EVERY LINE IN TURN,AND LOG ANY TX.ENABLE BIT THAT IS CLEAR.
2$: MOV R4,@CSRA ;WRITE TO DUT CSR TO SELECT LINE REGISTERS.
TSTB (R2) ;CHECK STATE OF TX.ENABLE BIT ON SELECTED LINE.
BMI 4$ ;SKIP NEXT INSTRUCTION IF TX.ENABLE SET.
BIS R1,R5 ;LOG TX ENABLE BIT CLEAR FOR SELECTED LINE.
;+
; SET TX.ENABLE ON LINES THAT HAVE A CORRESPONDING BIT SET IN THE TX ENABLE
; LINE BIT MAP.
4$: BIT R1,R0 ;CHECK STATE OF TX.ENABLE LINE BIT MAP.
BEQ 6$ ;BRANCH IF THIS LINE TO REMAIN UNALTERED.
BISB #BIT7,(R2) ;ENABLE TRANSMISSION ON SELECTED LINE.
6$: INC R4 ;PREPARE TO SELECT REGISTERS FOR NEXT LINE.
ASL R1 ;SHIFT BIT MAP FOR NEXT LINE.
DEC R3 ;DECREMENT LINE NUMBER.
BNE 2$ ;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 - LINE BIT MAP CORRESPONDING TO THE
; PREVIOUS LINES THAT WERE DISABLED.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83
GLOBAL SUBROUTINE

13:56 PAGE 159
- TXENBL -

6605 023504 000207

RTS PC

C
C

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 160
GLOBAL SUBROUTINE - TXFRPR -

6606
6607
6608
6609
6610
6611
6612
6613
6614
6615
6616
6617
6618
6619
6620
6621
6622
6623
6624
6625
6626
6627
6628
6629
6630
6631
6632
6633
6634
6635
6636
6637
6638
6639
6640
6641
6642
6643
6644
6645
6646
6647
6648
6649
6650
6651
6652
6653
6654
6655
6656
6657
6658
6659
6660
6661

```

.SBTTL GLOBAL SUBROUTINE - TXFRPR -
:++ *****
: * - TRANSMIT FRAMMING ERROR DATA ROUTINE -
: * THIS ROUTINE IS USED TO INITIATE DMA MODE TRANSMISSION
: * IN THE FRAMMING ERROR TEST. IT SENDS A SINGLE CHARACTER DMA BUFFER ON
: * EACH ACTIVE LINE IN THE BIT MAP, TO CAUSE FUTURE TX INTERRUPTS WHICH
: * WILL CONTINUE THE TRANSMISSION IF MORE THAN ONE BUFFER IS TO BE SENT.
: *
: * INPUTS: R4 - CONTAINS THE LINES ON WHICH TX IS TO TAKE PLACE.
: * ACTLNS - ACTIVE LINES BIT MAP.
: * BITTBL - LABEL OF TABLE OF WORDS EACH WITH A BIT SET.
: * CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
: * DPENDB - BASE OF THE DATA PATTERN END TABLE (ENTRY PER LINE).
: * DPLENB - BASE OF THE DATA PATTERN LENGTH TABLE.
: * IESTAT - PRESERVED STATES OF THE DUT INTERRUPT ENABLE BITS.
: * NUMLNS - EQUATED TO NUMBER OF LINES ON A DUT.
: * TXCNTB - LABEL AT BASE OF THE TX CHARACTER COUNTER TABLE.
: * TXPTRB - LABEL AT BASE OF THE TX DATA PATTERN POINTERS TABLE.
: *
: * OUTPUTS: CSR - DUT CSR IND.ADR.REG FIELD IS DESTROYED.
: * TXCNTX - COUNTERS INCREMENTED FOR LINES ON WHICH CHARS SENT.
: * TXINTF - TX INT FLAGS (BIT SET IF DMA.HO FOUND SET ON LINE).
: *
: * CALLING SEQUENCE: JSR PC,TXFRPR
: *
: * COMMENTS: THIS ROUTINE ASSUMES THAT AT LEAST ONE DATA PATTERN SHOULD BE
: * TRANSMITTED ON EACH ACTIVE LINE.
: * INTERRUPTS MUST BE DISABLED WHEN CALLING THIS ROUTINE.
: *
: * SUBORDINATE ROUTINES CALLED: DODMA.
:-- *****
TXFRPR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
MOV ACTLNS,R5 ;GET THE ACTIVE LINE BIT MAP.
COM R4 ;GET BIT MAP OF LINES THAT WILL RECEIVE DATA.
BIC R4,R5 ;CLEAR LINES THAT WILL RX FROM TX LINE BIT MAP.
: *
: * SET UP LOOP WHICH HANDLES ONE LINE PER ITERATION.
: *
: * CLR R1 ;CLEAR THE LINE NUMBER COUNTER.
: *
: * IF THE LINE IS INACTIVE SKIP TO SELECT THE NEXT LINE.
: *
2$: CLC ;CLEAR BOOLEAN REGISTER.
ASR R5 ;SHIFT BIT MAP OF LINES TO TX ON INTO BOOL.REG.
BCC 6$ ;DON'T TX ON THIS LINE IF IT IS NOT ACTIVE.
: *
: * LINE IS ACTIVE.
: * INITIATE DMA ON THIS LINE.
: * GET THE DATA PATTERN LENGTH FOR THIS LINE.
: *
MOV R1,R4 ;COPY LINE NUMBER.
ASL R4 ;CALCULATE WORD OFFSET FOR THIS LINE.
MOV DPLENB(R4),R3 ;GET DATA PATTERN LENGTH FOR THIS LINE.
MOV TXPTRB(R4),R2 ;PREPARE TO PASS DATA PATTERN ADR TO DODMA RTN.
: *
:++

```

C
C

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 161
GLOBAL SUBROUTINE - TXFRPR -

```

6662          : WRITE DMA PARAMETERS TO THE DUT.
6663          :-
6664 023546 004737 015662      JSR   PC,DODMA
6665 023552 103404              BCS   4$           ;SKIP ERROR IF DODMA WAS SUCCESSFUL.
6666          :+
6667          : SET THE PROPER BIT OF THE TX INTERRUPT FLAGS TO INDICATE THE LINE ERROR.
6668          :-
6669 023554 056437 002332 002252  BIS   BITTBL(R4),TXINTF      ;INDICATE THE ERROR.
6670 023562 000402              BR    6$           ;SKIP UPDATING POINTERS AND COUNTERS.
6671          :+
6672          : UPDATE THE TX CHARACTER COUNT FOR THIS LINE.
6673          :-
6674 023564 060364 003410      4$:   ADD   R3,TXCNTB(R4)  ;ADD THE DATA PATTERN LENGTH TO TX CHAR COUNT.
6675          :+
6676          : INCREMENT LINE COUNTER,GOTO NEXT LINE IF NOT DONE.
6677          :-
6678 023570 005201              6$:   INC   R1           ;INCREMENT THE LINE COUNTER.
6679 023572 005705              TST   R5           ;TEST THE TX LINE BIT MAP.
6680 023574 001353              BNE   2$           ;LOOP TO SEND CHAR TO ANOTHER LINE IF NOT DONE.
6681          :+
6682 023576              60$:  PASS                ;RESTORE GPRS.
6683 023576 004736              RTS   PC           JSR   PC,@(SP)+      ;RETURN TO PREG05 SUBRT.
6684 023600 000207

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 162
GLOBAL SUBROUTINE - TXIE0 -

6685
6686
6687
6688
6689
6690
6691
6692
6693
6694
6695
6696
6697
6698
6699
6700
6701
6702
6703
6704
6705
6706
6707
6708
6709
6710
6711
6712
6713
6714
6715
6716

023602 010046
023604
023604 104440
023606 010046
023610
023610 012700 000340
023614 104441
023616 042737 177677 002234
023624 013777 002234 156350
023632
023632 012670
023634 104441
023636 012600
023640 000207

```

.SBTTL GLOBAL SUBROUTINE - TXIE0 -
:++ *****
: * - TRANSMITTER INTERRUPT DISABLE -
: * THIS ROUTINE IS USED TO DISABLE TRANSMITTER INTERRUPTS IN THE DHV11.
: *
: * INPUTS: NONE.
: *
: * OUTPUTS: THE TX.INT.ENBL BIT IS CLEARED IN THE DUT CSR.
: * IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
: * ENABLE BITS.
: *
: * CALLING SEQUENCE: JSR PC,TXIE0
: *
: * COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
: * THE DUT CSR ARE DESTROYED.
: *
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
TXIE0:: MOV RO,-(SP) ;SAVE CONTENTS OF RO ON THE STACK.
GETPRI -(SP) ;SAVE CURRENT PROCESSOR PRIORITY ON THE STACK.
TRAP C$GPRI
MOV RO,-(SP)
SETPRI #PRI07 ;IGNORE ANY INTERRUPTS THAT MAY BE GENERATED.
MOV #PRI07,RO
TRAP C$SPRI
BIC #177677,IESTAT ;CLEAR TX.INT.ENBL BIT IN IESTAT.
MOV IESTAT,@CSRA ;DISABLE TX INTERRUPTS.
SETPRI (SP)+ ;ENABLE INTERRUPTS TO THE PROCESSOR AGAIN.
MOV (SP)+,RO
TRAP C$SPRI
MOV (SP)+,RO ;RESTORE RO.
RTS PC

```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 163
GLOBAL SUBROUTINE - TXIE1 -

6717
6718
6719
6720
6721
6722
6723
6724
6725
6726
6727
6728
6729
6730
6731
6732
6733
6734
6735
6736
6737
6738
6739

```
.SBTTL GLOBAL SUBROUTINE - TXIE1 -
: * *****
: * - TRANSMITTER INTERRUPT ENABLE -
: * THIS ROUTINE IS USED TO ENABLE TRANSMITTER INTERRUPTS IN THE DHV11.
: *
: * INPUTS: NONE.
: *
: * OUTPUTS: THE TX.INT.ENBL BIT IS SET IN THE DUT CSR.
: * IESTST -CONTAINS THE UPDATED STATUS OF THE TX AND RX INTERRUPT
: * ENABLE BITS.
: *
: * CALLING SEQUENCE: JSR PC,TXIE1
: *
: * COMMENTS: THE CONTENTS OF THE INDIRECT ADDRESS REGISTER FIELD IN
: * THE DUT CSR ARE DESTROYED.
: *
: * SUBORDINATE ROUTINES CALLED: NONE.
: * - *****
```

```
023642 052737 040000 002234 TXIE1:: BIS #BIT14,IESTAT ;SET TX.INT.ENBL BIT IN IESTAT.
023650 042737 137677 002234 BIC #137677,IESTAT ;CLEAR ALL BITS EXCEPT TX RX I.E BITS.
023656 013777 002234 156316 MOV IESTAT,@CSRA ;DISABLE TX INTERRUPTS.
023654 000207 RTS PC
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 164
GLOBAL SUBROUTINE - TXRINI -

```

6740 .SBTTL GLOBAL SUBROUTINE - TXRINI -
6741 *+ *****
6742 * - TRANSMIT AND RECEIVE INITIALIZATION ROUTINE -
6743 * THIS SUBROUTINE PERFORMS THE INITIALIZATION OF THE VARIOUS POINTERS,
6744 * COUNTERS, AND FLAGS WHICH ARE USED DURING THE TRANSMISSION AND
6745 * RECEPTION PORTION OF A TEST. THIS INITIALIZATION IS PERFORMED ON
6746 * THE SPECIFIED LINES ONLY, OTHER LINE VARIABLES REMAIN UNCHANGED.
6747 *
6748 * INPUTS:
6749 * CHCNTB - LABEL AT BASE OF LINE CHARACTER COUNT TABLE.
6750 * CHRTOT - MAX # OF CHARS TO RX ON LINES ALREADY INITIALIZED.
6751 * DPENDB - LABEL AT BASE OF LINE DATA PATTERN END TABLE.
6752 * DPLENB - LABEL AT BASE OF LINE DATA PATTERN LENGTH TABLE.
6753 * EXCNTB - LABEL AT BASE ADDRESS OF EXTRA CHAR COUNTERS TABLE.
6754 * IESTAT - PRESENT STATE OF THE RX.IE AND TX.IE BITS.
6755 * NUMLNS - EQUATED TO NUMBER OF LINES ON THE DUT.
6756 * RXCNTB - LABEL AT BASE ADDRESS OF RX CHARACTER COUNTERS TABLE.
6757 * RXPTRB - LABEL AT BASE ADR OF 'NEXT RX CHAR' POINTERS TABLE.
6758 * TXCNTB - LABEL AT BASE ADDRESS OF TX CHARACTER COUNTERS TABLE.
6759 * TXPTRB - LABEL AT BASE ADR OF 'NEXT TX CHAR' POINTERS TABLE.
6760 * CBB - LABEL AT BASE OF TX/RX CONTROL BLOCK.
6761 * CB CONTENTS - TX/RX CONTROL BLOCK CONTAINS THE FOLLOWING:
6762 * CBLPRA - DUT LPR CONTENTS.
6763 * CBLNCA - DUT LNCTRL CONTENTS.
6764 * CBDPAA - ADDRESS OF BEGINNING OF DATA PATTERN.
6765 * CBDPLA - LENGTH IN BYTES OF DATA PATTERN.
6766 * CBDPNA - NUMBER OF DATA PATTERNS TO TRANSMIT.
6767 * CBMAPA - BIT MAP OF LINES TO BE INITIALIZED.
6768 * CBLPBA - TYPE OF LOOPBACK TO BE USED FOR TEST.
6769 * CBOFSA - AMOUNT TO OFFSET EACH TX START IN THE DATA PAT.
6770 * TXRXLB - LABEL AT BASE OF TX/RX LINE ASSOCIATION TABLE.
6771 *
6772 * OUTPUTS:
6773 * CHCNT - TABLE OF NUMBER OF LINE TX CHARACTERS (INITIALIZED).
6774 * CHRTOT - MAXIMUM NUMBER OF CHARS TO RECEIVE (2 * PAT LENGTH).
6775 * DPEND - TABLE OF DATA PATTERN ENDS (INITIALIZED).
6776 * DPLEN - TABLE OF DATA PATTERN LENGTHS (INITIALIZED).
6777 * DUT LNCTRL - LINE CONTROL REGISTERS (INITIALIZED).
6778 * DUT LPR - LINE PARAMETER REGISTERS (INITIALIZED).
6779 * EXCNT - TABLE OF EXTRA RX CHAR COUNTS (CLEARED, SELECTED LINES).
6780 * RXCNT - TABLE OF RX CHARACTER COUNTS (CLEARED, SELECTED LINES).
6781 * RXDNF - 'RECEPTION DONE' FLAGS (CLEARED FOR SELECTED LINES).
6782 * RXPTR - TABLE OF RECEIVE POINTERS (INITIALIZED).
6783 * TXCNT - TABLE OF TX CHARACTER COUNTERS (CLEARED, SELECTED LINES).
6784 * TXDNF - 'TRANSMISSION DONE' FLAGS (CLEARED FOR SELECTED LINES).
6785 * TXPTR - TABLE OF TRANSMIT POINTERS (INITIALIZED).
6786 * TXRXL - TX/RX LINE ASSOCIATION TABLE (INITIALIZED).
6787 *
6788 * CALLING SEQUENCE: JSR PC, TXRINI
6789 *
6790 * COMMENTS: IF THE CALCULATION OF THE CHRTOT VALUE (2 TIMES THE DATA
6791 * PATTERN LENGTH) RESULTS IN A NUMBER GREATER THAN 64K THEN
6792 * CHRTOT IS INITIALIZED TO 64K - 1.
6793 * THIS ROUTINE WILL NOT FORCE INTERNAL LOOPBACK BASED ON THE
6794 * LOOPBACK TYPE IN CBLPBA. THE USER MUST SET UP CBLNCA CORRECTLY
6795 * TO GET INTERNAL LOOPBACK.
6796 *
6797 * SUBORDINATE ROUTINES CALLED: WTWLNC, WTWLPR,

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 165
GLOBAL SUBROUTINE - TXRINI -

```

6796                                     :-- *****
6797 023666                               TXRINI:: SAVE          ;SAVE CONTENTS OF GPRS R0 THRU R5.
6798 023666 004537 005232                 JSR      R5,PREG05      ;CALL REGISTER SAVE SUBRT.
6799                                     :+
6800                                     : SET UP THE LPR AND LNCTRL REGISTERS AS SPECIFIED IN THE TX/RX CONTROL BLOCK.
6801                                     :-
6802 023672 013705 003042                 MOV      CBMAPA,R5      ;GET THE BIT MAP OF SELECTED LINES.
6803 023676 013700 003032                 MOV      CBLNCA,R0     ;GET THE NEW LNCTRL CONTENTS.
6804 023702 023727 003044 000001         CMP      CBLPBA,#1     ;CHECK IF INTERNAL LOOPBACK HAS BEEN SELECTED.
6805 023710 001002                         BNE      2$            ;SKIP SETTING INT. LOPBCK IN MAINTENANCE FIELD.
6806 023712 052700 000200                 BIS      #200,R0      ;SET INTERNAL LOOPBACK IN MAINTENANCE FIELD.
6807 023716 004737 025014 2$:           JSR      PC,WTWLNC     ;SET UP THE LNCTRL REGS FOR SELECTED LINES.
6808 023722 013700 003030                 MOV      CBLPRA,R0     ;GET THE NEW LPR CONTENTS.
6809 023726 004737 025044                 JSR      PC,WTWLPR     ;SET UP THE LPR REGISTERS FOR SELECTED LINES.
6810 023732 004737 023412                 JSR      PC,TXENBL     ;ENABLE TX FOR ALL SELECTED LINES.
6811                                     :+
6812                                     : SET UP AND BEGIN LOOP WHICH HANDLES ONE LINE PER ITERATION.
6813                                     :-
6814 023736 005004                         CLR      R4            ;CLEAR THE LINE OFFSET.
6815 023740 013705 003034                 MOV      CBDPAA,R5     ;INITIALIZE THE TX START ADDRESS VALUE.
6816 023744 013703 003036                 MOV      CBDPLA,R3     ;GET THE LENGTH OF THE DATA PATTERN.
6817 023750 060503                         ADD      R5,R3         ;CALCULATE END ADDRESS OF THE DATA PATTERN.
6818 023752 036437 002332 003042 4$:   BIT      BITTBL(R4),CBMAPA ;CHECK IF THIS LINE IS SELECTED FOR INIT.
6819 023760 001452                         BEQ      12$          ;SKIP SET UP IF LINE IS NOT SELECTED.
6820                                     :+
6821                                     : THIS LINE IS SELECTED FOR INITIALIZATION.
6822                                     : SET UP PROPER ENTRY IN NUMBER OF CHARS TO TX AND RX TABLE.
6823                                     : INCLUDE CHAR COUNT ON THIS LINE IN MAX ALLOWABLE CHAR TOTAL FOR ALL LINES.
6824                                     :-
6825 023762 013701 003036                 MOV      CBDPLA,R1     ;GET THE LENGTH OF THIS LINE'S DATA PATTERN.
6826 023766 013702 003040                 MOV      CBDPNA,R2     ;GET THE NUMBER OF PATTERNS TO TX AND RX.
6827 023772 004737 017470                 JSR      PC,MUL16U     ;CALCULATE THE TOTAL NUMBER OF CHARS TO TX/RX.
6828 023776 010164 003350                 MOV      R1,CHCNTB(R4) ;SET UP THE NUMBER OF TX/RX CHARS FOR LINE.
6829 024002 060137 002404                 ADD      R1,CHRTOT     ;ADD TWICE THE NUMBER OF CHARACTERS TO TX/RX
6830 024006 103403                         BCS      6$           ; ON THIS LINE TO THE TOTAL NUMBER OF CHARS
6831 024010 060137 002404                 ADD      R1,CHRTOT     ; WHICH WE WILL ALLOW TO BE RECEIVED ON
6832 024014 103003                         BCC      8$           ; ALL LINES.
6833 024016 012737 177777 002404 6$:   MOV      #-1,CHRTOT    ; SET MAX CHAR TOTAL TO -1 IF OVERFLOW.
6834 024024 8$:
6835                                     :+
6836                                     : SET UP THE DATA PATTERN END AND LENGTH FOR THIS LINE.
6837                                     :-
6838 024024 013764 003036 003110         MOV      CBDPLA,DLENB(R4) ;SET UP TX DATA PATTERN LENGTH FOR THIS LINE.
6839 024032 010364 003050                 MOV      R3,DPENDB(R4) ;SET UP TX DATA PAT END ADDRESS FOR THIS LINE.
6840                                     :+
6841                                     : SET UP THE TX COUNTER AND CHARACTER POINTER FOR THIS LINE.
6842                                     :-
6843 024036 005064 003410                 CLR      TXCNTB(R4)    ;CLEAR THE TX COUNTER FOR THIS LINE.
6844 024042 010564 003250                 MOV      R5,TXPTRB(R4) ;SET UP THE TX CHAR POINTER FOR THIS LINE.
6845                                     :+
6846                                     : SET UP THE TX/RX LINE ASSOCIATION OFFSET TABLE ENTRY FOR THIS LINE.
6847                                     :-
6848 024046 010402                         MOV      R4,R2         ;SELECT LINE OFFSET FOR NON-STAGGERED LPBK.
6849 024050 023727 003044 000002         CMP      CBLPBA,#2     ;TEST FOR STAGGERED LOOPBACK.
6850 024056 001003                         BNE      10$          ;SKIP SETTING STAGGERED LPBK IF NOT.
6851 024060 006202                         ASR      R2            ;FORM BYTE OFFSET INTO TABLE FROM TX LINE #.

```

CVDHCAO DHV-11 FUNC TST PART1
 CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 166
 GLOBAL SUBROUTINE - TXRINI -

```

6852 024062 116202 005202
6853 024066 010264 005142
6854
6855
6856
6857
6858 024072 005062 003450
6859 024076 005062 003150
6860 024102 010562 003310
6861
6862
6863
6864 024106 063705 003046
6865 024112 020503
6866 024114 103403
6867 024116 163705 003036
6868 024122 000773
6869
6870
6871
6872 024124 005204
6873 024126 005204
6874
6875
6876
6877 024130 020427 000020
6878 024134 002706
6879
6880 024136
6881 024136 004736
6882 024140 000207

10$:   MOVB  STGTRB(R2),R2 ;GET THE RX LINE CORRESPONDING WITH TX LINE.
       MOV   R2,TRXRLB(R4) ;LOAD TX TABLE ENTRY WITH RX LINE OFFSET.
       :+
       : SET UP THE RX COUNTERS AND CHARACTER POINTER FOR THE RX LINE WHICH
       : IS ASSOCIATED WITH THIS TX LINE.
       : -
       CLR   RXCNTB(R2) ;CLEAR THE RX COUNTER FOR THIS RX LINE.
       CLR   EXCNTB(R2) ;CLEAR THE EXTRA CHAR COUNTER FOR THIS RX LINE.
       MOV   R5,RXPTRB(R2) ;SET UP THE RX CHAR POINTER FOR THIS RX LINE.
       :+
       : UPDATE THE TX START POINTER IN PREPARATION FOR THE NEXT LINE.
       : -
12$:   ADD   CBOFSA,R5 ;ADD THE TX OFFSET TO THE TX START POINTER.
14$:   CMP   R5,R3 ;COMPARE TX START WITH END OF DATA PATTERN.
       BLO  16$ ;SKIP WRAPAROUND IF START IS BEFORE PAT END.
       SUB  CBDPLA,R5 ;SUBTRACT DATA PATTERN LENGTH FROM START.
       BR   14$ ;LOOP UNTIL START IS WITHIN DATA PATTERN.
       :+
       : UPDATE THE TX LINE NUMBER OFFSET TO THE NEXT LINE.
       : -
16$:   INC   R4
       INC   R4
       :+
       : TEST FOR DONE HANDLING ALL POSSIBLE LINES ON THE DEVICE.
       : -
       CMP  R4,#NUMLNS*2 ;COMPARE OFFSET WITH 2 TIMES MAX # OF LINES.
       BLT  4$ ;LOOP IF NOT ALL LINES DONE.
60$:   PASS
       JSR  PC,@(SP)+ ;RESTORE GPRS. ;RETURN TO PREG05 SUBRT.
       RTS  PC
    
```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 167
GLOBAL SUBROUTINE - TXROFF -

6883
6884
6885
6886
6887
6888
6889
6890
6891
6892
6893
6894
6895
6896
6897
6898
6899
6900
6901
6902
6903
6904
6905
6906
6907
6908
6909
6910
6911
6912
6913
6914
6915
6916
6917
6918

```

.SBTTL GLOBAL SUBROUTINE - TXROFF -
*+ *****
* - TURN TX AND RX OFF ROUTINE -
* THIS SUBROUTINE IS USED TO TURN OFF DUT TRANSMISSION AND RECEPTION.
* THIS ROUTINE ACHIEVES THIS BY BOOSTING PROCESSOR PRIORITY TO 5 TO
* AVOID RX INTERRUPTS AND BY CLEARING ALL THE DUT TX.ENABLE BITS TO
* HALT TX (EITHER DMA OR SINGLE CHARACTER TX). THE STATES OF THE
* TX.ENABLE BITS AND THE PROCESSOR PRIORITY ARE SAVED FOR RESTORATION
* WHEN TX AND RX ARE RE-ENABLED.
* INPUTS: MAPLNS - BIT MAP OF ALL POSSIBLE LINES ON THE DUT.
* OUTPUTS: SAVPRI - SAVED PROCESSOR PRIORITY.
* SAVTEN - BIT MAP OF TX.ENBL BITS (BIT SETIF TX.ENBL WAS SET).
* CALLING SEQUENCE: JSR PC, TXROFF
* COMMENTS:
* SUBORDINATE ROUTINES CALLED: TXDSBL.
*-- *****

```

```

TXROFF:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
R5, PREG05 ;CALL REGISTER SAVE SUBRT.
GETPRI SAVPRI ;GET THE PRESENT PROCESSOR PRIORITY.
TRAP CSGPRI
MOV RO, SAVPRI
SETPRI #PRI05 ;DISABLE DUT INTERRUPTS.
MOV #PRI05, RO
TRAP CSSPRI
MOV #MAPLNS, R5 ;PREPARE TO DISABLE TX ON ALL DUT LINES.
JSR PC, TXDSBL ;CLEAR ALL DUT TX.ENABLE BITS.
MOV R5, SAVTEN ;PRESERVE THE PREVIOUS TX.ENABLE BIT STATES.
60$: PASS ;RESTORE GPRS.
PC, @ (SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 168
GLOBAL SUBROUTINE - TXRON -

6919
6920
6921
6922
6923
6924
6925
6926
6927
6928
6929
6930
6931
6932
6933
6934
6935
6936
6937
6938
6939
6940
6941
6942
6943
6944
6945
6946
6947
6948

```

.SBTTL GLOBAL SUBROUTINE - TXRON -
:++ *****
: * - TURN TX AND RX ON ROUTINE -
: * THIS SUBROUTINE IS USED TO TURN ON DUT TRANSMISSION AND RECEPTION.
: * THIS ROUTINE RESTORES THE DUT TX.ENABLE BITS AND THE PROCESSOR PRIORITY
: * TO THE STATES SAVED BY THE TXROFF ROUTINE.
: *
: * INPUTS: SAVPRI - SAVED PROCESSOR PRIORITY.
: * SAVTEN - BIT MAP OF TX.ENBL BITS (BIT SETIF TX.ENBL WAS SET).
: *
: * OUTPUTS: DUT TX.ENABLE BITS - SET TO SPECIFIED STATES.
: * PROCESSOR PRIORITY - SET TO SPECIFIED PRIORITY.
: *
: * CALLING SEQUENCE: JSR PC, TXRON
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: TXENBL.
:-- *****

```

```

TXRON:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5, PREG05 ;CALL REGISTER SAVE SUBRT.
MOV SAVTEN, R5 ;GET THE SAVED STATES OF THE TX.ENABLE BITS.
JSR PC, TXENBL ;SET THE SPECIFIED TX.ENABLE BITS.
SETPRI SAVPRI ;RESTORE THE PROCESSOR PRIORITY.
MOV SAVPRI, R0
TRAP C$SPRI

60$: PASS ;RESTORE GPRS.
PC, @ (SP)+ ;RETURN TO PREG05 SUBRT.

RTS PC

```

```

024202
024202 004537 005232
024206 013705 002244
024212 004737 023412
024216
024216 013700 002246
024222 104441
024224
024224 004736
024226 000207

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 169
GLOBAL SUBROUTINE - TXRREP -

6949
6950
6951
6952
6953
6954
6955
6956
6957
6958
6959
6960
6961
6962
6963
6964
6965
6966
6967
6968
6969
6970
6971
6972
6973
6974
6975
6976
6977
6978
6979
6980
6981
6982
6983
6984
6985
6986
6987
6988
6989
6990
6991
6992
6993
6994
6995
6996
6997

```

.SBTTL GLOBAL SUBROUTINE - TXRREP -
:++ *****
:          - REPORT FINAL TX/RX ERRORS ROUTINE -
:          THIS SUBROUTINE REPORTS ERRORS WHICH ARE FOUND AFTER THE COMPLETION
:          OF THE TX, RX, AND VERIFICATION OF DATA PATTERNS. IT REPORTS ERRORS
:          DEALING WITH INCOMPLETE TX OR RX AND WITH DMA_START BITS.
:
: INPUTS:      ACTLNS - BIT MAP OF ACTIVE DUT LINES.
:              DPLENB - LABEL AT BASE OF THE DATA PATTERN LENGTHS TABLE.
:              ERRMSG - ADDRESS OF PRIMARY ERROR MESSAGE FOR THIS ROUTINE.
:              ERRNBR - ERROR NUMBER OF ERROR REPORTED IN THIS ROUTINE.
:              RXCNTB - LABEL AT BASE OF THE RX CHARACTER COUNTERS TABLE.
:              RXDNF - RECEPTION DONE FLAGS.
:              TXCNTB - LABEL AT BASE OF THE TX CHARACTER COUNTERS TABLE.
:              TXDNF - TRANSMISSION DONE FLAGS.
:              TXINTF - CONTAINS BIT MAP OF LINES WITH DMA_START BIT ERRORS.
:
: OUTPUTS:     CARRY FLAG - RESTORED TO ITS ENTERING VALUE.
:              ERRBLK - ADDRESS OF THE ERROR REPORTING ROUTINE (DESTROYED).
:              MESSAGES MAY BE PRINTED AT THE OPERATOR CONSOLE.
:
: CALLING SEQUENCE: JSR PC, TXRREP
:
: COMMENTS:    THIS ROUTINE REPORTS ERRORS AT INITIAL ERRNBR THRU
:              INITIAL ERRNBR+2.
:              IF NO LINES FAILED TO COMPLETE THEIR RECEPTION OR FAILED TO
:              COMPLETE THEIR TRANSMISSION OR HAD DMA_START BIT ERRORS
:              THEN NO MESSAGES ARE PRINTED.
:
: SUBORDINATE ROUTINES CALLED: CONMAP, ER9005, ER9102, RDMAST, RRXNDN, RTXNDN.
:-- *****

```

```

TXRREP:: SAVE          ;SAVE CONTENTS OF GPRS R0 THRU R5.
                    JSR R5, PREG05 ;CALL REGISTER SAVE SUBRT.
                    ROR R3          ;ROTATE CARRY INTO GPR TO SAVE CARRY STATE.
                    MOV ERRNBR, R4 ;SAVE THE INITIAL ERROR NUMBER VALUE.
                    MOV ACTLNS, R5 ;GET THE ACTIVE LINES BIT MAP.
                    JSR PC, RDMAST  ;REPORT ANY DMA_START BIT ERRORS.
                    INC ERRNBR     ;SELECT INITIAL ERROR NUMBER + 1.
                    JSR PC, RTXNDN  ;REPORT TX NOT COMPLETE IF NECESSARY.
                    INC ERRNBR     ;SELECT INITIAL ERROR NUMBER + 2.
                    JSR PC, CONMAP  ;GENERATE AN ASSOCIATED LINE BIT MAP.
                    JSR PC, RRXNDN  ;REPORT RX NOT COMPLETE IF NECESSARY.
                    MOV R4, ERRNBR ;RESTORE THE INITIAL ERROR NUMBER VALUE.
:
60$: ROL R3           ;ROTATE SAVED CARRY STATE BACK INTO CARRY.
    PASS            ;RESTORE GPRS, THIS ROUTINE PRESERVES THE
                    JSR PC, @SP+  ;RETURN TO PREG05 SUBRT.
    RTS PC         ; INITIAL CARRY STATE.

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 170
GLOBAL SUBROUTINE

- UNSDIV -

```

6998 .SBTTL GLOBAL SUBROUTINE - UNSDIV -
6999 *+ *****
7000 * - UNSIGNED DIVIDE ROUTINE -
7001 * THIS SUBROUTINE IS USED TO DIVIDE A 32 BIT UNSIGNED DIVIDEND BY A
7002 * 16 BIT UNSIGNED DIVISOR GIVING A 16 BIT QUOTIENT. ALL NUMBERS ARE
7003 * CONSIDERED TO BE UNSIGNED. A SUCCESS FLAG IS NOT SET ON RETURN IF
7004 * THE QUOTIENT WAS TOO BIG TO BE CONTAINED IN 16 BITS.
7005 *
7006 * INPUTS: R1 - THE DIVISOR, UNSIGNED, 16 BITS.
7007 * R2 - MOST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
7008 * R3 - LEAST SIGNIFICANT WORD OF THE DIVIDEND, UNSIGNED, 16 BITS.
7009 *
7010 * OUTPUTS: R1 - QUOTIENT, UNSIGNED, 16 BITS (177777 IF OVERFLOW).
7011 * CARRY - SUCCESS FLAG, SET IF COMPLETE QUOTIENT FITS IN 16 BITS.
7012 *
7013 * CALLING SEQUENCE: JSR PC,UNSDIV
7014 *
7015 * COMMENTS: IF THE DIVISOR IS 0 THE QUOTIENT IS RETURNED AS ALL ONES
7016 * (177777) AND THE CARRY IS CLEAR REGARDLESS OF THE DIVIDEND.
7017 *
7018 * SUBORDINATE ROUTINES CALLED: NONE.
7019 *-- *****
7020
7021 024310 UNSDIV:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
7022 024310 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
7023
7024 *+ CHECK FOR QUOTIENT GREATER THAN 16 BITS CONDITION.
7025 *--
7026 024314 010204 MOV R2,R4 ;GET MSW OF DIVIDEND FOR SUBTRACT.
7027 024316 160104 SUB R1,R4 ;SUBTRACT DIVISOR FROM MSW OF DIVIDEND.
7028 024320 103403 BCS 2$ ;IF IT DIDN'T GO, WE HAVE QUOTIENT < 16 BITS.
7029 024322 012701 177777 MOV #-1,R1 ;SET QUOTIENT TO ALL ONES (177777),
7030 024326 000442 BR 60$ ;EXIT WITH CARRY CLEAR.
7031
7032 *+ SET UP COUNTERS AND VARIOUS WORKING GPRS.
7033 *--
7034 024330 005004 2$: CLR R4 ;CLEAR THE LSW OF THE DIVISOR.
7035 024332 000241 CLC ;CLEAR CARRY FOR THE SHIFT OF THE DIVISOR.
7036 024334 006001 ROR R1 ; DIVISOR BY
7037 024336 006004 ROR R4 ; 2(UNSIGNED)
7038 024340 012700 000020 MOV #16.,R0 ;SET UP INITIAL SHIFT COUNT TO 16.
7039
7040 *+ THE SUBTRACT AND SHIFT LOOP.
7041 *--
7042 024344 010246 4$: MOV R2,-(SP) ;SAVE MSWORD OF DIVIDEND.
7043 024346 010346 MOV R3,-(SP) ;SAVE LSWORD OF DIVIDEND.
7044 024350 160403 SUB R4,R3 ;LSWORD DIVIDEND - LSWORD OF DIVISOR.
7045 024352 005602 SBC R2 ;MSWORD DIVIDEND - BORROW
7046 024354 103402 BCS 6$ ;IF BORROW FROM BORROW SUBTRACT, IT DIDN'T GO.
7047 024356 160102 SUB R1,R2 ;MSWORD DIVIDEND - MSWORD OF DIVISOR.
7048 024360 103003 BCC 8$ ;IF NO BORROW, IT WENT, CARRY IS CLEAR.
7049
7050 *+ IT DIDN'T GO, SO WE SHIFT A 1 INTO THE QUOTIENT (COMPLEMENTED LATER).
7051 * CARRY IS SET.
7052
7053 024362 012603 6$: MOV (SP)+,R3 ;RESTORE LSWORD OF DIVIDEND.

```

CV
CV

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 171
GLOBAL SUBROUTINE - UNSDIV -

```

7054 024364 012602      MOV      (SP)+,R2      ;RESTORE MSWORD OF DIVIDEND.
7055 024366 000401      BR       10$          ;GOTO SHIFT 1 INTO THE QUOTIENT.
7056
7057      ;+
7058      ; IT WENT, SO WE RESTORE THE STACK AND SHIFT A 0 INTO QUOTIENT (WILL BE
7059      ; COMPLEMENTED LATER).  CARRY IS CLEAR.
7060 024370 012626      8$:     MOV      (SP)+,(SP)+  ;POP THE SAVED DIVIDEND OFF OF THE STACK.
7061
7062      ;+
7063      ; SHIFT THE RESULT OF THE SUBTRACT ATTEMPT INTO THE QUOTIENT SHIFT REG.
7064 024372 006105      10$:    ROL      R5          ;SHIFT NEXT BIT INTO THE INVERTED QUOTIENT.
7065 024374 000241      CLC          ;DIVIDE THE
7066 024376 006001      ROR      R1          ; DEVISOR BY
7067 024400 006004      ROR      R4          ; 2 (UNSIGNED).
7068 024402 005300      DEC      R0          ;COUNT THIS SHIFT AND SUBTRACT.
7069 024404 001357      BNE      4$          ;LOOP FOR ANOTHER SHIFT & SUB IF NOT DONE.
7070 024406 005105      COM      R5          ;GET QUOTIENT FROM INVERTED QUOTIENT.
7071
7072      ;+
7073      ; NOW WE EITHER ROUND UP OR LEAVE QUOTIENT ALONE.
7074 024410 000241      CLC          ;CLEAR THE CARRY FOR THE SHIFT OF THE DIVIDEND.
7075 024412 006103      ROL      R3          ;MULTIPLY LSWORD OF DIVIDEND BY 2, MSWORD IS 0.
7076 024414 103402      BCS      12$        ;IF CARRY FROM SHIFT, ROUND UP.
7077 024416 160403      SUB      R4,R3      ;SUBTRACT DIVISOR FROM DIVIDEND.
7078 024420 103403      BCS      14$        ;IF BORROW, DON'T ROUND UP.
7079
7080      ;+
7081      ; ROUND UP, EXTRA SUBTRACT WENT.
7082 024422 005205      12$:    INC      R5          ;INCREMENT THE QUOTIENT BY ONE.
7083 024424 001001      BNE      14$        ;IF NO OVERFLOW, WE LEAVE THE ROUND UP.
7084 024426 005305      DEC      R5          ;DON'T LET ROUNDING CAUSE OVERFLOW.
7085
7086      ;+
7087      ; ALL DONE, PASS QUOTIENT AND EXIT.
7088 024430 010501      14$:    MOV      R5,R1      ;PASS QUOTIENT BACK IN R1.
7089 024432 000261      SEC          ;INDICATE NO OVERFLOW.
7090
7091 024434 010501      60$:    PASS      R1          ;RESTORE GPRS, LEAVE THE FOLLOWING INTACT:
7092 024434 010166 000004      MOV      R1,R1SLOT(SP) ;PUT R1 IN STACK SLOT.
7093 024440 004736      JSR      PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
7094
7095 024442 000207      RTS      PC          ;R1 - 16 BIT, UNSIGNED QUOTIENT,
                          ;CARRY - SET INDICATES NO OVERFLOW (SUCCESS).

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 172
GLOBAL SUBROUTINE

- UPDCHR -

7096
7097
7098
7099
7100
7101
7102
7103
7104
7105
7106
7107
7108
7109
7110
7111
7112
7113
7114
7115
7116
7117
7118
7119
7120
7121
7122
7123
7124
7125
7126
7127
7128
7129
7130
7131
7132
7133
7134
7135
7136
7137
7138
7139
7140
7141
7142
7143
7144
7145
7146
7147
7148
7149
7150
7151

```

.SBTTL GLCBAL SUBROUTINE - UPDCHR -
*+ *****
* - UPDATE CHARACTER POINTERS AND COUNTERS ROUTINE -
* THIS SUBROUTINE UPDATES THE POINTERS AND COUNTERS ASSOCIATED WITH
* THE RECEPTION OF A CHARACTER ON A SPECIFIED LINE. THE RECEIVE CHAR
* POINTER IS SET TO THE NEXT EXPECTED CHARACTER, THE RECEIVE CHAR COUNT
* IS INCREMENTED, AND THE COUNT IS CHECKED TO DETERMINE IF THE RECEPTION
* IS COMPLETE. IF THE RECEPTION IS COMPLETE THE RECEPTION DONE FLAG
* IS SET FOR THE SPECIFIED LINE.
*
* INPUTS: R3 - LINE NUMBER TIMES 2 OF LINE ON WHICH CHAR WAS RECEIVED.
* BITTBL - LABEL OF TABLE OF WORDS USED TO FORM SINGLE BIT MAPS.
* CHCNTB - BASE OF NUMBER OF CHARS TO TX ON EACH LINE TABLE.
* DPENDB - BASE OF DATA PATTERN END ADDRESSES TABLE.
* DPLENB - BASE OF DATA PATTERN LENGTHS TABLE.
* RXCNTB - BASE OF THE RX CHARACTER COUNTERS TABLE.
* RXPTRB - BASE OF THE RX CHARACTER POINTERS TABLE.
* TXRXLB - BASE OF TX/RX LINE NUMBER ASSOCIATION TABLE.
*
* OUTPUTS: FOLLOWING VARIABLES UPDATED FOR LINE ON WHICH CHAR WAS RECEIVED:
* RXCNT - COUNT OF THE NUMBER OF CHARACTERS RECEIVED ON LINE.
* RXDNF - RX DONE FLAGS WITH BIT0 FOR LINE 0 ... (UPDATED).
* RXPTR - UPDATED TO POINT TO THE NEXT EXPECTED CHAR ON LINE.
*
* CALLING SEQUENCE: JSR PC,UPDCHR
*
* COMMENTS:
*
* SUBORDINATE ROUTINES CALLED: NONE.
*+ *****
UPDCHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;CALL REGISTER SAVE SUBRT.
JSR R5,PREG05
MOV TXRXLB(R3),R2 ;GET TX LINE NUMBER OFFSET FOR THIS RX LINE.
*+
; UPDATE THE RX DATA POINTER WITH WRAPAROUND AT THE END OF THE DATA PATTERN.
*+
MOV RXPTRB(R3),R1 ;GET THE RX DATA POINTER FROM THE RX PTR TABLE.
INC R1 ;INCREMENT THE RX POINTER VALUE BY 1.
CMP R1,DPENDB(R2) ;CMP RX PTR VALUE WITH ADR OF END OF DATA PAT.
BLO 2$ ;SKIP WRAPPING RX PTR AROUND IF NOT AT END.
SUB DPLENB(R2),R1 ;WRAP RX PTR AROUND TO START OF DATA PATTERN.
2$: MOV R1,RXPTRB(R3) ;UPDATE THE RX POINTER WITH THE NEW VALUE.
*+
; UPDATE THE RX CHARACTER COUNT WITH OVERFLOW DETECTION.
*+
MOV RXCNTB(R3),R1 ;GET THE RX CHARACTER COUNT.
INC R1 ;INCREMENT THE RX CHAR COUNT VALUE BY 1.
BNE 4$ ;SKIP SETTING COUNT TO MAX IF NO OVERFLOW.
MOV #-1,R1 ;SET RX CHAR COUNT VALUE TO MAX VALUE.
4$: MOV R1,RXCNTB(R3) ;UPDATE THE RX CHAR COUNT WITH NEW VALUE.
*+
; CHECK FOR RX COMPLETION ON THIS LINE.
; IF RX IS COMPLETE ON THIS LINE, SET THE CORRECT RX DONE FLAG.
*+
MOV CHCNTB(R2),R4 ;GET THE NUMBER OF TX CHARS IN COMPLETE TX.
CMP R1,R4 ;COMPARE RX CHAR COUNT WITH NUMBER OF TX CHARS.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 173
CVDHCA.P11 12-JUL-83 11:44 GLOBAL SUBROUTINE - UPDCHR -

```

7152 024526 103403          BLO  60$          ;EXIT ROUTINE IF NOT ALL CHARS RECEIVED.
7153 024530 056337 002332 002412  BIS  BITTBL(R3),RXDNF      ;SET THE RX DONE FLAG FOR THIS LINE.
7154                                     60$:  PASS          ;RESTORE GPRS.
7155 024536                                     JSR  PC,@(SP)+          ;RETURN TO PREG05 SUBRT.
7156 024536 004736          RTS  PC
7157 024540 000207

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 174
GLOBAL SUBROUTINE

- VANSUP -

7158
7159
7160
7161
7162
7163
7164
7165
7166
7167
7168
7169
7170
7171
7172
7173
7174
7175
7176
7177
7178
7179
7180
7181
7182
7183
7184
7185
7186
7187
7188
7189
7190
7191
7192
7193
7194
7195
7196
7197
7198
7199
7200
7201
7202
7203
7204
7205
7206
7207
7208
7209
7210
7211
7212
7213

```

.SBTTL GLOBAL SUBROUTINE - VANSUP -
:++ *****
: * - TRANSMISSION / RECEPTION SET-UP ROUTINE -
: *
: * THIS ROUTINE IS USED TO INITIALISE BOTH THE DUT AND THE
: * TRANSMISSION/RECEPTION CONTROL PARAMETERS TO THE CORRECT
: * STATE, PRIOR TO A SINGLE CHARACTER OR DMA TRANSMISSION,
: * RECEPTION TEST.
: *
: * INPUTS: R1 - TX, RX LPR CONTENTS.
: * R2 - START ADDRESS OF DATA PATTERN TO TRANSMIT.
: * R3 - LENGTH OF DATA PATTERN.
: * R4 - NUMBER OF PATTERNS TO TRANSMIT.
: * ACTLNS - CONTAINS A BIT MAP OF ALL CURRENTLY ACTIVE LINES.
: * LOPBCK - CONTAINS THE TYPE OF LOOPBACK MODE SELECTED.
: * CBB - LABEL AT BASE OF TX/RX CONTROL BLOCK.
: *
: * OUTPUTS: THE CONTENTS OF THE TX/RX CONTROL BLOCK (CCB) ARE DESTROYED.
: * THE INDIRECT ADDRESS FIELD OF THE DUT CSR MAY BE DESTROYED.
: * THE DUT'S LPR'S AND LNC'S MAY BE MODIFIED.
: * THE FOLLOWING POINTERS AND COUNTERS ARE INITIALISED:
: * CHCNT,CHRTOT,DPEND,DPLEN,EXCNT,RXCNT,RXPTR,TXCNT,
: * TXPTR,TXRXL.
: * CHRTOT, RXDONF, TXDONF AND TXINTF ARE CLEARED.
: *
: * CALLING SEQUENCE: JSR PC,VANSUP
: *
: * COMMENTS: MODEM LOOPBACK MODE IS INHIBITED IF IT HAS BEEN SELECTED
: * VIA HARDWARE P-TABLE QUESTIONS, AND INTERNAL LOOPBACK MODE
: * IS FORCED TO TAKE PLACE.
: *
: * SUBORDINATE ROUTINES CALLED: CONMAP,RXENBL,TXRINI.
:-- *****
VANSUP:: SAVE ;SAVE CONTENTS OF THE GPR'S R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
CLR CHRTOT ;CLEAR TOTAL RECEIVED CHAR COUNTER.
CLR TXINTF ;CLEAR FLAGS USED TO LOG DMA H.OVER ERRORS.
CLR TXDONF ;CLEAR THE TX DONE FLAGS.
CLR RXDONF ;CLEAR THE RX DONE FLAGS.
:++
: * SET UP THE TRANSMISSION/RECEPTION CONTROL BLOCK TO THE DESIRED STATE.
:--
MOV R1,CBB ;SET CONTENTS OF LPR PARAMS IN TX/RX C.BLK.
MOV #CBB,R1 ;GET BASE ADDRESS OF CONTROL BLOCK.
INC R1 ;INCREMENT ADDRESS FOR NEXT WORD
INC R1 ;INITIALISE THE FOLLOWING IN THE CNTRL.BLK:
MOV #4,(R1)+ ; LNCtrl PARAMETER, ENABLE RECEIVERS.
MOV R2,(R1)+ ; START ADDRESS OF DATA PATTERN.
MOV R3,(R1)+ ; DATA PATTERN LENGTH.
MOV R4,(R1)+ ; NUMBER OF DATA PATTERNS TO TRANSMIT.
MOV ACTLNS,(R1)+ ; BIT MAP OF LINES TO INITIALISE.
BIT #BIT2,LOPBCK ;TEST IF MODEM LOOPBACK MODE HAS BEEN SELECTED.
BEQ 2$ ;DONT SELECT INTERNL LOPBCK IF STAGRD OR LOCAL.
MOV #1,R2 ;FORCE INTERNAL LOOPBACK MODE TO BE SELECTED.

```

024542			
024542	004537	005232	
024546	005037	002404	
024552	005037	002252	
024556	005037	002410	
024562	005037	002412	
024566	010137	003030	
024572	012701	003030	
024576	005201		
024600	005201		
024602	012721	000004	
024606	010221		
024610	010321		
024612	010421		
024614	013721	002174	
024620	032737	000004	002176
024626	001404		
024630	012702	000001	

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 175

GLOBAL SUBROUTINE

- VANSUP -

```

7214 024634 110221          MOVB   R2,(R1)+      ;INITIALISE LOOPBACK MODE IN CONTROL BLOCK.
7215 024636 000402          BR     4$           ;SKIP NEXT INSTRUCTION IF IN MODEM LOOPBACK.
7216 024640 113721 002176    2$:   MOVB   LOPBCK,(R1)+ ;SET LOOPBACK MODE.
7217 024644 005201          4$:   INC    R1         ;INCREMENT ADDRESS FOR THE NEXT WORD.
7218 024646 012711 000002    MOV    #2,(R1)      ;SET AMOUNT OF OFFSET EACH TX STARTS AT TO 2.
7219
7220          :+
7221          : INITIALISE THE DUT AND THE ASSOCIATED POINTERS AND COUNTERS, TO THE STATE
7222          : DICTATED BY THE CONTENTS OF THE TX/RX CONTROL BLOCK.
7223 024652 004737 023666    :-
7224          :+
7225          : INITIALISE POINTERS AND COUNTERS FOR INACTIVE LINES TO ZERO.
7226          :-
7227 024656 012701 000377    MOV    #MAPLNS,R1   ;GET THE LINE BIT MAP FOR ALL LINES.
7228 024662 013702 002174    MOV    ACTLNS,R2   ;GET THE ACTIVE LINE BIT MAP.
7229 024666 005101          COM    R1
7230 024670 005102          COM    R2
7231 024672 040102          BIC   R1,R2        ;GENERATE AN IN-ACTIVE LINE BIT MAP.
7232 024674 010237 003042    MOV    R2,CBMAPA   ;MOVE BIT MAP TO THE CONTROL BLOCK.
7233 024700 005037 003040    CLR   CBDPNA       ;CLEAR THE REPEAT TX COUNT IN CNTRL BLCK.
7234 024704 004737 023666    JSR   PC,TXRINI    ;SET UP PARAMETERS FOR INACTIVE LINES.
7235
7236          :+
7237          : DISABLE RECEIVERS ON ALL LINES TO ENSURE CORRECT INITIALISATION OF ONLY THE
7238          : LINES THAT ARE SELECTED.
7239 024710 012705 000377    :-
7240 024714 004737 022124    MOV    #MAPLNS,R5   ;SET-UP BIT MAP FOR ALL LINES.
7241          :+
7242          :+
7243          :+
7244          :+
7244 024720 013705 002174    MOV    ACTLNS,R5   ;GET THE ACTIVE LINE BIT MAP.
7245 024724 004737 015546    JSR   PC,CONMAP    ;GENERATE AN ASSOCIATED LINE BIT MAP.
7246 024730 004737 022220    JSR   PC,RXENBL    ;ENABLE RECEIVERS ON ASSOCIATED LINES.
7247 024734          60$:  PASS          ;RESTORE GPR'S.
7248 024734 004736          JSR   PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
7249 024736 000207          RTS    PC

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 176
GLOBAL SUBROUTINE - WAIBIS -

7250
7251
7252
7253
7254
7255
7256
7257
7258
7259
7260
7261
7262
7263
7264
7265
7266
7267
7268
7269
7270
7271
7272
7273
7274
7275
7276
7277
7278
7279
7280
7281
7282
7283
7284
7285
7286
7287
7288
7289
7290
7291
7292
7293
7294
7295
7296
7297

```

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

```

```

WAIBIS:: SAVE
        JSR    R5,PREG05    ;CALL REGISTER SAVE SUBRT.
        MOV    R2,R4        ;SET UP THE ADDRESS PARAMETER FOR MSLGET.
        MOV    R1,R2
        BIC    #170000,R1    ;SEPERATE DELAY COUNT OUT OF PASSED PARAMETER.
        BIC    #7777,R2     ;SEPERATE LINE NUMBER FIELD OF PASSED PARAM.
        SWAB   R2           ;PUT LINE NUMBER FIELD IN LSBYTE.
        ASR    R2           ;SHIFT THE LINE NUMBER FIELD INTO THE PROPER
        ASR    R2           ; POSITION TO USE IT AS A WORD TABLE OFFSET
        ASR    R2           ; FOR THE TABLE LOOKUP OF THE LINE BIT MAP.
        MOV    BITTBL(R2),R2 ;GET BIT MAP OF LINE TO TEST FROM TABLE.
        MOV    R2,R3        ;INDICATE THAT THE BIT SHOULD BE SET.
        JSR    PC,MSLGET    ;WAIT FOR THE BIT TO BE SET WITHIN TIME-OUT.
        ; CARRY IS CORRECT UPON MSLGET RETURN.
        MOV    R0,R2        ;PASS LAST VALUE READ AS OUTPUT PARAMETER.
        PASS   R2          ;RESTORE GPRS, EXCEPT THE FOLLOWING:
        MOV    R2,R2SLOT(SP) ;PUT R2 IN STACK SLOT.
        JSR    PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
        ; R2 - LAST VALUE READ LOOKING FOR CONDITION.
        ; CARRY - SUCCESS FLAG (SET IF BIT FOUND SET).
RTS    PC

```

```

024740
024740 004537 005232
024744 010204
024746 010102
024750 042701 170000
024754 042702 007777
024760 000302
024762 006202
024764 006202
024766 006202
024770 016202 002332
024774 010203
024776 004737 017340
025002 010002
025004
025004 010266 000006
025010 004736
025012 000207

```

60\$:

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 177
GLOBAL SUBROUTINE - WTWLNC -

```

7298 .SBTTL GLOBAL SUBROUTINE - WTWLNC -
7299 :+ *****
7300 :* - LINE CONTROL REGISTER SETUP ROUTINE -
7301 :* THIS SUBROUTINE IS USED TO SET THE DEVICE UNDER TEST (DUT) LINE
7302 :* CONTROL REGISTERS (LNCTRL) TO THE SPECIFIED STATE. ONLY THE LNCTRLS
7303 :* FOR THE SPECIFIED LINES ARE ALTERED.
7304 :*
7305 :* INPUTS: R0 - NEW LINE PARAMETERS.
7306 :* R5 - BIT MAP OF LINES TO BE ALTERED.
7307 :* CSRA - CONTAINS ADDRESS OF THE DUT CSR.
7308 :* IESTAT - CONTAINS THE CURRENT STATE OF THE TX AND RX INTERRUPT
7309 :* ENABLE BITS IN THE CSR.
7310 :* LNCTRA - CONTAINS ADDRESS OF THE DUT LNCTRL REGISTERS.
7311 :*
7312 :* OUTPUTS: LNCTRL - SPECIFIED DUT LINE CONTROL REGISTERS ARE ALTERED.
7313 :*
7314 :* CALLING SEQUENCE: JSR PC,WTWLNC
7315 :*
7316 :* COMMENTS:
7317 :*
7318 :* SUBORDINATE ROUTINES CALLED: ALTFLD.
7319 :-- *****
7320
7321 025014 WTWLNC:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
7322 025014 004537 005232 JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
7323
7324 :+ SET UP THE PARAMETERS FOR THE CALL TO ALTFLD.
7325 :-
7326 025020 013701 002212 MOV LNCTRA,R1 ;SET UP THE REGISTER ADDRESS PARAMETER.
7327 025024 010002 MOV R0,R2 ;SET UP THE DESIRED REGISTER CONTENTS.
7328 025026 010503 MOV R5,R3 ;SET UP THE BIT MAP OF LINES TO ALTER.
7329 025030 012704 177777 MOV #-1,R4 ;SELECT ALL REGISTER BITS TO BE ALTERED.
7330
7331 :+ CALL THE SUBROUTINE WHICH ALTERS THE REGISTER CONTENTS.
7332 :-
7333 025034 004737 014150 JSR PC,ALTFLD ;ALTER THE REGISTER CONTENTS.
7334
7335 025040 60$: PASS ;RESTORE GPRS.
7336 025040 004736 JSR PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
7337 025042 000207 RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 178
GLOBAL SUBROUTINE - WTWLPR -

7338
7339
7340
7341
7342
7343
7344
7345
7346
7347
7348
7349
7350
7351
7352
7353
7354
7355
7356
7357
7358
7359
7360
7361
7362
7363
7364
7365
7366
7367
7368
7369
7370
7371
7372
7373
7374
7375
7376
7377

025044
025044 004537 005232

025050 013701 002206
025054 010002
025056 010503
025060 012704 177777

025064 004737 014150

025070
025070 004736
025072 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 PARAMETER REGISTERS ARE ALTERED.
: *
: * CALLING SEQUENCE: JSR PC,WTWLPR
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: ALTFLD.
:-- *****
WTWLPR:: SAVE ;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.
60$: PASS ;RESTORE GPRS.
; PC,@(SP)+ ;RETURN TO PREG05 SUBRT.
RTS PC

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 179
INTERRUPT SERVICE ROUTINE - CLKINT -

```

7378 .SBTTL INTERRUPT SERVICE ROUTINE - CLKINT -
7379 :+ *****
7380 :* THIS ROUTINE IS EXECUTED CLKHRZ TIMES PER SECOND. IT DECREASES THE
7381 :* TWO TIMER COUNTERS DOWN TO ZERO.
7382 :*
7383 :* INPUTS: TIMER1 - TIMER COUNTER #1.
7384 :*          TIMER2 - TIMER COUNTER #2.
7385 :*          TIMER3 - TIMER COUNTER FOR CALL OF BREAK MACRO.
7386 :*
7387 :* OUTPUTS: THE 2 TIMER COUNTERS ARE DECREMENTED IF THEY ARE NOT ZERO.
7388 :*
7389 :* CALLING SEQUENCE: PUT #CLKINT IN THE CLOCK INTERRUPT VECTOR SLOT.
7390 :*                   PUT THE DESIRED TIME PERIOD (SECONDS TIMES CLKHRZ) IN
7391 :*                   EITHER TIMER1 OR TIMER2 AND POLL THE RESPECTIVE TIMER
7392 :*                   COUNTER TO DETECT ITS GOING TO 0 ON TIME-OUT.
7393 :*
7394 :* COMMENTS: THE 2 COUNTERS WILL NOT WRAPAROUND BUT WILL STOP AT 0. THIS
7395 :*            ALLOWS THE DETECTION OF A TIME-OUT ANY TIME AFTER THE TIME-OUT
7396 :*            HAS OCCURRED UNTIL THE TIMER COUNTER IS SET TO ANOTHER VALUE.
7397 :*
7398 :* SUBORDINATE ROUTINES CALLED: NONE.
7399 :-- *****
7400
7401 025074 005737 002270 CLKINT:: TST TIMER1 ;CHECK FOR TIMER1 AT ZERO.
7402 025100 001402 BEQ 2$ ;BRANCH TO LEAVE IT AT ZERO IF IT IS ZERO.
7403 025102 005337 002270 DEC TIMER1 ;DECREMENT TIME COUNT.
7404 025106 005737 002272 2$: TST TIMER2 ;CHECK FOR TIMER2 AT ZERO.
7405 025112 001402 BEQ 4$ ;BRANCH TO LEAVE IT ALONE IF IT'S ALREADY ZERO.
7406 025114 005337 002272 DEC TIMER2 ;DECREMENT TIME COUNT.
7407 025120 005337 002274 4$: DEC TIMER3 ;DECREMENT THE BREAK COUNT.
7408 025124 001006 BNE 60$ ;EXIT IF NOT TIME TO CALL BREAK.
7409 025126 013737 002276 002274 MOV BCOUNT,TIMER3 ;SET UP TIME TILL NEXT BREAK.
7410 025134 010046 MOV RO,-(SP) ;SAVE CONTENTS OF RO FROM BREAK MACRO.
7411 025136 BREAK ;CHECK FOR OPERATOR CONTROL/C. TRAP CSBRK
7412 025136 104422 MOV (SP)+,RO ;RESTORE CONTENTS OF RO.
7413 025140 012600 60$: RTI
7414 025142 000002

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.011 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 180
INTERRUPT SERVICE ROUTINE - RXCHRS -

7415
7416
7417
7418
7419
7420
7421
7422
7423
7424
7425
7426
7427
7428
7429
7430
7431
7432
7433
7434
7435
7436
7437
7438
7439
7440
7441
7442
7443
7444
7445
7446
7447
7448
7449
7450
7451
7452
7453
7454
7455
7456
7457
7458
7459
7460
7461
7462
7463
7464
7465
7466
7467
7468
7469
7470

```

.SBTTL INTERRUPT SERVICE ROUTINE - RXCHRS -
*****
- DMA RECEIVE INTERRUPT SERVICE ROUTINE -
THIS ROUTINE EXECUTES IN RESPONSE TO AN INTERRUPT CAUSED BY THE DUT
RX.DATA.AVAIL BIT BECOMING ACTIVE. THIS ROUTINE READS CHARACTERS FROM
THE DUT RECEIVE CHARACTER FIFO AND DEPOSITS THEM INTO THE RECEIVE
BUFFER IN MEMORY. IF THE NUMBER OF CHARACTERS IN THE RECEIVE BUFFER
EXCEEDS A SPECIFIED THRESHOLD, TRANSMISSION IS HALTED (BY CLEARING ALL
DUT TX.ENABLE BITS) AND IF THE RECEIVE BUFFER IS FULL RECEPTION IS
HALTED (BY DISABLING RX INTERRUPTS). THE ROUTINE EXITS IF THE RECEIVE
BUFFER BECOMES FULL OR IF A CHARACTER IS READ FROM THE FIFO WITH THE
DATA.VALID BIT CLEAR.

INPUTS:  RBUFA - CONTAINS ADDRESS OF THE DUT RX CHARACTER FIFO.
         RXBCNT - RX BUFFER CHARACTER COUNT.
         RXBDTX - EQUATED TO RX BUFFER LEVEL AT WHICH TO DISABLE TX.
         RXBEND - LABEL AFTER END OF THE RX BUFFER AREA IN MEMORY.
         RXBFUL - EQUATED TO THE CAPACITY OF THE RX BUFFER.
         RXBIPT - POINTER TO NEXT AVAILABLE INPUT SLOT OF RX BUFFER.
         RXBSTA - LABEL AT START OF RX BUFFER AREA IN MEMORY.

OUTPUTS: RXBIPT - UPDATED TO POINT TO NEXT INPUT SLOT OF RX BUFFER.
         RXBCNT - RX BUFFER CHARACTER COUNT (INCREMENTED).
         TXENBM - MAP OF PREVIOUS DUT TX.ENABLE STATES.
         CARRY - "SUCCESS" FLAG (SET IF BUFFER IS NOT FULL).

CALLING SEQUENCE:  PUT THE ADDRESS OF THE LABEL RXCHRS IN THE VECTOR
                   LOCATION.

COMMENTS:  IF THE RX BUFFER IS FULL UPON ENTRY, THIS ROUTINE ABORTS THE
           PROGRAM.

SUBORDINATE ROUTINES CALLED: RXIEO, TXDSBL.
*****

```

```

RXCHRS::  MOV     R2,-(SP)      ;SAVE CONTENTS OF GPR R2.
2$:      MOV     @RBUFA,R2    ;READ A CHARACTER FROM THE DUT RX FIFO.
        BPL     60$          ;EXIT THE ROUTINE IF THE DATA.VALID BIT IS CLR.

        CMP     RXBCNT,#RXBFUL ;COMPARE BUFFER COUNT WITH BUFFER CAPACITY.
        BLO     4$          ;SKIP ABORT IF BUFFER IS NOT FULL.
        JSR     PC,OOPS      ;ABORT, MUST BE A PROGRAM BUG.
4$:      MOV     R2,@RXBIPT   ;PUT THE CHAR IN THE BUFFER.
        ADD     #2,RXBIPT    ;UPDATE POINTER TO THE NEXT BUFFER SLOT.
        CMP     RXBIPT,#RXBEND ;CHECK IF POINTER SHOULD WRAP AROUND.
        BLO     6$          ;SKIP WRAPAROUND IF POINTER IS NOT AT END.
        MOV     #RXBSTA,RXBIPT ;WRAP INPUT POINTER AROUND.

6$:      INC     RXBCNT      ;COUNT THIS CHARACTER AS BEING IN THE BUFFER.
        CMP     RXBCNT,#RXBDTX ;CHECK FOR BUFFER AT DISABLE TX LEVEL.
        BNE     8$          ;SKIP DISABLING TX IF BUFFER LEVEL NOT CORRECT.
        TST     TXDBLF      ;CHECK STATE OF TX DISABLE FLAG.
        BMI     8$          ;BRANCH IF TRANSMISSION ALREADY DISABLED.
        MOV     R5,-(SP)    ;SAVE THE VALUE OF GPR R5.
        MOV     #MAPLNS,R5  ;SPECIFY THAT ALL LINES SHOULD BE AFFECTED.
        JSR     PC,TXDSBL   ;CLEAR THE TX ENABLES FOR ALL LINES.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 181
 CVDHCA.P11 12-JUL-83 11:44 INTERRUPT SERVICE ROUTINE - RXCHRS -

7471	025254	010537	002250			MOV	R5, TXENBM	:SAVE PREVIOUS TX ENABLE STATES IN STORAGE.
7472	025260	012605				MOV	(SP)+, R5	:RESTORE GPR R5.
7473	025262	012737	100000	002414		MOV	#BIT15, TXDBLF	:PREVENT TX FROM BEING DISABLED AGAIN.
7474								
7475	025270	023727	002624	000100	8\$:	CMP	RXBCNT, #RXBFUL	:CHECK FOR BUFFER FULL CONDITION.
7476	025276	103723				BLO	2\$:LOOP TO READ ANOTHER CHAR IF BUFFER NOT FULL.
7477								
7478	025300	004737	022314			JSR	PC, RXIE0	:BUFFER IS FULL, DISABLE RX INTERRUPTS.
7479								
7480	025304	012602			60\$:	MOV	(SP)+, R2	:RESTORE R2 TO ITS SAVED VALUE.
7481	025306	000002				RTI		

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 182
GLOBAL TRAP SERVICE ROUTINE - TP4RTN -

```

7482 .SBTTL GLOBAL TRAP SERVICE ROUTINE - TP4RTN -
7483 :*****
7484 :* BUS TIME-OUT TRAP (004 TRAP) SERVICE ROUTINE -
7485 :* THIS ROUTINE IS USED DURING THE DEVICE REGISTER ADDRESS ACCESS TEST.
7486 :* IT DETERMINES IF THE 004 TRAP WAS CAUSED BY AN 'EXPECTED' ERROR OR
7487 :* NOT BY EXAMINING THE RETURN PC VALUE ON THE STACK. IF THE TRAP IS
7488 :* UNEXPECTED, THIS ROUTINE JUMPS TO THE NORMAL DIAGNOSTIC SUPERVISOR
7489 :* 004 TRAP HANDLING ROUTINE.
7490 :*
7491 :* INPUTS: SP - POINTS TO THE PC WHERE THE TRAP OCCURED.
7492 :* ADRPTR - LABEL AT THE ADDRESS WHERE 'EXPECTED' TRAPS OCCUR.
7493 :* TP4FLG - 004 TRAP FLAGS.
7494 :*
7495 :* OUTPUTS: TP4FLG - BIT 15 IS SET IF 'EXPECTED' TRAP OCCURED.
7496 :*
7497 :* CALLING SEQUENCE: PUT ADDRESS POINTED TO BY TP4RTN IN 004 VECTOR.
7498 :* OCCURENCE OF 004 TRAP VECTORS TO THIS ROUTINE.
7499 :*
7500 :* COMMENTS: ANY 004 TRAP WHICH OCCURS AT AN ADDRESS OTHER THAN THAT LABELED
7501 :* ADRPTR WILL BE HANDLED BY THE NORMAL 004 TRAP SERVICE ROUTINE.
7502 :*
7503 :* SUBORDINATE ROUTINES CALLED: NONE.
7504 :*****
7505
7506 025310 021627 015464 TP4RTN:: CMP (SP),#ADRPTR ;COMPARE EXPECTED ADR AGAINST TRAP RET PC.
7507 025314 001402 BEQ 2$ ;IF THEY MATCH, CONTINUE THIS ROUTINE.
7508 025316 000177 154732 JMP @TP4VEC ;IF NOT, JUMP TO NORMAL 004 TRAP SERVICE RTN.
7509 025322 052737 100000 002256 2$: BIS #BIT15,TP4FLG ;SET THE 004 TRAP OCCURED FLAG.
7510 025330 000002 RTI ;ALL DONE, GO BACK TO THE TEST.

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 183
INTERRUPT SERVICE ROUTINE - TXDMA -

7511
7512
7513
7514
7515
7516
7517
7518
7519
7520
7521
7522
7523
7524
7525
7526
7527
7528
7529
7530
7531
7532
7533
7534
7535
7536
7537
7538
7539
7540 025332
7541 025332 004537 005232
7542 025336 017701 154640
7543 025342 010100
7544 025344 000402
7545
7546
7547
7548
7549
7550
7551
7552 025346 017701 154630
7553 025352 100033
7554 025354 000301
7555 025356 042701 177760
7556 025362 010104
7557 025364 006304
7558 025366 016405 002332
7559
7560
7561
7562
7563
7564
7565
7566 025372 026464 003410 003350

```

.SBTTL INTERRUPT SERVICE ROUTINE - TXDMA -
:++ *****
: *
: * - DMA TRANSMIT INTERRUPT SERVICE ROUTINE -
: * THIS ROUTINE EXECUTES IN RESPONSE TO AN INTERRUPT CAUSED BY THE DUT
: * TX.ACTION BIT BECOMING ACTIVE. THIS ROUTINE INITIATES THE TX OF A
: * NEW DMA BUFFER OF CHARACTERS OR SETS THE TX DONE FLAG FOR THE CORRECT
: * LINE IF TX IS COMPLETE ON THAT LINE.
: *
: * INPUTS: BITTBL - LABEL OF TABLE OF WORDS EACH WITH A BIT SET.
: *          CNCNTB - BASE OF # OF CHARS TO TX/RX TABLE.
: *          CSRA - CONTAINS THE ADDRESS OF THE DUT CSR.
: *          DPENDB - BASE OF THE DATA PATTERN END TABLE (ENTRY PER LINE).
: *          DPLENB - BASE OF THE DATA PATTERN LENGTH TABLE.
: *          IESTAT - PRESERVED STATES OF THE DUT INTERRUPT ENABLE BITS.
: *          TXCNTB - LABEL AT BASE OF THE TX CHARACTER COUNTER TABLE.
: *          TXPTRB - LABEL AT BASE OF THE TX DATA PATTERN POINTERS TABLE.
: *
: * OUTPUTS: TXCNTX - COUNTERS INCREMENTED FOR LINES ON WHICH CHARS SENT.
: *          TXDNF - TX DONE FLAGS SET FOR LINES WHICH HAVE SENT ALL CHARS.
: *          TXINTF - TX INT FLAGS (BIT SET IF DMA.HO FOUND SET ON LINE).
: *
: * CALLING SEQUENCE: PUT THE ADDRESS OF THE LABEL TXDMA IN THE VECTOR
: *                   LOCATION.
: *
: * COMMENTS:
: *
: * SUBORDINATE ROUTINES CALLED: DODMA.
:-- *****
TXDMA:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
          JSR R5,PREG05 ;CALL REGISTER SAVE SUBRT.
          MOV @CSRA,R1 ;READ THE CONTENTS OF THE DUT CSR.
          MOV R1,R0 ;SAVE INITIAL CONTENTS OF IND.ADR.REG FIELD.
          BR 4$ ;BRANCH TO SKIP DOUBL READING OF DUT CSR.
: +
: READ THE CONTENTS OF THE DUT CSR. THIS WILL CLEAR THE TX.ACTION CSR BIT.
: IF TX.ACTION IS NOT SET, EXIT THIS ROUTINE.
: DETERMINE THE LINE FOR WHICH THE TX.ACTION WAS SET.
: CALCULATE AN OFFSET FOR USE IN ACCESSING TABLES (2 TIMES THE LINE NUMBER).
: GET THE BIT MAP OF THIS LINE.
: -
2$: MOV @CSRA,R1 ;READ THE CONTENTS OF THE DUT CSR.
4$: BPL 60$ ;EXIT ROUTINE IF TX.ACTION IS CLEAR.
    SWAB R1 ;CALCULATE THE LINE NUMBER OF THE LINE WHICH IS
    BIC #177760,R1 ; ASSOCIATED WITH THE TX.ACTION.
    MOV R1,R4 ;CALCULATE AN OFFSET FOR USE IN ACCESSING
    ASL R4 ; LINE COUNTER AND POINTER IN TABLES.
    MOV BITTBL(R4),R5 ;GET THE BIT MAP OF THIS LINE.
: +
: GET THE TX CHARACTER COUNTER FOR THIS LINE.
: IF ALL THE CHARACTERS HAVE BEEN SENT FOR THIS LINE:
: SET THE TX DONE FLAG FOR THIS LINE.
: DON'T SEND A CHAR TO THE LINE (NO MORE TX.ACTIONS ON THIS LINE).
: LOOP TO CHECK THE TX.ACTION FOR ANOTHER LINE.
: -
CMP TXCNTB(R4),CNCNTB(R4) ;COMPARE # CHARS SENT AND TX COUNT.

```

CV
CV

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 184
INTERRUPT SERVICE ROUTINE - TXDMA -

```

7567 025400 103403          BLO      6$          ;GO TO SEND A CHAR IF NOT ALL CHARS SENT.
7568 025402 050537 002410  BIS      R5, TXDNF    ;SET THIS LINE'S TX DONE FLAG.
7569 025406 000757          BR       2$          ;LOOP TO CHECK TX.ACTION AGAIN.
7570
7571      ;+ START THE DMA OF THE NEXT BUFFER (DATA PATTERN) ON THIS LINE.
7572      ; GET THE DATA PATTERN LENGTH FOR THIS LINE.
7573      ; GET THE START ADDRESS OF THE DATA PATTERN.
7574
7575 025410 016403 003110 6$:      MOV      DPLENB(R4),R3 ;PASS DATA PATTERN LENGTH FOR LINE TO DODMA.
7576 025414 016402 003250      MOV      TXPTRB(R4),R2 ;PASS THE TX START ADR TO DODMA.
7577
7578      ;+ WRITE DMA PARAMETERS TO THE DUT.
7579
7580 025420 004737 015662      JSR      PC,DODMA
7581 025424 103403          BCS      8$          ;SKIP ERROR IF DODMA WAS SUCCESSFUL.
7582
7583      ;+ SET THE PROPER BIT OF THE TX INTERRUPT FLAGS TO INDICATE THE LINE ERROR.
7584
7585 025426 050537 002252      BIS      R5, TXINTF   ;INDICATE THE ERROR.
7586 025432 000402          BR       10$         ;SKIP UPDATING POINTERS AND COUNTERS.
7587
7588      ;+ UPDATE THE TX CHARACTER FOR THIS LINE.
7589      ; UPDATE THE TX BUFFER POINTER FOR THIS LINE.
7590
7591 025434 060364 003410 8$:      ADD      R3, TXCNTB(R4) ;ADD THE DATA PAT LENGTH TO THE TX COUNT.
7592
7593      ;+ LOOP TO CHECK THE TX.ACTION BIT FOR ANOTHER LINE.
7594
7595 025440 000742 10$:     BR       2$          ;LOOP BACK TO CHECK TX.ACTION BIT AGAIN.
7596
7597 025442 013701 002234 60$:     MOV      IESTAT,R1      ;GET THE PRESENT STATES OF TX.IE & RX.IE BITS.
7598 025446 042700 177760      BIC      #177760,R0    ;GET SAVED IND.ADR.REG FIELD BITS.
7599 025452 050001          BIS      R0,R1        ;COMBINE IND.ADR.REG FIELD BITS WITH IE BITS.
7600 025454 010177 154522      MOV      R1,@CSRA     ;RESTORE THE DUT CSR IND.ADR.REG FIELD.
7601 025460          PASS          ;RESTORE GPRS.
7602 025460 004736          JSR      PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
7603 025462 000002          RTI

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 185
INTERRUPT SERVICE ROUTINE - TXSCHR -

7604
7605
7606
7607
7608
7609
7610
7611
7612
7613
7614
7615
7616
7617
7618
7619
7620
7621
7622
7623
7624
7625
7626
7627
7628
7629
7630
7631
7632
7633
7634
7635
7636
7637
7638
7639
7640
7641
7642
7643
7644
7645
7646
7647
7648
7649
7650
7651
7652
7653
7654
7655
7656
7657
7658
7659

025464
025464 004537 005232
025470 017701 154506
025474 010105
025476 042705 137660
025502 010546
025504 005005
025506 005701
025510 000402

025512 017701 154464
025516 100051
025520 010102
025522 000302
025524 042702 177760
025530 010203

```

.SBTTL INTERRUPT SERVICE ROUTINE - TXSCHR -
:++ *****
: * - SINGLE CHARACTER MODE TRANSMIT INTERRUPT SERVICE ROUTINE -
: * THIS ROUTINE EXECUTES IN RESPONSE TO AN INTERRUPT CAUSED BY THE DUT
: * TX.ACTION BIT BECOMING ACTIVE. THIS ROUTINE SENDS THE NEXT TX CHAR
: * OR SETS THE TX DONE FLAG FOR THE CORRECT LINE IF TX IS COMPLETE ON
: * THAT LINE.
: *
: * INPUTS: ACTLNS - BIT MAP OF ACTIVE DUT LINES.
: * BITTBL - LABEL OF TABLE OF WORDS EACH WITH A BIT SET.
: * CHCNTB - BASE OF # OF CHARS TO TX/RX TABLE.
: * CSRA - CONTAINS THE ADDRESS OF THE CSR.
: * DPENDB - BASE OF THE DATA PATTERN END TABLE (ENTRY PER LINE).
: * DPLENB - BASE OF THE DATA PATTERN LENGTH TABLE.
: * IACBIT - BIT MASK OF INACTIVE TX/RX BITS.
: * IBM - INACTIVE BITS MASK (REFLECTING BITS PER CHAR).
: * TXCHRA - CONTAINS THE ADDRESS OF THE DUT TXCHAR REGISTER.
: * TXCNTB - LABEL AT BASE OF TX CHARACTER COUNTERS TABLE.
: * TXPTRB - LABEL AT THE BASE ADDRESS OF THE TX POINTERS TABLE.
: *
: * OUTPUTS: CSR - DUT CSR IND.ADR.REG FIELD IS DESTROYED.
: * TXCHAR - DUT TXCHARS HAVE WORDS WRITTEN TO THEM.
: * TXCNTX - COUNTERS INCREMENTED FOR LINES ON WHICH CHARS SENT.
: * TXDONF - TX DONE FLAGS SET FOR LINES WHICH HAVE SENT ALL CHARS.
: * TXPTRB - EACH POINTER IN TABLE POINTS TO NEXT TX CHAR FOR LINE.
: *
: * CALLING SEQUENCE: PUT THE ADDRESS OF THE LABEL TXSCHR IN THE VECTOR
: * LOCATION.
: *
: * COMMENTS:
: * SUBORDINATE ROUTINES CALLED: NONE.
:-- *****
TXSCHR:: SAVE ;SAVE CONTENTS OF GPRS R0 THRU R5.
;R5,PREG05 ;CALL REGISTER SAVE SUBRT.
JSR ;READ THE CONTENTS OF THE DUT CSR.
MOV @CSRA,R1 ;SAVE THE CONTENTS OF THE DUT IND.ADR.REG
MOV R1,R5 ;FIELD FOR RESTORATION BEFORE RETURN.
BIC #137660,R5 ;SAVE THE DUT IN.ADD FIELD ON THE STACK.
MOV R5,-(SP) ;CLEAR CHARACTER TRANSMISSION COUNTER.
CLR R5 ;SET FLAG FOR TEST AFTER THE FOLLOWING BRANCH.
TST R1 ;GO HANDLE THE LINE THAT GOT THE TX.ACTION.
BR 4$

: * READ THE CONTENTS OF THE DUT CSR. THIS WILL CLEAR THE TX.ACTION CSR BIT.
: * IF TX.ACTION IS NOT SET, EXIT THIS ROUTINE.
: * DETERMINE THE LINE FOR WHICH THE TX.ACTION WAS SET.
: * CALCULATE AN OFFSET FOR USE IN ACCESSING TABLES (2 TIMES THE LINE NUMBER).
: * DETERMINE THE STATES OF THE DUT CSR INTERRUPT ENABLE BITS.
:--
2$: MOV @CSRA,R1 ;READ THE CONTENTS OF THE DUT CSR.
4$: BPL 60$ ;EXIT ROUTINE IF TX.ACTION IS CLEAR.
MOV R1,R2 ;CALCULATE THE LINE NUMBER
SWAB R2 ;OF THE LINE WHICH IS
BIC #177760,R2 ;ASSOCIATED WITH THE TX.ACTION.
MOV R2,R3 ;CALCULATE AN OFFSET FOR USE IN ACCESSING

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 186
INTERRUPT SERVICE ROUTINE - TXSCHR -

```

7660 025532 006303          ASL   R3           ; LINE COUNTER AND POINTER IN TABLES.
7661 025534 042701 137677  BIC   #137677,R1   ;GET BIT MASK OF INTERRUPT ENABLE STATES.
7662
7663      ;+ GET THE TX CHARACTER COUNTER FOR THIS LINE.
7664      ; IF ALL THE CHARACTERS HAVE BEEN SENT FOR THIS LINE:
7665      ; SET THE TX DONE FLAG FOR THIS LINE.
7666      ; DON'T SEND A CHAR TO THE LINE (NO MORE TX.ACTIONS ON THIS LINE).
7667      ; LOOP TO CHECK THE TX.ACTION FOR ANOTHER LINE.
7668
7669 025540 026363 003410 003350  CMP   TXCNTB(R3),CHCNTB(R3) ;COMPARE TX CHAR COUNT AND # TO TX.
7670 025546 103404          BLO   6$           ;GO TO SEND A CHAR IF NOT ALL CHARS SENT.
7671 025550 056337 002332 002410  BIS   BITTBL(R3),TXDNF     ;SET THIS LINE'S TX DONE FLAG.
7672 025556 000757          BR    4$           ;LOOP TO CHECK TX.ACTION AGAIN.
7673
7674      ;+ SEND THE NEXT CHAR TO THE SPECIFIED LINE.
7675      ; SET UP THE IND.ADR.REG FIELD OF THE DUT CSR USING THE PREVIOUSLY READ
7676      ; STATES OF THE INTERRUPT ENABLE BITS.
7677      ; FETCH THE CORRECT CHARACTER FROM THE DATA PATTERN.
7678      ; UPDATE THE DATA PATTERN POINTER FOR THIS LINE USING WRAPAROUND.
7679      ; MASK OUT INACTIVE DATA BITS AND SEND THE CHARACTER.
7680      ; COUNT THE CHARACTER ON THE TX CHAR COUNTER FOR THE LINE.
7681      ; DECREMENT THE FIFO SLACK COUNT TO RESERVE ROOM FOR THIS CHAR IN THE FIFO.
7682      ; EXIT IF A MAXIMUM OF 8 CHARACTERS HAVE BEEN TRANSMITTED, IE.GIVE RECEPTION
7683      ; A CHANCE TO REMOVE CHARACTERS FROM THE FIFO.
7684
7685 025560 050201          6$:   BIS   R2,R1           ;SET UP THE IND.ADR.REG FIELD OF THE DUT CSR
7686 025562 010177 154414  MOV   R1,@CSRA       ; WITHOUT AFFECTING THE INTERRUPT ENABLES.
7687 025566 016304 003250  MOV   TXPTRB(R3),R4  ;FETCH THE TX POINTER FOR THIS LINE.
7688 025572 112400          MOVB  (R4)+,R0        ;GET THE NEXT CHAR FOR THIS LINE.
7689 025574 020463 003050  CMP   R4,DPENDB(R3)  ;COMPARE POINTER WITH END OF DATA PATTERN.
7690 025600 103402          BLO   8$           ;SKIP RESETTING OF POINTER IF NOT PAST END.
7691 025602 166304 003110  SUB   DPLENB(R3),R4  ;WRAP POINTER AROUND TO BEGINNING OF PATTERN.
7692 025606 010463 003250  8$:   MOV   R4,TXPTRB(R3) ;UPDATE THE TX POINTER FOR THIS LINE.
7693 025612 043700 002226  BIC   IBM,R0         ;CLEAR UNUSED BITS OF THE TX CHAR WORD.
7694 025616 052700 100000  BIS   #BIT15,R0      ;SET THE TX.DATA.VALID BIT OF TX CHAR WORD.
7695 025622 010077 154356  MOV   R0,@TXCHA     ;SEND THE CHAR TO THE DUT.
7696 025626 005205          INC   R5             ;INCREMENT TX CHAR COUNT.
7697 025630 005263 003410  INC   TXCNTB(R3)     ;INCREMENT THE TX CHAR COUNT FOR THIS LINE.
7698
7699      ;+ LOOP TO CHECK THE TX.ACTION BIT FOR ANOTHER LINE.
7700
7701 025634 020527 000010  CMP   R5,#NUMLNS     ;CHECK IF MAX NUMBER OF CHAR HAVE BEEN TX'D.
7702 025640 103724          BLO   2$           ;LOOP BACK TO CHECK TX.ACTION BIT AGAIN.
7703 025642 012677 154334  60$:  MOV   (SP)+,@CSRA   ;RESTORE THE IND.ADR.REG FIELD OF THE DUT CSR.
7704 025646          PASS          ;RESTORE GPRS.
7705 025646 004736          JSR   PC,@(SP)+    ;RETURN TO PREG05 SUBRT.
7706 025650 000002          RTI

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 187
INTERRUPT SERVICE ROUTINE - TXSCHR -

7707
7708
7709
7710
7711
7712
7713
7714
7715
7716
7717
7718
7719
7720
7721
7722
7723
7724
7725
7726
7727

025652
025652

025652
025652 000167
025654 000000

025656
025656
025656 104425

.SBTTL REPORT CODING SECTION

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

BGNRPT

LSRPT::

EXIT RPT

.WORD JSJMP
.WORD L10017-2-.

.EVEN

ENDRPT

L10017:
TRAP CSRPT

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 188
PROTECTION TABLE

.SBTTL PROTECTION TABLE

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

7728
7729
7730
7731
7732
7733
7734
7735
7736
7737
7738
7739
7740
7741
7742
7743
7744

025660
025660

177777
177777
177777

BGNPROT

-1
-1
-1

ENDPROT

L\$PROT::

:OFFSET INTO P-TABLE FOR CSR ADDRESS
:OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
:OFFSET INTO P-TABLE FOR DRIVE NUMBER

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 189
PROTECTION TABLE

```

7745
7746
7747      .SBTTL  INITIALIZE SECTION
7748      :++
7749      :*****
7750      :      THIS SECTION CONTAINS THE CODE WHICH IS PERFORMED AT THE BEGINNING OF
7751      :      EACH PASS OR AFTER A CONTINUE COMMAND.
7752      :      THIS CODE PERFORMS THE FOLLOWING ACTIONS:
7753      :
7754      :      MOVES THE INFORMATION HELD IN THE HARDWARE P-TABLE INTO THE GLOBAL
7755      :      DATA AREA.
7756      :
7757      :*****
7758      :--
7759      025666      BGNINIT
7760      025666      L$INIT::
7761      :SEE IF PROGRAM JUST STARTED, BR IF YES
7762      025666      READEF  #EF.START
7763      025666      012700 000040      MOV      #EF.START,RO
7764      025672      104447      TRAP     C$REFG
7765      025674      BCOMPLETE      NEWSTA
7766      025674      103416      BCS      NEWSTA
7767      :SEE IF PROGRAM JUST RESTARTED, BR IF YES
7768      025676      READEF  #EF.RESTART
7769      025676      012700 000037      MOV      #EF.RESTART,RO
7770      025702      104447      TRAP     C$REFG
7771      025704      BCOMPLETE      NEWRES
7772      025704      103556      BCS      NEWRES
7773      :SEE IF THIS IS A NEW PASS, BR IF YES
7774      025706      READEF  #EF.NEW
7775      025706      012700 000035      MOV      #EF.NEW,RO
7776      025712      104447      TRAP     C$REFG
7777      025714      BCOMPLETE      NEWPAS
7778      025714      103555      BCS      NEWPAS
7779      :SEE IF PROGRAM WAS JUST CONTINUED
7780      025716      READEF  #EF.CONTINUE
7781      025716      012700 000036      MOV      #EF.CONTINUE,RO
7782      025722      104447      TRAP     C$REFG
7783      025724      BNCOMPLETE      GETPRM
7784      025724      103161      BCC      GETPRM
7785      025726      000137 026510      JMP      ENDIT
7786      025732      NEWSTA:
7787      025732      BRESET      ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
7788      025732      104433      TRAP     C$RESET
7789      :+
7790      :SET UP FOR LINE TIME CLOCK INTERRUPTS.
7791      :--
7792      025734      CLOCK  L,R1      ;GET THE CLOCK PARAMETERS.
7793      025734      012700 000114      MOV      #'L,RO
7794      025740      104462      TRAP     C$CLCK
7795      025742      010001      MOV      RO,R1
7796      025744      012137 002260      MOV      (R1)+,CLKCSR      ;STORE CLOCK CSR ADDRESS.
7797      025750      012137 002262      MOV      (R1)+,CLKBRL     ;STORE CLOCK BUS REQ INT LEVEL.
7798      025754      012137 002264      MOV      (R1)+,CLKVEC     ;STORE CLOCK INTERRUPT VECTOR.
7799      025760      012137 002266      MOV      (R1)+,CLKHRZ     ;STORE CLOCK FREQUENCY.
7800      025764      023727 002266 000062      CMP      CLKHRZ,#50.      ;TEST FOR 50HZ LINE FREQUENCY.
    
```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 190
 CVDHCA.P11 12-JUL-83 11:44 INITIALIZE SECTION

```

7801 025772 001004          BNE      2$          ;BRANCH IF CLOCK IS NOT 50HZ.
7802 025774 012737 000024 002300  MOV      #20.,MSTICK ;INDICATE 20MS PER CLOCK TICK.
7803 026002 000403          BR       4$
7804 026004 012737 000021 002300 2$:  MOV      #17.,MSTICK ;INDICATE 17 MS PER CLOCK TICK.
7805 026012          4$:  SETVEC  CLKVEC,#CLKINT,PRI06 ;INITIALIZE CLOCK INTERRUPT VECTOR.
7806 026012 013746 000300          MOV      PRI06,-(SP)
7807 026016 012746 025074          MOV      #CLKINT,-(SP)
7808 026022 013746 002264          MOV      CLKVEC,-(SP)
7809 026026 012746 000003          MOV      #3,-(SP)
7810 026032 104437          TRAP    C$$VEC
7811 026034 062706 000010          ADD     #10,SP
7812 026040 013700 002266          MOV     CLKHRZ,R0 ;INITIALIZE THE BREAK COUNT
7813 026044 006300          ASL     R0 ; TO CAUSE A BREAK
7814 026046 010037 002276          MOV     R0,BCOUNT ; EVERY 2 SECONDS.
7815 026052          SETPRI #PRI05 ;ALLOW CLOCK INTERRUPTS DISABLE OTHERS.
7816 026052 012700 000240          MOV     #PRI05,R0
7817 026056 104441          TRAP   C$$PRI
7818
7819          :+
7820          : ENABLE THE LINE TIME CLOCK (LTC) CHECKING TO MAKE SURE THAT THE CSR
7821          : IS ACCESSABLE.
7822          : FIRST SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:
7823          : -
7823 026060 013737 000004 002254  MOV     4,TP4VEC ;SAVE THE EXISTING 004 TRAP VECTOR.
7824 026066 012737 025310 000004  MOV     #TP4RTN,4 ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
7825          :+
7826          : ENABLE LTC CHECKING FOR 004 TRAP IN CASE CSR IS NOT THERE.
7827          : -
7828 026074 005037 002256          CLR     TP4FLG ;CLEAR THE 004 TRAP FLAG.
7829 026100 012737 000100 002240  MOV     #BIT6,WORD1 ;SET UP TO SET BIT6 OF THE LTC CSR.
7830 026106 012700 002240          MOV     #WORD1,R0 ;SET UP WORD1 AS THE CKTRAP MOVE SOURCE.
7831 026112 013701 002260          MOV     CLKCSR,R1 ;SET UP LTC CSR AS DESTINATION FOR CKTRAP MOVE.
7832 026116 004737 015452          JSR    PC,CKTRAP ;MOVE AND CHECK FOR TRAP.
7833 026122 013737 002254 000004  MOV     TP4VEC,4 ;RESTORE THE NORMAL 004 TRAP VECTOR.
7834 026130 103403          BCS    6$ ;IF NO TRAP, LTC IS THERE SO CONTINUE.
7835 026132 005037 002266          CLR     CLKHRZ ;CLEAR LTC FREQUENCY WORD TO INDICATE NO LTC.
7836 026136 000402          BR     8$ ;BYPASS THE FOLLOWING CALIBRATION PROCEDURES.
7837          :+
7838          : CALIBRATE THE DELAY ROUTINE MILLI-SECOND DELAY COUNT VALUE.
7839          : -
7840 026140 004737 014222 6$:  JSR    PC,CALMSL
7841          :+
7842          : CHECK FOR MEMORY MANAGEMENT PRESENT ON THIS MACHINE.
7843          : IF MEM MGT IS PRESENT, DISABLE IT.
7844          : -
7845 026144 013737 000004 002254 8$:  MOV     4,TP4VEC ;SAVE THE EXISTING 004 TRAP VECTOR.
7846 026152 012737 025310 000004  MOV     #TP4RTN,4 ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
7847 026160 005037 002256          CLR     TP4FLG ;CLEAR THE 004 TRAP FLAG.
7848 026164 005037 002240          CLR     WORD1 ;PREPARE TO CLEAR THE MEM MGT SRO REGISTER.
7849 026170 012700 002240          MOV     #WORD1,R0 ;SELECT CLEARED WORD AS CKTRAP RTN SOURCE.
7850 026174 013701 002304          MOV     MMSRO,R1 ;SELECT MEM MGT SRO REGISTER AS DESTINATION.
7851 026200 005037 002306          CLR     MMPRES ;INDICATE NO MEM MGT PRESENT IN CASE IT ISN'T.
7852 026204 005037 002310          CLR     MMENAB ;INDICATE MEM MGT IS NOT ENABLED.
7853 026210 004737 015452          JSR    PC,CKTRAP ;CLEAR THE MEM MGT SRO REG AND CHECK FOR TRAP.
7854 026214 013737 002254 000004  MOV     TP4VEC,4 ;RESTORE THE NORMAL 004 TRAP VECTOR.
7855 026222 103003          BCC    10$ ;SKIP INDICATING MEM MGT PRESENT IF IT ISN'T.
7856 026224 012737 000001 002306  MOV     #1,MMPRES ;INDICATE THAT MEM MGT IS PRESENT.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(.052) 12-JUL-83 13:56 PAGE 191
 CVDHCA.P11 12-JUL-83 11:44 INITIALIZE SECTION

```

7857 026232 005037 002236      10$: CLR PASCNT      ;CLR COUNTER USED IN REPORTING ROM VERSION #.
7858 026236 000137 026250      JMP NEWPAS      ;SKIP AROUND THE BUS RESET, IT'S BEEN DONE.
7859
7860 026242      NEWRES: BRESET      ;RESET THE BUS TO PREVENT ILLEGAL INTERRUPTS.
7861 026242 104433      TRAP CSRESET
7862 026244 005037 002236      CLR PASCNT      ;CLR COUNTER USED IN REPORTING ROM VERSION #.
7863
7864 026250      NEWPAS:
7865 026250 012737 177777 002200  MOV #-1,UNITN    ;RESET LOGICAL DEVICE TO -1
7866
7867      ;+
7868      ; INCREMENT THE PASS COUNTER, CORRECT FOR ANY OVERFLOW.
7869      ; THIS COUNTER IS USED IN THE ROM VERSION TEST.
7870
7870 026256 005237 002236      INC PASCNT      ;INCREMENT THE PASS COUNTER.
7871 026262 001002      BNE GETPRM     ;BRANCH IF WE HAVE NOT YET! OVERFLOWED.
7872 026264 005337 002236      DEC PASCNT     ;SET PASS COUNT TO 177777 OCTAL.
7873
7874      ; GET THE HARDWARE PARAMETERS FOR THIS UNIT.
7875      GETPRM:
7876 026270 005237 002200      INC UNITN      ;INCREMENT LOGICAL DEVICE NUMBER
7877 026274 023737 002200 002012  CMP UNITN,LSUNIT ;SEE IF MAXIMUM UNIT NO. EXCEEDED
7878 026302 002362      BGE NEWPAS     ;BR IF YES
7879
7880      GPHARD UNITN,R1      ;GET P-TABLE POINTER INTO R1
7881 026304 013700 002200      MOV UNITN,R0
7882 026310 104442      TRAP CS$GPHRD
7883 026312 010001      MOV R0,R1
7884 026314      BCOMPLETE 30$      ;BR IF DEVICE AVAILABLE
7885 026314 103401      BCS 30$
7886 026316 000764      BR GETPRM     ;SKIP THIS DEVICE
7887
7888
7889      ;***** HARDWARE PARAMETER MOVING CODE *****
7890 026320 012137 002202      30$: MOV (R1)+,CSRA  ;STORE DHV-11 CSR ADDRESS IN DEV.REG.ADDRESS TABLE
7891 026324 012102      MOV (R1)+,R2   ;GET THE RX INTERRUPT VECTOR ADDRESS.
7892 026326 010237 002170      MOV R2,RXVECA ;STORE RX INT VECTOR ADDRESS.
7893 026332 062702 000004      ADD #4,R2     ;CALCULATE TX INTERRUPT VECTOR ADDRESS.
7894 026336 010237 002172      MOV R2,TXVECA ;STORE TX INT VECTOR ADDRESS.
7895 026342 012137 002174      MOV (R1)+,ACTLNS ;STORE DHV-11 ACTIVE LINE BIT MAP
7896 026346 012702 000377      MOV #MAPLNS,R2 ;GET THE BIT MAP FOR ALL LINES.
7897 026352 005102      COM R2       ;GET A BIT MAP OF NON-EXISTANT LINES.
7898 026354 040237 002174      BIC R2,ACTLNS ;CLEAR NON-EXISTANT LINES FROM ACTLNS.
7899 026360 112137 002176      MOVB (R1)+,LOPBCK ;STORE DHV-11 LOOPBACK MODE
7900 026364 112137 002177      MOVB (R1)+,BRLEVL ;STORE DHV-11 INTERUPT BUS REQUEST LEVEL
7901
7902      ;+
7903      ; CALCULATE DEVICE REGISTER ADDRESSES,AND PUT THEM IN THE
7904      ; DEVICE REGISTER ADDRESS TABLE.
7905
7905 026370 013701 002202      MOV CSRA,R1   ;COPY CSR ADDRESS
7906 026374 005201      INC R1       ;INCREMENT CSR ADDRESS
7907 026376 005201      INC R1       ; COPY BY 2.
7908 026400 012703 000007      MOV #7,R3    ;SET UP REGISTER COUNT
7909 026404 012702 002204      MOV #RBUFA,R2 ;GET LOCATION WHERE RBUF ADDRESS GOES IN TABLE
7910 026410 010122      12$: MOV R1,(R2)+  ;STORE REGISTER ADDRESS IN TABLE
7911 026412 005201      INC R1       ;INCREMENT REGISTER ADDRESS
7912 026414 005201      INC R1       ; BY 2,FOR THE NEXT DEVICE REGISTER.

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 192
INITIALIZE SECTION

```

7913 026416 005303          DEC    R3          ;DECREMENT REGISTER COUNT
7914 026420 001373          BNE    12$        ;LOOP IF NOT DONE
7915
7916          ;+
7917          ; INITIALISE THE BMP CODE QUEUE.
7918          ;-
7919 026422 012700 002420      MOV    #BMPQCB,R0  ;GET THE START ADDRESS OF THE QUEUE.
7920 026426 012701 002620      MOV    #BMPQCE,R1  ;GET THE END ADDRESS OF THE QUEUE.
7921 026432 010037 002416      MOV    R0,BMPQCP   ;SET THE POINTER TO THE START OF THE QUEUE.
7922 026436 005020          14$: CLR    (R0)+       ;CLEAR OUT THE CONTENTS OF THE QUEUE.
7923 026440 020001          CMP    R0,R1       ;CHECK IF END OF QUEUE HAS BEEN REACHED.
7924 026442 103775          BLO    14$        ;LOOP IF NOT ALL DONE.
7925
7926          ;+
7927          ; REPORT THE UNIT NUMBER IF THE SOFTWARE P-TABLE QUESTION WAS ANSWERED YES,
7928          ; AND THE MAXIMUM UNIT NUMBER IS GREATER THAN 1.
7929 026444 032737 000020 002164 BIT    #BIT4,OPTION ;CHECK IF THE QUESTION WAS ANSWERED YES.
7930 026452 001416          BEQ    16$        ;SKIP REPORTING UNIT NUMBER IF IT IS DISABLED.
7931 026454 023727 002012 000001 CMP    LSUNIT,#1   ;CHECK MAXIMUM NUMBER OF UNITS SELECTED.
7932 026462 003412          BLE    16$        ;DO NOT REPORT UNIT NUMBER IF MAX NUMBER < 1.
7933 026464          PRINTF #MFUNIT,UNITN ;REPORT UNIT NUMBER.
7934 026464 013746 002200          MOV    UNITN,-(SP)
7935 026470 012746 005332          MOV    #MFUNIT,-(SP)
7936 026474 012746 000002          MOV    #2,-(SP)
7937 026500 010600          MOV    SP,R0
7938 026502 104417          TRAP  C$PNTF
7939 026504 062706 000006          ADD    #6,SP
7940 026510          16$:
7941
7942 026510          ENDIT:
7943          ;+
7944          ; SET THE PROCESSOR PRIORITY TO ALLOW LTC INTERRUPTS BUT NOT OTHERS.
7945          ;-
7946 026510          SETPRI #PRI07          ;SET PROCESSOR PRIORITY TO 5.
7947 026510 012700 000340          MOV    #PRI07,R0
7948 026514 104441          TRAP  C$SPRI
7949
7950          ENDINIT
7951 026516          L10021:
7952 026516 104411          TRAP  C$INIT

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 193
INITIALIZE SECTION

7953
7954
7955
7956
7957
7958
7959
7960
7961
7962
7963
7964
7965
7966
7967
7968
7969
7970
7971

026520
026520

026520
026520
026520 104461

.SBTTL AUTODROP SECTION

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

BGNAUTO

LSAUTO::

ENDAUTO

L10022: TRAP CSAUTO

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 194
AUTODROP SECTION

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

026522
026522

026522 005737 002222
026526 001401
026530
026530 104433
026532
026532 104432
026534 000002

026536
026536
026536 104412

.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
BEQ 2\$
BRESET

:DID WE GET HERE BY CTRL-C FROM TEST?
:CTRL-C FROM TEST? NO, SKIP BUS RESET.
:YES, CLR ANY DMAS OR OUTSTANDING INTERRUPTS.
TRAP C\$RESET

2\$:

EXIT CLN

TRAP C\$EXIT
.WORD L10023-

.EVEN

ENDCLN

L10023: TRAP C\$CLEAN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 195
CLEANUP CODING SECTION

8000
8001
8002
8003
8004
8005
8006
8007
8008
8009
8010
8011
8012
8013
8014
8015
8016
8017
8018
8019
8020
8021
8022
8023
8024
8025
8026
8027
8028
8029
8030
8031
8032
8033
8034
8035
8036
8037
8038

026540
026540
026540
026542 010046
026542 012746 026564
026546 012746 000002
026552 010600
026554 104417
026556 062706 000006
026562 000427

026564 040445 052440 044516
026572 022524 033104 040445
026600 042040 047522 050120
026606 042105 043040 047522
026614 020115 052506 052122
026622 042510 020122 042524
026630 052123 047111 027107
026636 047045 000
026642
026642
026642 000167
026644 000000

026646
026646
026646 104453

.SBTTL DROP UNIT SECTION

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

BGNDU

PRINTF #DROP,R0

LSDU::
;REPORT UNIT THAT HAS BEEN DROPPED.

MOV R0,-(SP)
MOV #DROP,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #6,SP

BR EDROP

;BRANCH AROUND THE MESSAGE.

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

.EVEN

EDROP:

EXIT DU

.WORD JSJMP
.WORD L10024-2-

ENDDU

L10024:

TRAP C\$DU

B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
[
\
]
^
_
`
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z
{|
~

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 197
HARDWARE TEST - ADRA -

```

8062 .SBTTL HARDWARE TEST - ADRA -
8063 :++
8064 :*****
8065 : - REGISTER ADDRESS TEST -
8066 :
8067 : THIS TEST VERIFIES THAT THE Q-BUS CAN READ AND WRITE TO THE DHV11
8068 : DEVICE REGISTERS. IF THE DHV11 DOES NOT RESPOND TO THE ACCESS
8069 : ATTEMPTS (IF THE DHV11 IS AT THE WRONG ADDRESS, FOR EXAMPLE) THE
8070 : 004 BUS TIME-OUT TRAP IS DETECTED BY THIS ROUTINE AND AN ERROR
8071 : IS REPORTED.
8072 :
8073 :*****
8074 :--
8075
8076 026656 BGNTST
8077 026656 T1::
8078 000001 TNUM == 1 ;THIS TEST MUST ALWAYS BE INCLUDED AS TEST 1.
8079 026656 012737 000001 002224 MOV #TNUM,TSTNUM ;SET THE TEST NUMBER TO 1.
8080 026664 012737 177777 002222 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
8081
8082 :+
8083 : SET UP TO CATCH ANY 004 TRAPS WHICH OCCUR:
8084 :
8085 026672 013737 000004 002254 MOV 4,TP4VEC ;SAVE THE EXISTING 004 TRAP VECTOR.
8086 026700 012737 025310 000004 MOV #TP4RTN,4 ;SET 004 TRAP VECTOR TO OUR SERVICE RTN ADR.
8087 026706 005005 CLR R5 ;CLEAR THE ERROR FLAGS.
8088
8089 :+
8090 : SET UP FOR THE INITIAL ITERATION OF THE TEST LOOP:
8091 :
8092 026710 005004 CLR R4 ;CLEAR THE LINE COUNTER.
8093
8094 :+
8095 : HERE BEGINS THE LOOP TO TEST THE REGISTERS FOR A LINE.
8096 : FIRST TEST THE CSR AND SET THE IND.ADR.REG (I.A.R) FIELD.
8097 :
8098 026712 005037 002256 2$: CLR TP4FLG ;CLEAR THE 004 TRAP FLAG.
8099 026716 013700 002202 MOV CSRA,R0 ;SET UP CSR AS THE CKTRAP MOVE SOURCE.
8100 026722 012701 027136 MOV #52$,R1 ;SET UP DESTINATION LOCATION FOR CKTRAP MOVE.
8101 026726 004737 015452 JSR PC,CKTRAP ;MOVE AND CHECK FOR TRAP.
8102 026732 103402 BCS 4$ ;IF NO TRAP, BYPASS ERROR.
8103 026734 052705 100001 BIS #100001,R5 ;SET FATAL READ ERROR FLAGS.
8104 026740 042737 000017 027136 4$: BIC #17,52$ ;CLEAR THE I.A.R FIELD OF THE CSR DATA.
8105 026746 050437 027136 BIS R4,52$ ;OR IN THE LINE COUNTER TO THE I.A.R FIELD.
8106 026752 010100 MOV R1,R0 ;USE OLD DESTINATION FOR SOURCE OF CKTRAP MOVE.
8107 026754 013701 002202 MOV CSRA,R1 ;SET UP CSR AS THE CKTRAP MOVE DESTINATION.
8108 026760 004737 015452 JSR PC,CKTRAP ;MOVE AND CHECK FOR TRAP.
8109 026764 103403 BCS 6$ ;IF NO TRAP, BYPASS ERROR.
8110 026766 052705 100002 BIS #100002,R5 ;SET FATAL WRITE ERROR FLAGS.
8111 026772 000440 BR 40$ ;EXIT AND REPORT FATAL ERROR.
8112
8113 :+
8114 : NOW, WE TEST EACH REGISTER FOR THIS LINE.
8115 :
8116 026774 012702 000010 6$: MOV #10,R2 ;INIT REGISTER COUNTER TO 8.
8117 027000 013737 002202 027134 MOV CSRA,50$ ;INITIALIZE THE REGISTER POINTER.
8118 027006 012700 027134 8$: MOV #50$,R0 ;SET UP REGISTER AS THE SOURCE FOR CKTRAP MOVE.
8119 027012 012701 027136 MOV #52$,R1 ;SET UP LOCAL STORAGE AS THE DES FOR CKTRAP.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 198
CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - ADRA -

```

8118 027016 004737 015452      JSR    PC,CKTRAP      ;PERFORM THE MOVE, CHECK FOR TRAP.
8119 027022 103402              BCS    10$            ;IF NO TRAP, BYPASS THE SETTING OF ERROR FLAGS.
8120 027024 052705 100001              BIS    #100001,R5    ;SET FATAL READ ERROR FLAGS.
8121 027030 010100      10$:   MOV    R1,R0        ;USE OLD DEST AS SRC FOR CKTRAP MOVE.
8122 027032 012701 027134              MOV    #50$,R1      ;SET UP REGISTER AS THE DEST FOR CKTRAP MOVE.
8123 027036 004737 015452      JSR    PC,CKTRAP      ;PERFORM THE MOVE, CHECK FOR TRAP.
8124 027042 103402              BCS    12$            ;IF NO TRAP, BYPASS THE SETTING OF ERROR FLAGS.
8125 027044 052705 100002              BIS    #100002,R5    ;SET FATAL WRITE ERROR FLAGS.
8126 027050 005237 027134      12$:   INC    50$        ;INCREMENT THE REGISTER
8127 027054 005237 027134              INC    50$        ; POINTER BY 2.
8128 027060 005302              DEC    R2           ;COUNT THE REGISTER.
8129 027062 001351              BNE    8$           ;LOOP TO TEST THE NEXT REGISTER ADDRESS.

```

:+
: NOW WE SET UP TO TEST THE NEXT LINE, OR TO EXIT IF WE ARE DONE.
:-

```

8134 027064 005204              INC    R4           ;INCREMENT THE LINE COUNTER.
8135 027066 020427 000010              CMP    R4,#NUMLNS   ;COMPARE LINE COUNTER .INST NUMBER OF LINES.
8136 027072 002707              BLT    2$           ;LOOP TO TEST THE NEX: LINE IF WE'RE NOT DONE.

```

:+
: DONE CHECKING DEVICE REGISTER ADDRESSES.
: REPORT ANY ERRORS AND EXIT.
:-

```

8142 027074 013737 002254 000004 40$:   MOV    TP4VEC,4      ;RESTORE THE NORMAL 004 TRAP VECTOR.
8143 027102 005705              TST    R5           ;CHECK THE ERROR FLAGS.
8144 027104 100015              BPL    60$          ;EXIT ROUTINE IF NO ERRORS.
8145              : REPORT 'DEVICE REGISTER ACCESS ERRORS'
8146 027106              ERRDF 101,EM0103,ER0101: >>>> ERRGR #101 <<<<.

```

```

TRAP  C$ERDF
.WORD 101
.WORD EM0103
.WORD ER0101

```

```

8152 027116              DODU   UNITN        ;DROP THIS UNIT FROM FUTHER TESTING.
8153 027116 013700 002200              MOV    UNITN,R0     ;
8154 027122 104451              TRAP  C$DCLN        ;
8155 027124 005037 002222              CLR    CTRLCF       ;INDICATE NO CTRL-C ABORT FROM TEST.
8156 027130              DOCLN              ;ABORT THIS SUB PASS.
8157 027130 104444              TRAP  C$DCLN        ;
8158 027132 000402              BR     60$           ;

```

:+
: LOCAL STORAGE.
:-

```

8162 027134 000000      50$:   .WORD 0          ;STORAGE FOR THE SOURCE OR DEST OF THE CKTRAP MOVE.
8163 027136 000000      52$:   .WORD U          ;STORAGE FOR THE SOURCE OR DEST OF THE CKTRAP MOVE.
8164 027140 005037 002222      60$:   CLR    CTRLCF     ;INDICATE THAT we ARE NOT WITHIN A TEST.
8165 027144              ENDTST

```

```

L10026: TRAP  C$SETST

```

8166 027144 104401

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 199
HARDWARE TEST - FRMERR -

8168
8169
8170
8171
8172
8173
8174
8175
8176
8177
8178
8179
8180
8181
8182
8183 027146
8184 027146
8185
8186
8187
8188 027146 123727 002176 000002
8189 027154 001162
8190 027156
8191 027156 012700 000240
8192 027162 104441
8193 027164 012737 177777 002222
8194 000002
8195 027172 012737 000002 002224
8196 027200 012737 000001 005222
8197 027200 012737 014071 005224
8198 027214 012737 007533 005226
8199 027222 005037 002406
8200
8201
8202
8203
8204
8205 027226 004737 015502
8206 027232 105133
8207
8208
8209
8210
8211 027234
8212 027234 012700 000340
8213 027240 104441
8214 027242
8215 027242 012746 000240
8216 027246 012746 025332
8217 027252 013746 002172
8218 027256 012746 000003
8219 027262 104437
8220 027264 062706 000010
8221 027270
8222 027270 012700 000140
8223 027274 104441

.SBTTL HARDWARE TEST - FRMERR -

- FRAMING ERROR GENERATION TEST -
*
* THIS TEST IS USED TO VERIFY THE FRAMING ERROR DETECTION CAPABILITIES
* OF THE DHV11.
* WHEN IN STAGGARED LOOPBACK MODE, CHARACTERS ARE TRANSMITTED FROM
* ONE GROUP OF LINES AT 8 BITS/CHAR, AND RECEIVED BY THE OTHER GROUP
* AT 5 BITS/CHAR. THIS WILL GENERATE A FRAMING ERROR FOR EACH CHARACTER.
* THIS TEST WILL ONLY EXECUTE IF THE STAGGARED LOOPBACK MODE IS SELECTED.
* THE SPECIAL STAGGARED LOOPBACK BERG CONNECTOR MUST BE FITTED.
* THE ACTIVE LINES BIT MASK IS USED TO INDICATE WHICH LINES HAVE BEEN
* REMOVED FROM FURTHER TESTING.
*
*-----*****

BGNTST
T2::

EXECUTE THIS TEST IN STAGGARED LOOPBACK MODE ONLY.

```

CMPB LOPBCK,#2 ;CHECK MODE SELECTED.
BNE 60$ ;EXIT IF STAGGERD LOOPBACK MODE NOT SELECTED.
SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
MOV #PRI05,R0
TRAP C$SPRI

MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (62)
MOV #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
MOV #6201,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
MOV #EM6201,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
R ERSMRF ;INITIALIZE THE 'REPORT ERROR SUMMARY' FLAGS.

```

RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
CLEAR TX AND RX INTERRUPT ENABLE BITS.
THIS SUBROUTINE REPORTS ERROR >>>> 6201 <<<<.

```

JSR PC,CLNRST ;RESET THE DUT.
BCC 60$ ;ABORT THE TEST IF FATAL ERROR FCJND IN RESET.

```

DISABLE ALL INTERRUPTS.
SET UP DMA TX AND RX INTERRUPT SERVICE ROUTINES.

```

SETPRI #PRI07 ;DISABLE ALL INTERRUPTS.
MOV #PRI07,R0
TRAP C$SPRI

SETVEC TXVECA,#TXDMA,#PRI05 ;SELECT DMA TX INT SERVICE RTN.
MOV #PRI05,-(SP)
MOV #TXDMA,-(SP)
MOV TXVECA,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP

SETPRI #PRI03 ;ALLOW INTERRUPTS.
MOV #PRI03,R0
TRAP C$SPRI

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 200
HARDWARE TEST - FRMERR -

```

8224
8225      :+ CLEAR TX, RX, AND DMA_START ERROR FLAGS.
8226      :-
8227 027276 005037 002410      CLR TXDNF      ;CLEAR TX DONE FLAGS FOR ALL LINES.
8228 027302 005037 002412      CLR RXDNF      ;CLEAR RX DONE FLAGS FOR ALL LINES.
8229 027306 005037 002252      CLR TXINTF     ;CLEAR TX ERROR FLAGS FOR ALL LINES.
8230
8231      :+ SET 'P ERROR TABLE AND DATA PATTERN TABLE.
8232      : THE NUMERICAL VALUE OF THE CHARACTER INDICATES THE NUMBER OF THE LINE
8233      : THAT TRANSMITTED IT.
8234      :-
8235 027312 012700 003210      MOV #ERCNTB,R0 ;PASS THE ADDRESS OF THE TABLE TO BE CLEARED.
8236 027316 004737 015524      JSR PC,CLR16W ;CLEAR THE RX ERROR COUNTERS TABLE.
8237 027322 005037 003510      CLR BUFBAS    ;SET SINGLE CHAR DATA TO BE A NULL.
8238
8239      :+ INITIALISE DMA PARAMETERS IN THE CONTROL BLOCK.
8240      : TRANSMISSION ON LINE GROUP 1 AT 8 BITS/CHAR,1 STOP BITS,ODD PARITY.
8241      : RECEPTION ON LINE GROUP 2 AT 5 BITS/CHAR,1 STOP,ODD PARITY.
8242      :-
8243 027326 012700 156470      MOV #156470,R0 ;PASS LPR PARAMETER FOR 8 BITS/CHAR.
8244 027332 012701 156440      MOV #156440,R1 ;PASS LPR PARAMETER FOR 5 BITS/CHAR.
8245 027336 004737 016746      JSR PC,GETTIM ;GET TIME-OUT BASED ON MINIMUM BAUDRATE IN USE.
8246 027342 012702 003510      MOV #BUFBAS,R2 ;PASS START ADDRESS OF DATA PATTERN.
8247 027346 012703 000001      MOV #1,R3      ;PASS LENGTH OF DATA PATTERN.
8248 027352 013704 002230      MOV LGRP1M,R4 ;PASS LINE GROUP OF LINES THAT ARE TO TX.
8249 027356 004737 016116      JSR PC,FRPSUP ;SET UP DUT FOR TRANSMISSION AND RECEPTION.
8250
8251      :+
8252      : PURGE THE FIFO OF ANY UN-WANTED CHARACTERS. THIS ROUTINE REPORTS ERRORS
8253      : WITH WITH ERROR NUMBERS FROM >>>> 6202 THRU 6204 <<<<.
8254      : PERFORM TRANSMISSION AND RECEPTION AT 9600 BAUD.
8255      : REPORT ANY ERRORS FOUND, IE. FRAMING ERROR BIT CLEAR OR PARITY ERROR SET.
8256      :-
8257 027362 005237 005224      INC ERRNBR    ;SET THE ERROR REPORT NUMBER TO 6202.
8258 027366 004737 020674      JSR PC,PUFIFR ;CLEAN OUT THE FIFO.
8259 027372 103053              BCC 60$      ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
8260 027374 012737 014075 005224      MOV #6205,ERRNBR ;SET THE ERROR NUMBER TO 6205.
8261 027402 004737 023506      JSR PC,TXFRPR ;TX DATA PATTERN ON SELECTED ACTIVE LINES.
8262 027406 012705 100000      MOV #100000,R5 ;PASS FRAMING ERROR TEST FLAG.
8263
8264      :+ THIS SUBROUTINE REPORTS ERROR NUMBER >>>> 6205 <<<<.
8265      :-
8266 027412 004737 015174      JSR PC,CKFRPR ;READ CHARACTERS, REPORT ANY ERRORS FOUND.
8267
8268      :+ REVERSE TRANSMISSION/RECEPTION ROLES ON ALL ACTIVE LINES, AND REPEAT TEST.
8269      :-
8270 027416 005104              COM R4        ;REVERSE ROLES FOR TRANSMISSION AND RECEPTION.
8271 027420 004737 016116      JSR PC,FRPSUP ;SET UP DUT FOR TRANSMISSION AND RECEPTION.
8272 027424 005237 005224      INC ERRNBR    ;SET ERROR NUMBER TO 6206.
8273
8274      :+ THIS ROUTNE REPORTS ERRORS WITH NUMBERS >>>> 6206 THRU 6208 <<<<.
8275      :-
8276 027430 004737 020674      JSR PC,PUFIFR ;CLEAN OUT THE FIFO.
8277 027434 103032              BCC 60$      ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
8278 027436 012737 014101 005224      MOV #6209,ERRNBR ;SET ERROR NUMBER TO 6209.
8279 027444 004737 023506      JSR PC,TXFRPR ;TX DATA PATTERN ON SELECTED ACTIVE LINES.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 201
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - FRMERR -

```

8280 027450 012705 100000          MOV    #100000,R5          ;PASS FRAMING ERROR TEST FLAG.
8281
8282          :+ THIS SUBROUTINE REPORTS ERRORS >>>> 6209 <<<<.
8283          :-
8284 027454 004737 015174          JSR    PC,CKFRPR          ;READ CHARACTERS, REPORT ANY ERRORS FOUND.
8285 027460 005237 005224          INC    ERRNBR             ;SET ERROR NUMBER TO 6210.
8286
8287          :+
8288          :. DISABLE INTERRUPTS.
8289          :. CLEAR THE INTERRUPT VECTORS.
8290          :. UPDATE THE ACTIVE LINES BIT MAP TO REFLECT LINES REMOVED FROM TESTING.
8291          :. THIS SUBROUTINE REPORTS ERRORS >>>> 6210 THRU 6212 <<<<.
8292          :-
8293 027464 004737 023602          JSR    PC, TXIEO          ;DISABLE ALL TX INTERRUPTS.
8294 027470 004737 024230          JSR    PC, TXRREP         ;REPORT FINAL ERRORS FROM TX/RX.
8295
8296 027474 012700 000340          SETPRI #PRI07            ;DISABLE ALL INTERRUPTS.
8297 027500 104441
8298 027502
8299 027502 013700 002172          CLRVEC TXVECA           ;RETURN TX INT VECTOR TO UNUSED POOL.
8300 027506 104436
8301 027510 012737 014105 005224          MOV    #6213.,ERRNBR     ;SET ERROR NUMBER TO 6213.
8302          :+
8303          :. THIS SUBROUTINE REPORTS ERRORS >>>> 6213 <<<<.
8304          :-
8305 027516 004737 021664          JSR    PC,REPSMR         ;REPORT ERROR SUMMARIES IF CALLED FOR.
8306 027522 005037 002222          CLR    CTRLCF           ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8307
8308          ENDTST
8309 027526
8310 027526 104401
    
```

L10027: TRAP C\$ETST

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 202
HARDWARE TEST - PARERR -

```

8311 .SBTTL HARDWARE TEST - PARERR -
8312 +*****
8313 - PARITY ERROR GENERATION TEST -
8314 *
8315 * THIS TEST IS USED TO VERIFY THE PARITY ERROR DETECTION AND REPORT
8316 * CAPABILITIES OF THE DUT.
8317 * WHEN STAGGARED LOOPBACK MODE IS SELECTED, DATA IS TRANSMITTED
8318 * ON ALL ACTIVE LINES IN LINE GROUP 1 WITH ODD PARITY SELECTED,
8319 * AND RECEIVED ON LINES IN GROUP 2 WITH EVEN PARITY SELECTED.
8320 * THIS WILL GENERATE A PARITY ERROR FOR EACH CHARACTER RECEIVED.
8321 * THE PARITY SELECTION IS THEN REVERSED ON THE LINES IN EACH GROUP
8322 * AND THE TEST IS REPEATED.
8323 * THIS TEST WILL ONLY EXECUTE IF THE STAGGARED LOOPBACK MODE IS SELECTED.
8324 * THE SPECIAL STAGGARED LOOPBACK BERG CONNECTOR MUST BE FITTED.
8325 *
8326 +-----+
8327 027530 BGNSTST
8328 027530 T3::
8329
8330 + EXECUTE THIS TEST IN STAGGARED LOOPBACK MODE ONLY.
8331 -
8332 027530 123727 002176 000002 CMPB LOPBCK,#2 ;CHECK MODE SELECTED.
8333 027536 001167 BNE 60$ ;EXIT IF STAGGERD LOOPBACK MODE NOT SELECTED.
8334 027540 SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
8335 027540 012700 000240 MOV #PRI05,R0
8336 027544 104441 TRAP C$SPRI
8337 027546 012737 177777 002222 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
8338 000003 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
8339 027554 012737 000003 002224 MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (63)
8340 027562 012737 000001 005222 MOV #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
8341 027570 012737 014235 005224 MOV #6301,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
8342 027576 012737 007566 005226 MOV #EM6301,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
8343 027604 005037 002406 CLR ERSMRF ;INITIALIZE THE 'REPORT ERROR SUMMARY' FLAGS.
8344
8345 + RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
8346 + CLEAR TX AND RX INTERRUPT ENABLE BITS.
8347 + THIS SUBROUTINE REPORTS ERROR >>>> 6301 <<<<<.
8348 -
8349 027610 004737 015502 JSR PC,CLNRST ;RESET THE DUT.
8350 027614 103140 BCC 60$ ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
8351
8352 + DISABLE ALL INTERRUPTS.
8353 + SET UP DMA TX AND RX INTERRUPT SERVICE ROUTINES.
8354 -
8355 027616 SETPRI #PRI07 ;DISABLE ALL INTERRUPTS.
8356 027616 012700 000340 MOV #PRI07,R0
8357 027622 104441 TRAP C$SPRI
8358 027624 SETVEC TXVECA,#TXDMA,#PRI05 ;SELECT DMA TX INT SERVICE RTN.
8359 027624 012746 000240 MOV #PRI05,-(SP)
8360 027630 012746 025332 MOV #TXDMA,-(SP)
8361 027634 013746 002172 MOV TXVECA,-(SP)
8362 027640 012746 000003 MOV #3,-(SP)
8363 027644 104437 TRAP C$SVEC
8364 027646 062706 000010 ADD #10,SP
8365 027652 SETPRI #PRI03 ;ALLOW INTERRUPTS.
8366 027652 012700 000140 MOV #PRI03,R0

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 203
HARDWARE TEST - PARERR -

```

8367 027656 104441 TRAP C$SPRI
8368
8369
8370
8371 027660 005037 002410
8372 027664 005037 002412
8373 027670 005037 002252
8374
8375
8376
8377 027674 012700 003210
8378 027700 004737 015524
8379
8380
8381
8382
8383 027704 012700 156470
8384 027710 012701 156570
8385 027714 004737 016746
8386 027720 012702 005062
8387 027724 012703 000020
8388 027730 013704 002230
8389 027734 004737 016116
8390
8391
8392
8393
8394
8395
8396
8397
8398
8399
8400
8401
8402 027740 004737 020674
8403 027744 103064
8404 027746 012737 014241 005224
8405 027754 004737 017166
8406 027760 005005
8407
8408
8409
8410 027762 004737 015174
8411 027766 005237 005224
8412
8413
8414
8415 027772 004737 024230
8416 027776 012737 014246 005224
8417 030004 005037 002410
8418 030010 005037 002412
8419 030014 005037 002252
8420
8421
8422

```

```

: +
: CLEAR TX/RX FLAGS.
: -
      CLR TXDNF ;CLEAR TX DONE FLAGS FOR ALL LINES.
      CLR RXDNF ;CLEAR RX DONE FLAGS FOR ALL LINES.
      CLR TXINTF ;CLEAR TX ERROR FLAGS FOR ALL LINES.
: +
: SET UP ERROR COUNTER TABLE.
: -
      MOV #ERCNTB,R0 ;PASS THE ADDRESS OF THE TABLE TO BE CLEARED.
      JSR PC,CLR16W ;CLEAR THE RX ERROR COUNTERS TABLE.
: +
: INITIALISE DMA PARAMETERS IN THE CONTROL BLOCK.
: -
      MOV #156470,R0 ;PASS LPR PARAMETER WITH ODD PARITY.
      MOV #156570,R1 ;PASS LPR PARAMETER WITH EVEN PARITY.
      JSR PC,GETTIM ;GET TIME-OUT BASED ON MINIMUM BAUDRATE IN USE.
      MOV #SDP2B,R2 ;PASS START ADDRESS OF DATA PATTERN.
      MOV #16,R3 ;PASS LENGTH OF DATA PATTERN.
      MOV LGRP1M,R4 ;PASS BIT MAP OF LINES TO BE SET WITH ODD PAR.
      JSR PC,FRPSUP ;SET UP DUT FOR TRANSMISSION AND RECEPTION.
: +
: PURGE THE FIFO OF ANY UN-WANTED CHARACTERS.
: PERFORM TRANSMISSION AND RECEPTION OF THE 16 BYTE DATA PATTERN AT 9600 BAUD.
: TRANSMISSION ON LINE IN GROUP 1, 8 BITS/CHAR, 1 STOP BITS, ODD PARITY.
: RECEPTION ON LINES IN GROUP 2 AT 8 BITS/CHAR, 1 STOP, EVEN PARITY.
: REMOVE CHARACTERS FROM THE FIFO AND LOOK FOR THE PARITY ERROR BIT BEING SET.
: REPORT ANY ERRORS FOUND, IE. FRAMMING ERROR BIT SET OR PARITY ERROR CLEAR.
: -
: +
: THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6302 THRU 6304 <<<<.
: -
      JSR PC,PUFIFR ;CLEAN OUT THE FIFO.
      BCC 60$ ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
      MOV #6305,ERRNBR ;SET ERROR NUMBER TO 6305
      JSR PC,INIDMA ;TX DATA PATTERN ON ALL ACTIVE LINES.
      CLR R5 ;PASS PARITY ERROR TEST FLAG.
: +
: THIS SUBROUTINE REPORTS ERROR NUMBER >>>> 6305 <<<<.
: -
      JSR PC,CKFRPR ;READ CHARACTERS, REPORT ANY ERRORS FOUND.
      INC ERRNBR ;SET ERROR NUMBER TO 6306.
: +
: THIS SUBROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6306 THRU 6309 <<<<
: -
      JSR PC,TXRREP ;REPORT FINAL ERRORS FROM TX/RX.
      MOV #6310,ERRNBR ;SET ERROR NUMBER TO 6310.
      CLR TXDNF ;CLEAR TX DONE FLAGS FOR ALL LINES.
      CLR RXDNF ;CLEAR RX DONE FLAGS FOR ALL LINES.
      CLR TXINTF ;CLEAR TX DMA HANDOVER ERROR FLAGS.
: +
: REVERSE TRANSMISSION/RECEPTION ROLES ON ALL ACTIVE LINES, AND REPEAT TEST.
: -

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 204
CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - PARERR -

```

8423 030020 005104          COM      R4          ;REVERSE ROLES FOR TRANSMISSION AND RECEPTION.
8424 030022 004737 016116  JSR      PC,FRPSUP  ;SET UP DUT FOR TRANSMISSION AND RECEPTION.
8425
8426          ;+ THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6310 THRU 6311 <<<<.
8427          ;-
8428 030026 004737 020674  JSR      PC,PUFIFR  ;CLEAN OUT THE FIFO.
8429 030032 103031          BCC     60$        ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
8430 030034 012737 014250 005224  MOV     #6312.,ERRNBR ;SET ERROR NUMBER TO 6312.
8431 030042 004737 017166  JSR      PC,INIDMA  ;TX DATA PATTERN ON SELECTED ACTIVE LINES.
8432
8433          ;+ THIS SUBROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6312 THRU 6316 <<<<.
8434          ;-
8435 030046 004737 015174          JSR     PC,CKFRPR  ;READ CHARACTERS, REPORT ANY ERRORS FOUND.
8436 030052 012737 014255 005224  MOV     #6317.,ERRNBR ;SET ERROR NUMBER TO 6317.
8437
8438          ;+
8439          ;-
8440          ;-
8441          ;-
8442 030060 004737 023602          JSR     PC,TXIE0   ;DISABLE ALL TX INTERRUPTS.
8443          ;+
8444          ;-
8445          ;-
8446 030064 004737 024230          JSR     PC,TXRREP  ;REPORT FINAL ERRORS FROM TX/RX.
8447
8448 030070          SETPRI  #PRI07     ;DISABLE ALL INTERRUPTS.
8449 030070 012700 000340          MOV     #PRI07,RO  ;
8450 030074 104441          TRAP   C$SPRI     ;
8451 030076          CLRVEC  TXVECA    ;RETURN TX INT VECTOR TO UNUSED POOL.
8452 030076 013700 002172          MOV     TXVECA,RO ;
8453 030102 104436          TRAP   C$CVEC    ;
8454
8455          ;+
8456          ;-
8457          ;-
8458 030104 012737 014261 005224  MOV     #6321.,ERRNBR ;SET ERROR NUMBER TO 6321.
8459 030112 004737 021664          JSR     PC,REPSMR  ;REPORT ERROR SUMMARIES IF CALLED FOR.
8460
8461 030116 005037 002222          60$:   CLR     CTRLCF  ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8462 030122          ENDTST
8463 030122
8464 030122 104401          L10030: TRAP   C$ETST

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 205
HARDWARE TEST - BREAKB -

8465
8466
8467
8468
8469
8470
8471
8472
8473
8474
8475
8476
8477
8478
8479
8480
8481
8482
8483
8484
8485
8486
8487
8488
8489
8490
8491
8492
8493
8494
8495
8496
8497
8498
8499
8500
8501
8502
8503
8504
8505
8506
8507
8508
8509
8510
8511
8512
8513
8514
8515
8516
8517
8518
8519
8520

030124
030124
030124 012737 177777 002222
000004
030132 012737 000004 002224
030140 012737 000001 005222
030146 012737 014401 005224
030154 012737 007611 005226

030162 004737 015502
030166 103165

030170 004737 023602
030174 004737 022314
030200 012705 000377
030204 012700 000200
030210 004737 025014
030214 012704 000012
030220 004737 015622

030224 012700 156430
030230 004737 025044

030234 013705 002174
030240 004737 023412

030244 005237 005224
030250 004737 020674

```
.SBTTL HARDWARE TEST - BREAKB -
:*****:*****
: - BREAK GENERATION TEST -
: THIS TEST VERIFIES THAT ALL SERIAL TRANSMIT LINES CAN GENERATE A BREAK
: BY SETTING THE BRK BIT IN THE ASSOCIATED LNCTRL REGISTER.
: USE OF THE INTERNAL LOOPBACK FEATURE OF THE DUARTS IS MADE TO MINIMISE
: ANY EXTERNAL EFFECTS CAUSED ON THE SERIAL LINES BY THIS TEST.
: FRAMING ERROR DETECTION IS USED TO INDICATE THE PRESENCE OF A BREAK,
: BY SETTING THE APPROPRIATE BIT IN THE RBUF REGISTER.
:-----:*****

BGNTST
T4::

MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (64)
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #6401,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
MOV #EM6401,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERRRTL.

: +
: 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 >>>> 6401 <<<<.
: -
JSR PC,CLNRST ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
BCC 60$ ;EXIT TEST IF FATAL ERROR FOUND.

: +
: SET UP DEVICE UNDER TEST (DUT) TO:
: DISABLE TRANSMISSION AND RECEPTION INTERRUPTS.
: DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME TO CLEAR ANY BREAKS.
: -
JSR PC,TXIEO ;DISABLE TRANSMISSION INTERRUPTS.
JSR PC,RXIEO ;DISABLE RECEPTION INTERRUPTS.
MOV #MAPLNS,R5 ;PASS ACTIVE LINE BIT MAP.
MOV #200,R0 ;PASS INTERNAL LOOPBACK MODE.
JSR PC,WTWLNC ;SELECT INTERNAL LOOPBACK,DISABLE DMA.
MOV #10,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
JSR PC,DELAY ;DELAY TO ALLOW ANY BREAKS TO BE CLEARED.

: +
: SET UP TRANSMISSION AN RECEPTION PARAMETERS FOR ALL LINES.
: 9600 BAUD,8 CHAR,1 STOPBIT,NO PARITY.
: -
MOV #156430,R0 ;SET UP BAUD RATE,ETC.
JSR PC,WTWLPR ;SET COMMUNICATION PARAMETERS ON ALL LINES.

: +
: ENABLE TRANSMITTERS ON ALL ACTIVE LINES.
: -
MOV ACTLNS,R5 ;PASS ACTIVE LINE BIT MAP.
JSR PC,TXENBL ;ENABLE TRANSMISSIONS ON ALL LINES.

: +
: PURGE THE FIFO OF ANY UNWANTED CHARACTERS.
: THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 6402 THRU 6404 <<<<.
: -
INC ERRNBR ;SET ERROR NUMBER TO 6402.
JSR PC,PUFIFR ;PURGE FIFO.
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 206
HARDWARE TEST - BREAKB -

```

8521 030254 103132          BCC      60$          ;ABORT TEST IF FIFO WILL NOT CLEAR.
8522
8523          :+
8524          :   VERIFY BREAK GENERATION ON INDIVIDUAL LINES.
8525          :   CLEAR BREAKS ON ALL LINES.
8526          :   DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME FOR ANY BREAKS TO BE CLEARED.
8527          :   SELECT LINE,SET BREAK BIT IN LNCTRL REGISTER.
8528          :   TEST FOR A CHARACTER IN THE FIFO WITH FRAME ERROR.
8529 030256 005002          2$:      CLR      R2          ;CLEAR LINE COUNTER.
8530 030260 012703 000001          :      MOV      #1,R3        ;SET UP ACTIVE LINE BIT MASK.
8531 030264 030337 002174          4$:      BIT      R3,ACTLNS    ;CHECK IF THIS LINE IS ACTIVE.
8532 030270 001434          :      BEQ      8$          ;GO SELECT NEXT LINE IF THIS ONE IS INACTIVE.
8533 030272 012700 000200          :      MOV      #200,R0       ;SET UP PARAMETER TO CLEAR BREAK BITS.
8534 030276 004737 025014          :      JSR      PC,WTWLNLC    ;CLEAR BREAK BIT,RESELECT INTERNAL LOOPBACK.
8535 030302 012704 000012          :      MOV      #10.,R4      ;PASS DELAY TIME OF 10 MILLI SECONDS.
8536 030306 004737 015622          :      JSR      PC,DELAY      ;DELAY TO ALLOW BREAKS TO BE CLEARED.
8537
8538          :+
8539          :   SET BREAK BIT ON SELECTED LINE.
8540          :   SET UP PARAMETERS TO TEST FOR THE FRAME ERROR BIT SET IN RBUF.
8541          :   TIME-OUT = 5 MILLI SECONDS.
8542          :   CALL ROUTINE TO CHECK FOR CONDITION FOUND.
8543 030312 010305          6$:      MOV      R3,R5          ;COPY ACTIVE LINE BIT MASK.
8544 030314 012700 000214          :      MOV      #214,R0       ;SET BREAK,RESELECT LOOPBACK,ENABLE RECEPTION.
8545 030320 004737 025014          :      JSR      PC,WTWLNLC    ;SET BREAK ON SELECTED LINE.
8546
8547          :+
8548          :   DELAY FOR 5 MS TO ALLOW TIME FOR BREAK TO BE GENERATED AND RECEIVED.
8549          :   VERIFY RECEPTION OF A CHARACTER WITH FRAME ERROR BIT SET.
8550 030324 012704 000005          :      MOV      #5.,R4        ;SET DELAY VALUE TO 5 MILLI SECS.
8551 030330 004737 015622          :      JSR      PC,DELAY      ;ALLOW TIME FOR CHARACTER RECEPTION.
8552 030334 017700 151644          :      MOV      @RBUFA,R0     ;GET CHARACTER FROM RBUF REGISTER.
8553 030340 032700 020000          :      BIT      #BIT13,R0     ;CHECK FOR FRAME ERROR BIT.
8554 030344 001006          :      BNE      8$          ;SKIP ERROR REPORT IF SET.
8555 030346 012701 007640          :      MOV      #EM6402,R1    ;SELECT MESSAGE TO BE PRINTED.
8556          :REPORT ERROR'BREAK NOT RECEIVED ON LINE #NN'
8557 030352          :      ERRDF  6405,EM6401,ER6401 ; >>>> ERROR #6405 <<<<.
8558 030352 104455          :      TRAP   C$ERDF
8559 030354 014405          :      .WORD  6405
8560 030356 007611          :      .WORD  EM6401
8561 030360 012562          :      .WORD  ER6401
8562 030362 006303          8$:      ASL      R3          ;SHIFT BIT MASK FOR NEXT LINE.
8563 030364 005202          :      INC      R2          ;NEXT LINE
8564 030366 020227 000010          :      CMP      R2,#NUMLNS    ;CHECK FOR MAX LINE COUNT.
8565 030372 001334          :      BNE      4$          ;IF <>,LOOP TO CHECK NEXT LINE
8566
8567          :+
8568          :   VERIFY BREAK GENERATION ON ALL LINES SIMULTANEOUSLY.
8569          :   CLEAR BREAKS ON ALL LINES.
8570          :   DELAY FOR 10 MILLI-SECONDS TO ALLOW TIME FOR ANY BREAKS TO BE CLEARED.
8571          :   PURGE THE FIFO.
8572          :   SET BREAK BIT IN LNCTRL REGISTERS ON ALL ACTIVE LINES.
8573          :   TEST FOR CHARACTERS IN THE FIFO WITH FRAME ERROR.
8574 030374 012705 000377          :      MOV      #MAPLNS,R5    ;SET UP LINE TO CLEAR BREAKS ON.
8575 030400 012700 000200          :      MOV      #200,R0       ;SET UP PARAMETER TO CLEAR BREAK BITS.
8576 030404 004737 025014          :      JSR      PC,WTWLNLC    ;CLEAR BREAK BIT,RESELECT INTERNAL LOOPBACK.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 207
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - BREAKB -

```

8577 030410 012704 000012      MOV    #10.,R4      ;PASS DELAY TIME OF 10 MILLI SECONDS.
8578 030414 004737 015622      JSR    PC,DELAY    ;DELAY TO ALLOW BREAKS TO BE CLEARED.
8579
8580      ;+ PURGE THE FIFO OF UNWANTED CHARACTERS.
8581      ;-
8582 030420 004737 020612      JSR    PC,PUFIFO   ;PURGE FIFO.
8583 030424 103044                BCC    50$         ;GO REPORT ERROR IF FAILED TO CLEAN_OUT FIFO.
8584
8585      ;+ SET UP PARAMETERS FOR SETTING THE BREAK BIT ON ALL ACTIVE LINES.
8586      ;- THEN CALL ROUTINE TO DO IT.
8587
8588 030426 013705 002174 10$:  MOV    ACTLNS,R5   ;SET UP ACTIVE LINE BIT MASK.
8589 030432 012700 000214      MOV    #214,R0    ;SET BREAK,RESELECT LOOPBACK,ENABLE RECEPTION.
8590 030436 004737 025014      JSR    PC,WTWLC   ;SET BREAK ON SELECTED LINES.
8591
8592      ;+ DELAY FOR 10 MILLI SECONDS,TO ALLOW TIME FOR RECEPTION.
8593      ;- TEST FOR CHARACTERS IN FIFO WITH FRAME ERROR BIT SET.
8594
8595 030442 012704 000012      MOV    #10.,R4    ;SET DELAY VALUE TO 10 MILLI SECS.
8596 030446 004737 015622      JSR    PC,DELAY   ;ALLOW TIME FOR CHARACTER RECEPTION.
8597 030452 010502                MOV    R5,R2      ;COPY ACTIVE LINE BIT MAP.
8598 030454 004737 017306      JSR    PC,MAPCNT  ;COUNT THE NUMBER OF LINES AVAILABLE.
8599 030460 017701 151520 12$:  MOV    @RBUFA,R1   ;GET CHARACTER FROM RBUF REGISTER.
8600 030464 100011                BPL    14$        ;BRANCH IF DATA VALID NOT SET.
8601 030466 032701 020000      BIT    #BIT13,R1 ;CHECK FOR FRAME ERROR BIT.
8602 030472 001406                BEQ    14$        ;DO NOT CLR FLG FOR THIS LINE IF FRAME BIT CLR.
8603 030474 0C3301                SWAB   R1         ;GET LINE NUMBER IN LOW BYTE.
8604 030476 042701 177400      BIC    #177400,R1 ;CLEAR EVERYTHING BUT THE LINE NUMBER.
8605 030502 004737 017260      JSR    PC,LINBIT  ;CALC BIT MASK FROM LINE NUMBER.
8606 030506 040005                BIC    R0,R5      ;CLEAR LINE FLAG.
8607 030510 005302 14$:  DEC    R2         ;DECREMENT THE LINE NUMBER COUNTER.
8608 030512 001362                BNE    12$        ;LOOP TO GET THE NEXT CHARACTER.
8609 030514 005705                TST    R5         ;CHECK IF ANY BREAKS NOT RECEIVED.
8610 030516 001411                BEQ    60$        ;EXIT TEST IF ALL CLEAR.
8611 030520 012701 007640      MOV    #EM6402,R1 ;SELECT MESSAGE TO BE PRINTED.
8612      ;REPORT ERROR BREAK NOT RECEIVED ON LINE #NN".
8613 030524                ERRDF  6406,EM6401,ER6401; >>>> ERROR #6407 <<<<.
8614 030524 104455                TRAP   C$ERDF
8615 030526 014406                .WORD 6406
8616 030530 007611                .WORD EM6401
8617 030532 012562                .WORD ER6401
8618 030534 000402                BR     60$        ;EXIT THE TEST.
8619
8620 030536 004737 023074 50$:  JSR    PC,TSABRT  ;ABORT THE TEST.
8621 030542 005037 002222 60$:  CLR    CTRLCF    ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8622 030546
8623 030546
8624 030546 104401                L10031: TRAP   C$ETST

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 208
HARDWARE TEST - NORERR -

8625
8626
8627
8628
8629
8630
8631
8632
8633
8634
8635
8636
8637
8638
8639 030550
8640 030550
8641 030550
8642 030550 012700 J00240
8643 030554 104441
8644 000005
8645 030556 012737 000005 002224
8646 030564 012737 177777 002222
8647 030572 012737 000001 005222
8648 030600 012737 014711 005224
8649 030606 012737 007701 005226
8650
8651
8652
8653
8654
8655 030614 004737 015502
8656 030620 103402
8657 030622 000137 031246
8658
8659
8660
8661
8662 030626 004737 016036
8663 030632 103402
8664 030634 000137 031246
8665 030640 004737 017024
8666
8667
8668
8669
8670
8671
8672
8673
8674 030644 005237 005224
8675 030650 012700 000204
8676 030654 004737 025014
8677 030660 012700 177670
8678 030664 004737 025044
8679 030670 013704 000012
8680 030674 004737 015622

```
.SBTTL HARDWARE TEST - NORERR -
*****
- NO OVERRUN ERROR TEST -
*****
THIS TEST VERIFIES THAT THE DUT WILL NOT REPORT DATA OVERRUN
ERRORS WHEN THEY DO NOT OCCUR.
THIS TEST PUTS 256 CHARACTERS IN THE DUT FIFO PLUS 4 IN EACH ACTIVE
UART AND VERIFIES THAT NO OVERRUN ERRORS 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
SETPRI #PRIOS ;ALLOW LTC INTERRUPTS. T5::
MOV #PRIOS,R0
TRAP C$SPRI
TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (66)
MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #6601,ERRNBR ;SET ERROR NUMBER TO 6601.
MOV #EM6601,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
+
: RESET THE DUT TO A KNOWN STATE, REMOVE THE STATUS CODES FROM THE FIFO.
: CLEAR TX AND RX INTERRUPT ENABLE BITS IN THE CSR.
: THIS SUBROUTINE REPORTS ERROR >>>> 6601 <<<<.
-
JSR PC,CLNRST ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
BCS +6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
JMP 60$ ;EXIT THE TEST, FATAL ERROR WAS FOUND.
+
: FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
: INITIALIZE THE 256 BYTE DATA PATTERN.
-
JSR PC,FINACT ;FIND AN ACTIVE LINE.
BCS +6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
JMP 60$ ;EXIT THE TEST, FATAL ERROR WAS FOUND.
JSR PC,INDATP ;INITIALISE 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.
-
INC ERRNBR ;SET THE ERROR REPORT NUMBER TO 6602.
MOV #204,R0 ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
JSR PC,WTWLNCR ;INITILAISE THE LINE CONTROL REGISTER.
MOV #177670,R0 ;PASS THE LPR CONTENTS.
JSR PC,WTWLPR ;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.
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 209
HARDWARE TEST - NORERR -

```

8681 030700 012702 003510      MOV    #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
8682 030704 012703 000400      MOV    #BUFMID-BUFBAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN.
8683 030710 004737 015662      JSR    PC,DODMA        ;TRANSMIT THE DATA PATTERN.
8684 030714 103152              BCC    50$             ;EXIT IF ERROR FOUND DURING DMA TX.
8685
8686      ;+
8687      ;: WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
8688      ;:- THE FIFO.
8689 030716 005237 005224      INC    ERRNBR          ;SET ERROR NUMBER TO 6603.
8690 030722 012701 170454      MOV    #170454,R1     ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
8691 030726 013702 002202      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
8692 030732 004737 024740      JSR    PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX ACTION SET.
8693 030736 103141              BCC    50$             ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
8694 030740 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
8695 030744 004737 015622      JSR    PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
8696
8697      ;+
8698      ;: TRANSMIT 4 CHARACTERS ON EACH ACTIVE LINE.
8699      ;:-
8700 030750 013705 002174      MOV    ACTLNS,R5      ;ALTER PARAMETERS FOR ALL ACTIVE LINES.
8701 030754 012700 000204      MOV    #204,R0        ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
8702 030760 004737 025014      JSR    PC,WTWLNCR     ;INITIALISE THE LINE CONTROL REGISTER.
8703 030764 012700 177670      MOV    #177670,R0     ;PASS THE LPR CONTENTS.
8704 030770 004737 025044      JSR    PC,WTWLPRL     ;SET THE LPR CONTENTS TO 38.4K BAUD.
8705 030774 013704 000012      MOV    #10,R4         ;PASS DELAY TIME OF 10 MILLI SECONDS.
8706 031000 004737 015622      JSR    PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
8707
8708 031004 012702 003510      MOV    #BUFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
8709 031010 012703 000004      MOV    #4,R3          ;PASS THE LENGTH OF THE DATA PATTERN.
8710 031014 005001              CLR    R1              ;CLEAR THE LINE COUNTER.
8711 031016 005237 005224      INC    ERRNBR         ;SET ERROR NUMBER TO 6604.
8712 031022 010100      2$: MOV    R1,R0          ;CALCULATE THE LINE OFFSET FROM THE LINE #.
8713 031024 006300              ASL    R0              ;TEST FOR THIS LINE BEING ACTIVE.
8714 031026 036037 002332 002174 BIT    BITBL(R0),ACTLNS ;SKIP THE TX ON THIS LINE IF IT IS NOT ACTIVE.
8715 031034 001403              BEQ    4$              ;TRANSMIT THE 5 CHAR DATA PATTERN.
8716 031036 004737 015662      JSR    PC,DODMA        ;ABORT IF ERROR FOUND DURING DMA TX.
8717 031042 103077              BCC    50$             ;INCREMENT THE LINE COUNTER.
8718 031044 005201      4$: INC    R1              ;TEST FOR ALL POSSIBLE LINES HANDLED
8719 031046 020127 000010      CMP    R1,#NUMLNS     ;LOOP IF NOT ALL LINES HANDLED.
8720 031052 002763              BLT    2$
8721
8722 031054 005237 005224      INC    ERRNBR         ;SET ERROR NUMBER TO 6605.
8723 031060 012701 170040      MOV    #170040,R1     ;PASS TIME-OUT VALUE OF 32 MILLI SECS.
8724 031064 013702 002202      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
8725 031070 004737 024740      JSR    PC,WAIBIS      ;WAIT FOR A DMA TO COMPLETE, TX ACTION SET.
8726 031074 103062              BCC    50$             ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
8727 031076 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
8728 031102 004737 015622      JSR    PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
8729
8730      ;+
8731      ;: READ THE FIFO CHECKING FOR OVERRUN ERRORS. REPORT ERRORS IF FOUND.
8732      ;:- ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
8733 031106 013702 002174      MOV    ACTLNS,R2      ;GET THE NUMBER OF ACTIVE LINES.
8734 031112 004737 017306      JSR    PC,MAPCNT
8735 031116 006302              ASL    R2
8736 031120 006302              ASL    R2              ;MULTIPLY NUMBER OF ACTIVE LINES BY 4.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 210
CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - NORERR -

```

8737 031122 012705 000400      MOV    #256,R5
8738 031126 060205              ADD    R2,R5      ;CALCULATE NUMBER OF CHARACTERS TO RX.
8739 031130 005004              CLR    R4         ;CLEAR THE CHARACTER COUNTER.
8740 031132 012737 014716 005224 6$:  MOV    #6606,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
8741 031140 017702 151040      MOV    @RBUFA,R2  ;READ A CHARACTER FROM THE FIFO.
8742 031144 100032              BPL   10$        ;EXIT THE READ LOOP IF THE FIFO IS EMPTY.
8743
8744      ;+
8745      ;CHECK IF THE READ CHARACTER IS A BMP CODE.
8746      ;IF IT IS A BMP CODE SAVE IT ON THE QUEUE TO BE REPORTED LATER, AND
8747      ;ABORT THE TEST.
8748 031146 004737 014446      JSR    PC,CHKBMP  ;CHECK IF CHARACTER IS A BMP CODE.
8749 031152 103002              BCC   8$         ;BRANCH IF NOT A BMP CODE.
8750 031154              ERROR  >>>> ERROR #6606 <<<<<.
8751 031154 104460              TRAP  C$ERROR
8752 031156 000433              BR    60$       ;EXIT THIS TEST.
8753
8754 031160 005237 005224      8$:   INC    ERRNBR  ;SET ERROR NUMBER TO 6607.
8755 031164 005204              INC    R4        ;COUNT THIS CHARACTER.
8756 031166 020405              CMP    R4,R5    ;COMPARE # OF CHARS WITH MAX # OF CHARS.
8757 031170 003024              BGT   50$       ;ABORT TEST IF TOO MANY VALID CHARS READ.
8758 031172 032702 040000      BIT    #BIT14,R2 ;TEST THE OVERRUN BIT OF THE READ CHAR.
8759 031176 001755              BEQ   6$        ;LOOP TO READ THE NEXT CHAR IF NO ERROR.
8760 031200 005237 005224      INC    ERRNBR   ;SET ERROR NUMBER TO 6608.
8761 031204 012737 012660 005230      MOV    #ER7801,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
8762 031212 012701 007727      MOV    #EM6602,R1 ;PASS THE MESSAGE TO BE REPORTED.
8763 031216 010203              MOV    R2,R3
8764 031220 000303              SWAB  R3
8765 031222 042703 177760      BIC    #177760,R3 ;GET FAILING LINE NUMBER.
8766      ;REPORT 'OVERRUN ERROR REPORTED WHEN NONE FORCED, ON LINE NN...'
8767 031226              ERROR  >>>> ERROR #6608 <<<<<.
8768 031226 104460              TRAP  C$ERROR
8769 031230 000740              BR    6$        ;LOOP TO READ THE NEXT CHAR.
8770
8771 031232 005237 005224      10$:  INC    ERRNBR   ;SET ERROR NUMBER TO 6609.
8772 031236 020405              CMP    R4,R5    ;COMPARE NUMBER OF CHARS READ WITH EXPECTED.
8773 031240 001402              BEQ   60$       ;EXIT TEST WITHOUT ABORT IF CORRECT # OF CHARS.
8774
8775 031242 004737 023074      50$:  JSR    PC,TSABRT ;ABORT THE TEST, NON-RELATED TEST ERROR FOUND.
8776 031246 005037 002222      60$:  CLR    CTRLCF   ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8777 031252
8778 031252
8779 031252 104401              L10032: TRAP  C$SETST

```

CVDHCA0 DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 211
HARDWARE TEST - ORERR -

```

8780
8781
8782
8783
8784
8785
8786
8787
8788
8789
8790
8791
8792
8793
8794 031254
8795 031254
8796 031254
8797 031254 012700 000240
8798 031260 104441
8799 000006
8800 031262 012737 000006 002224
8801 031270 012737 177777 002222
8802 031276 012737 000001 005222
8803 031304 012737 015055 005224
8804 031312 012737 010001 005226
8805
8806
8807
8808
8809
8810 031320 004737 015502
8811 031324 103402
8812 031326 000137 032202
8813
8814
8815
8816
8817 031332 004737 016036
8818 031336 103402
8819 031340 000137 032202
8820 031344 004737 017024
8821
8822
8823
8824
8825
8826
8827
8828
8829 031350 005237 005224
8830 031354 012700 000204
8831 031360 004737 025014
8832 031364 012700 177670
8833 031370 004737 025044
8834 031374 013704 000012
8835 031400 004737 015622

```

```

.SBTTL HARDWARE TEST - ORERR -
*****
- OVERRUN ERROR TEST -

THIS TEST VERIFIES THAT THE DUT WILL REPORT DATA OVERRUN ERRORS WHEN
THEY OCCUR.
THIS TEST PUTS 256 CHARACTERS IN THE DUT FIFO PLUS 5 IN EACH ACTIVE
UART AND VERIFIES THAT OVERRUN ERRORS ARE REPORTED ON ALL ACTIVE LINES.
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.
*****

BGMTST
SETPRI #PRI05 ;ALLOW LTC INTERRUPTS. T6::
MOV #PRI05,R0
TRAP C$SPRI

TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (67)
MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #6701,ERRNBR ;SET ERROR NUMBER TO 6701.
MOV #EM6701,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 >>>> 6701 <<<<.
-
JSR PC,CLNRST ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
BCS .+6 ;SKIP EXIT OF TEST IF NO FATAL ERROR FOUND.
JMP 60$ ;EXIT THE TEST, FATAL ERROR WAS FOUND.

+
: FIND AN ACTIVE LINE ON WHICH TO PERFORM THE TEST.
: INITIALIZE THE 256 BYTE DATA PATTERN.
-
JSR PC,FINACT ;FIND AN ACTIVE LINE.
BCS .+6 ;IF ACTIVE LINE IS FOUND, DON'T ABORT TEST.
JMP 60$ ;ABORT THE TEST, NO ACTIVE LINES WERE FOUND.
JSR PC,INDATP ;INITIALISE 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.
-
INC ERRNBR ;SET ERROR NUMBER TO 6702.
MOV #204,R0 ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
JSR PC,WTWLNCR ;INITIALISE THE LINE CONTROL REGISTER.
MOV #177670,R0 ;PASS THE LPR CONTENTS.
JSR PC,WTWLPR ;SET THE LPR CONTENTS TO 38.4K BAUD.
MOV 10.,R4 ;PASS DELAY TIME OF 10 MILLI SECONDS.
JSR PC,DELAY ;WAIT FOR LNCRTL AND LPR REGS TO BE UPDATED.

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 212
HARDWARE TEST - ORERR -

```

8836 031404 012702 003510      MOV    #BUFBAS,R2      ;PASS THE START OF THE DATA PATTERN TO TX.
8837 031410 012703 000400      MOV    #BUFMID-BUFBAS,R3 ;PASS THE LENGTH OF THE DATA PATTERN.
8838 031414 004737 015662      JSR    PC,DODMA        ;TRANSMIT THE DATA PATTERN.
8839 031420 103402              BCS    +6              ;IF NO ERROR FOUND DURING DMA TX, DON'T ABORT.
8840 031422 000137 032176      JMP    50$            ;ABORT TEST, ERROR FOUND DURING DMA TX.
8841
8842      ;+
8843      ;: WAIT FOR DMA TO COMPLETE, THEN WAIT FOR THE LAST CHARACTER TO ARRIVE IN
8844      ;: THE FIFO.
8845      ;-
8845 031426 005237 005224      INC    ERRNBR          ;SET ERROR NUMBER TO 6703.
8846 031432 012701 170454      MOV    #170454,R1     ;PASS TIME-OUT VALUE OF 300 MILLI SECS.
8847 031436 013702 002202      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
8848 031442 004737 024740      JSR    PC,WAIBIS      ;WAIT FOR DMA TO COMPLETE, TX ACTION SET.
8849 031446 103402              BCS    +6              ;IF NO TIME-OUT ON DMA COMPLETION, DON'T ABORT.
8850 031450 000137 032176      JMP    50$            ;ABORT TEST, TIME-OUT ON DMA COMPLETION.
8851 031454 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
8852 031460 004737 015622      JSR    PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
8853
8854      ;+
8855      ;: TRANSMIT 5 CHARACTERS ON EACH ACTIVE LINE.
8856      ;-
8856 031464 013705 002174      MOV    ACTLNS,R5      ;ALTER PARAMETERS FOR ALL ACTIVE LINES.
8857 031470 012700 000204      MOV    #204,R0        ;PASS PARAMETER FOR INTERNAL LOPBCK,ENABLE RX.
8858 031474 004737 025014      JSR    PC,WTWLNCR     ;INITIALISE THE LINE CONTROL REGISTER.
8859 031500 012700 177670      MOV    #177670,R0     ;PASS THE LPR CONTENTS.
8860 031504 004737 025044      JSR    PC,WTWLPRL     ;SET THE LPR CONTENTS TO 38.4K BAUD.
8861 031510 013704 000012      MOV    10,R4          ;PASS DELAY TIME OF 10 MILLI SECONDS.
8862 031514 004737 015622      JSR    PC,DELAY       ;WAIT FOR LNCTRL AND LPR REGS TO BE UPDATED.
8863
8864 031520 012702 003510      MOV    #BUFBAS,R2     ;PASS THE START OF THE DATA PATTERN TO TX.
8865 031524 012703 000005      MOV    #5,R3          ;PASS THE LENGTH OF THE DATA PATTERN.
8866 031530 005001              CLR    R1              ;CLEAR THE LINE COUNTER.
8867 031532 005237 005224      INC    ERRNBR         ;SET ERROR NUMBER TO 6704.
8868 031536 010100      2$: MOV    R1,R0
8869 031540 006300      ASL    R0              ;CALCULATE LINE OFFSET FROM THE LINE #.
8870 031542 036037 002332 002174 BIT    BITTBL(R0),ACTLNS ;TEST FOR THIS LINE BEING ACTIVE.
8871 031550 001405      BEQ    4$              ;SKIP THE TX ON THIS LINE IF IT IS NOT ACTIVE.
8872 031552 004737 015662      JSR    PC,DODMA        ;TRANSMIT THE 5 CHAR DATA PATTERN.
8873 031556 103402              BCS    +6              ;IF NO TIME-OUT ON DMA COMPLETION, DON'T ABORT.
8874 031560 000137 032176      JMP    50$            ;ABORT TEST, TIME-OUT ON DMA COMPLETION.
8875 031564 005201      4$: INC    R1              ;INCREMENT THE CHARACTER COUNTER.
8876 031566 020127 000010      CMP    R1,#NUMLNS     ;TEST FOR ALL POSSIBLE LINES HANDLED
8877 031572 002761      BLT    2$              ;LOOP IF NOT ALL LINES HANDLED.
8878
8879 031574 005237 005224      INC    ERRNBR         ;SET ERROR NUMBER TO 6705.
8880 031600 012701 170040      MOV    #170040,R1     ;PASS TIME-OUT VALUE OF 32 MILLI SECS.
8881 031604 013702 002202      MOV    CSRA,R2        ;PASS THE ADDRESS OF THE CSR.
8882 031610 004737 024740      JSR    PC,WAIBIS      ;WAIT FOR A DMA TO COMPLETE, TX ACTION SET.
8883 031614 103170              BCC    50$            ;ABORT THE TEST IF TIME-OUT ON DMA COMPLETION.
8884 031616 012704 000005      MOV    #5,R4          ;PASS DELAY OF 5 MILLI SECS.
8885 031622 004737 015622      JSR    PC,DELAY       ;WAIT FOR LAST CHAR TO ARRIVE IN THE FIFO.
8886
8887      ;+
8888      ;: READ 256 CHARS FROM THE FIFO CHECKING FOR BMP CODES.
8889      ;: ABORT THE TEST IF A BMP CODE WAS FOUND IN THE FIFO.
8890      ;-
8890 031626 012704 000400      MOV    #256,R4        ;SET UP THE CHARACTER COUNTER.
8891 031632 012737 015062 005224 6$: MOV    #6706,ERRNBR    ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 213
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - ORERR -

```

8892 031640 017702 150340      MOV    @RBUFA,R2      ;READ A CHARACTER FROM THE FIFO.
8893 031644 100154      BPL    50$           ;ABORT THE TEST IF DATA.VALID IS CLEAR.
8894 031646 005237 005224      INC    ERRNBR        ;SET ERROR NUMBER TO 6707.
8895 031652 004737 014446      JSR    PC,CHKBMP     ;CHECK IF CHARACTER IS A BMP CODE.
8896 031656 103545      BCS    24$           ;REPORT ERROR AND ABORT TEST IF A BMP CODE.
8897 031660 005304      DEC    R4            ;COUNT THIS CHARACTER.
8898 031662 001363      BNE    6$           ;LOOP IF NOT 256 CHARS READ FROM FIFO.
8899
8900      ;+ READ THE REMAINING AND VERIFY 1 OVERRUN PLUS 1 CHAR FROM EACH LINE.
8901      ;-
8902 031664 005004      CLR    R4            ;CLEAR THE OVERRUN ERROR FLAGS.
8903 031666 012700 003450      MOV    #RXCNTB,R0
8904 031672 004737 015524      JSR    PC,CLR16W     ;CLEAR RX CHAR COUNT TABLE.
8905 031676 012737 015064 005224 8$:      MOV    #6708.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
8906 031704 017702 150274      MOV    @RBUFA,R2     ;READ A CHARACTER FROM THE FIFO.
8907 031710 100047      BPL    14$           ;GO ANALYZE THE RESULTS IF ALL CHARS READ.
8908 031712 004737 014446      JSR    PC,CHKBMP     ;CHECK IF CHAR IS A BMP CODE.
8909 031716 103525      BCS    24$           ;REPORT ERROR AND ABORT TEST IF A BMP CODE.
8910 031720 005237 005224      INC    ERRNBR        ;SET ERROR NUMBER TO 6709.
8911 031724 010200      MOV    R2,R0
8912 031726 000300      SWAB   R0
8913 031730 042700 177760      BIC    #177760,R0    ;CALCULATE THE LINE NUMBER OF THE CHAR.
8914 031734 006300      ASL    R0            ;FORM WORD TABLE OFFSET FOR TABLE ACCESS.
8915 031736 042702 007400      BIC    #7400,R2      ;REMOVE LINE NUMBER FROM THE READ CHAR.
8916 031742 056037 002332 002174      BIT    BITTBL(R0),ACTLNS ;TEST FOR ACTIVE LINE.
8917 031750 001512      BEQ    50$           ;ABORT TEST IF FOR INACTIVE LINE.
8918 031752 005237 005224      INC    ERRNBR        ;SET ERROR NUMBER TO 6710.
8919 031756 005760 003450      TST    RXCNTB(R0)    ;CHECK THE RX CHAR COUNTER FOR THIS LINE.
8920 031762 001006      BNE    10$          ;IS THIS FIRST CHAR ON LINE?
8921 031764 020227 140000      CMP    R2,#140000   ;YES, TEST FOR NULL CHAR WITH OVERRUN.
8922 031770 001414      BEQ    12$          ;IS CHAR A NULL?
8923 031772 056004 002332      BIS    BITTBL(R0),R4 ;NO, SET THE OVERRUN BIT ERROR FLAG FOR LINE.
8924 031776 000411      BR     12$          ;GO COUNT THE CHAR AND CONTINUE.
8925 032000 026027 003450 000004 10$:      CMP    RXCNTB(R0),#4
8926 032006 002073      BGE    50$           ;5TH CHAR ON THIS LINE? YES, ABORT.
8927 032010 032702 040000      BIT    #BIT14,R2    ;NO, CHECK OVERRUN BIT.
8928 032014 001402      BEQ    12$          ;IS OVERRUN BIT CLEAR? YES, GO COUNT CHAR.
8929 032016 056004 002332      BIS    BITTBL(R0),R4 ;NO, SET THE OVERRUN BIT ERROR FLAG FOR LINE.
8930 032022 005260 003450 12$:      INC    RXCNTB(R0)    ;COUNT THIS CHARACTER.
8931 032026 000723      BR     8$           ;LOOP UNTIL ALL CHARS ARE READ FROM FIFO.
8932
8933      ;+ TEST FOR ABORT CONDITIONS. ONLY NONE ABORT CONDITIONS ARE:
8934      ; 1) 2 CHARS RXED ON A LINE AND NO OVERRUN ERROR BIT FAILURE DETECTED.
8935      ; 2) 2 TO 4 CHARS RXED ON A LINE AND AN OVERRUN BIT FAILURE DETECTED.
8936      ;-
8937 032030 005001 14$:      CLR    R1            ;INITIALIZE LINE LOOP, CLEAR LINE OFFSET.
8938 032032 012737 015067 005224 16$:      MOV    #6711.,ERRNBR ;SET UP ERROR NUMBER EACH TIME AROUND LOOP.
8939 032040 036137 002332 002174      BIT    BITTBL(R1),ACTLNS
8940 032046 001415      BEQ    18$           ;LINE ACTIVE? NO, NEXT LINE.
8941 032050 026127 003450 000002      CMP    RXCNTB(R1),#2 ;YES.
8942 032056 002447      BLT    50$           ;FEWER THAN 2 CHARS RXED? YES, ABORT.
8943 032060 036104 002332      BIT    BITTBL(R1),R4 ;NO.
8944 032064 001006      BNE    18$           ;OVERRUN BIT ERROR FLAG SET? YES, NEXT LINE.
8945 032066 005237 005224      INC    ERRNBR        ;SET LINE NUMBER TO 6712.
8946 032072 026127 003450 000002      CMP    RXCNTB(R1),#2
8947 032100 001036      BNE    50$           ;NOT 2 CHARS RXED? YES, ABORT. NO, NEXT LINE.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 214
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - ORERR -

```

8948 032102 062701 000002      18$:  ADD    #2,R1      ;SET LINE OFFSET TO THE NEXT LINE.
8949 032106 020127 000020      CMP    R1,#NUMLNS*2
8950 032112 002747              BLT    16$           ;ALL LINES DONE? NO, LOOP. YES, CONTINUE.
8951
8952      ;+
8953      ;: CHECK FOR OVERRUN ERROR BIT FAILURES, PRINT ERROR MESSAGE IF FOUND.
8954 032114 012737 015071 005224      MOV    #6713.,ERRNBR ;SET UP ERROR NUMBER.
8955 032122 005001              CLR    R1           ;INITIALIZE LOOP. CLEAR LINE OFFSET.
8956 032124 010102      20$:  MOV    R1,R2       ;COPY THE LINE OFFSET.
8957 032126 036104 002332      BIT    BITTBL(R1),R4 ;OVERRUN BIT FAILURE FLAGS ARE IN R4.
8958 032132 001411              BEQ    22$         ;ERROR FLAG CLEAR? YES, NEXT LINE.
8959 032134 010103              MOV    R1,R3
8960 032136 006203              ASR    R3         ;CALCULATE LINE NUMBER FROM LINE OFFSET.
8961 032140 012737 012660 005230      MOV    #ER7801,ERRBLK ;SELECT THE CORRECT ERROR REPORTING ROUTINE.
8962 032146 012701 010024      MOV    #EM6702,R1   ;PASS THE MESSAGE TO BE REPORTED.
8963      ;REPORT 'OVERRUN ERROR NOT REPORTED CORRECTLY WHEN FORCED, ON LINE NN ...'
8964 032152              ERROR          ;          >>>>> ERROR #6713 <<<<<.
8965 032152 104460              TRAP   C$ERROR
8966 032154 010201      22$:  MOV    R2,R1       ;RESTORE THE LINE OFFSET THAT WAS DESTROYED.
8967 032156 046104 002332      BIC    BITTBL(R1),R4 ;CLEAR THE LINE ERROR FLAG WE JUST HANDLED.
8968 032162 001407              BEQ    60$         ;ALL FAILURE BITS HANDLED? YES, EXIT TEST.
8969 032164 062701 000002      ADD    #2,R1       ;NO, INCREMENT THE LINE OFFSET.
8970 032170 000755              BR    20$         ;LOOP TO HANDLE THE NEXT LINE.
8971
8972 032172      24$:  ;REPORT 'BMP CODE FOUND IN FIFO, TEST INVAILEDATED.'
8973 032172              ERROR          ;          >>>>> ERROR <<<<<.
8974 032172 104460              TRAP   C$ERROR
8975 032174 000402              BR    60$         ;EXIT THIS TEST.
8976
8977 032176 004737 023074      50$:  JSR    PC,TSABRT  ;ABORT THE TEST. ERROR # INDICATES FAULT TYPE.
8978 032202 005037 002222      60$:  CLR    CTRLCF    ;INDICATE THAT WE ARE NOT WITHIN A TEST.
8979
8980 032206              ENDTST
8981 032206
8982 032206 104401              L10033: TRAP   C$ETST
    
```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 215
HARDWARE TEST - SINGLC -

8983
8984
8985
8986
8987
8988
8989
8990
8991
8992
8993
8994
8995
8996
8997
8998 032210
8999 032210
9000 032210
9001 032210 012700 000240
9002 032214 104441
9003 000007
9004 032216 012737 000007 002224
9005 032224 012737 177777 002222
9006 032232 012737 000001 005222
9007 032240 012737 021451 005224
9008 032246 012737 010101 005226
9009 032254 005037 002406
9010
9011
9012
9013
9014
9015 032260 004737 015502
9016 032264 103402
9017 032266 000137 032674
9018 032272 012737 021452 005224
9019
9020
9021
9022
9023 032300
9024 032300 012700 000340
9025 032304 104441
9026 032306
9027 032306 012746 000240
9028 032312 012746 025464
9029 032316 013746 002172
9030 032322 012746 000003
9031 032326 104437
9032 032330 062706 000010
9033 032334
9034 032334 012746 000240
9035 032340 012746 025144
9036 032344 013746 002170
9037 032350 012746 000003
9038 032354 104437

.SBTTL HARDWARE TEST - SINGLC -
:++ *****
: * - SINGLE CHARACTER MODE TEST -
: * THIS TEST VERIFIES THAT THE DEVICE UNDER TEST (DUT) WILL PERFORM
: * TRANSMISSION AND RECEPTION CORRECTLY USING THE SINGLE CHARACTER
: * MODE INTERRUPTS. THE TEST IS PERFORMED AT 3 BAUDRATES (SLOWEST,
: * MIDDLE, AND HIGHEST) AT ALL COMBINATIONS OF # OF STOP BITS, # OF
: * BITS PER CHARACTER, AND TYPES OF PARITY USING SHORT DATA PATTERNS.
: * A HIGH SPEED TEST IS ALSO PERFORMED AT THE HIGHEST BAUDRATE WITH ALL
: * COMBINATIONS OF LINE PARAMETERS USING LONGER DATA PATTERNS.
: * THIS TEST IS PERFORMED IN INTERNAL LOOPBACK REGARDLESS OF THE TYPE
: * OF LOOPBACK WHICH IS SELECTED FOR THE DUT IN THE HARDWARE P-TABLE.
:-- *****

BGNTST
T7::
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. (90)
MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
MOV #1,ERRTYP ;SET ERROR TYPE AS FATAL IN ERROR TABLE.
MOV #9001,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
MOV #EM9001,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERRTBL.
CLR ERSRPF ;INITIALIZE THE 'REPORT ERROR SUMMARY' FLAGS.

:+
: 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 >>>> 9001 <<<<.
:-

JSR PC,CLNRST ;RESET THE DHV-11, REPORT ANY ERRORS FOUND.
BCS .+6 ;IF NO ERROR FOUND DURING RESETT, DON'T ABORT.
JMP 60\$;ABORT THE TEST, FATAL ERROR WAS FOUND.
MOV #9002,ERRNBR ;SET THE ERROR NUMBER.

:+
: SET UP FOR TRANSMIT AND RECEIVE INTERRUPTS.
:-

SETPRI #PRI07 ;DISABLE ALL INTERRUPTS.
MOV #PRI07,R0
TRAP C\$SPRI
SETVEC TXVECA,#TXSCHR,#PRI05 ;SELECT SINGLE CHAR TX INT SERVICE RTN.
MOV #PRI05,-(SP)
MOV #TXSCHR,-(SP)
MOV TXVECA,-(SP)
MOV #3,-(SP)
TRAP C\$SVEC
ADD #10,SP
SETVEC RXVECA,#RXCHRS,#PRI05 ;SELECT RX INT SERVICE RTN.
MOV #PRI05,-(SP)
MOV #RXCHRS,-(SP)
MOV RXVECA,-(SP)
MOV #3,-(SP)
TRAP C\$SVEC

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 221
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - DMA -

```

9266 033250 105201          INCB  R1          ;GET THE NEXT BYTE FOR THE DATA PATTERN.
9267 033252 001375          BNE   6$          ;LOOP UNTIL FIRST 1/2 OF PATTERN IS DONE.
9268 033254 105301      8$:  DECB  R1          ;GET THE NEXT BYTE FOR THE DATA PATTERN.
9269 033256 110122          MOVB  R1,(R2)+    ;WRITE A BYTE OF THE DATA PATTERN.
9270 033260 105701          TSTB  R1          ;CHECK FOR DONE WRITING DATA PATTERN.
9271 033262 001374          BNE   8$          ;LOOP IF DATA PATTERN IS NOT DONE.
9272 033264 110122      10$: MOVB  R1,(R2)+    ;WRITE A BYTE OF THE 32 BYTE OVERFLOW REGION.
9273 033266 005201          INC   R1          ;COUNT THIS BYTE.
9274 033270 020127 000040    CMP   R1,#32.     ;TEST FOR 32 BYTES WRITTEN.
9275 033274 001373          BNE   10$        ;LOOP UNTIL 32 BYTES ARE WRITTEN.
9276
9277      :+
9278      : PREPARE TO LOOP ON THE 3 DIFFERENT BAUDRATES (9.6K, 19.2K, AND 38.4K).
9279 033276 012705 005000    MOV   #DLPRTB,R5 ;GET THE BASE ADR OF THE DMA BAUDRATE TABLE.
9280
9281      :+
9282      : SPECIFY THE PROPER BAUDRATE.
9283      : SPECIFY 8 BITS PER CHARACTER.
9284      : PERFORM DMA TRANSMISSION AND RECEPTION OF 512 BYTE DATA PATTERN.
9285      :-
9286      :+
9287      : THE FOLLOWING ROUTINE REPORTS THE ERROR WITH NUMBERS 914 THRU 921.
9288      : LPR CHANGE BIT ERROR FLAGS MAY BE SET BY THIS SUBROUTINE.
9289
9289 033302 012501      12$: MOV   (R5)+,R1    ;SET UP LPR PARAM AT NEXT BAUD, 8 BITS/CHAR.
9290 033304 004737 016746    JSR   PC,GETTIM  ;GET TIME-OUT BASED ON MINIMUM BAUDRATE IN USE.
9291 033310 012702 003510    MOV   #BUFBAS,R2 ;SET UP THE START ADR OF THE DATA PATTERN.
9292 033314 012703 001000    MOV   #512.,R3   ;SET UP THE DATA PATTERN LENGTH.
9293 033320 012704 000001    MOV   #1,R4      ;SPECIFY TO SEND 1 DATA PATTERN TO EACH LINE.
9294 033324 004737 024542    JSR   PC,VANSUP  ;SET UP 'VANILLA FLAVORED' TX/RX.
9295 033330 012737 177400 002226 MOV   #177400,IBM ;FORM BIT MAP OF UNUSED BITS FOR 8 BITS/CHAR.
9296 033336 012737 021633 005224 MOV   #9115.,ERRNBR ;SET ERROR NUMBER TO 9115.
9297
9298      :+
9299      : THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9115 THRU 9117 <<<<.
9300
9300 033344 004737 020674    JSR   PC,PUFIFR  ;PURGE THE DUT RECEIVE CHARACTER FIFO.
9301 033350 103072          BCC   60$        ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
9302 033352 012737 021636 005224 MOV   #9118.,ERRNBR ;SET ERROR NUMBER TO 9118.
9303
9304 033360 004737 021064    JSR   PC,PURRXB  ;PURGE THE RX CHAR BUFFER IN MEMORY.
9305 033364 004737 017166    JSR   PC,INIDMA  ;SEND THE FIRST BATCH OF DATA PATTERNS.
9306
9307      :+
9308      : THIS ROUTINE REPORTS THE ERROR WITH NUMBERS >>>> 9118 THRU 9123 <<<<.
9309
9309 033370 004737 021120      JSR   PC,RDCHRS  ;READ AND VERIFY THE RX CHARACTERS.
9310 033374 012737 021644 005224 MOV   #9124.,ERRNBR ;SET ERROR NUMBER TO 9124.
9311
9312      :+
9313      : THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9124 THRU 9127 <<<<.
9314
9314 033402 004737 024230      JSR   PC,IXRREP  ;REPORT FINAL ERRORS FROM RX/RX.
9315 033406 012737 021650 005224 MOV   #9128.,ERRNBR ;SET ERROR NUMBER TO 9128.
9316
9317      :+
9318      : SPECIFY 5 BITS PER CHARACTER.
9319      : PERFORM DMA TRANSMISSION AND RECEPTION OF 512 BYTE DATA PATTERN.
9320
9320 033414 042701 000030      BIC   #30,R1     ;SET UP CHAR LENGTH PARAM TO 5 BITS/CHAR.
9321 033420 004737 024542      JSR   PC,VANSUP  ;SET UP 'VANILLA FLAVORED' TX/RX.

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 222
CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - DMA -

```

9322 033424 012737 177740 002226      MOV    #177740,IBM      ;FORM BIT MAP OF UNUSED BITS FOR 5 BITS/CHAR.
9323                                     :+
9324                                     :+ THIS ROUTINE REPORTS THE ERROR WITH NUMBERS >>> 9128 THRU 9131 <<<.
9325                                     :-
9326 033432 004737 020674      JSR    PC,PUFIFR      ;PURGE THE DUT RECEIVE CHARACTER FIFO.
9327 033436 103037             BCC    60$            ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
9328 033440 012737 021654 005224      MOV    #9132.,ERRNBR  ;SET THE ERROR REPORT NUMBER TO 9132.
9329
9330 033446 004737 021064      JSR    PC,PURRXB      ;PURGE THE RX CHAR BUFFER IN MEMORY.
9331 033452 004737 017166      JSR    PC,INIDMA      ;SEND THE FIRST BATCH OF DATA PATTERNS.
9332                                     :+
9333                                     :+ THIS ROUTINE REPORTS THE ERROR WITH NUMBERS >>>> 9132 THRU 9137 <<<<.
9334                                     :-
9335 033456 004737 021120      JSR    PC,RDCHRS      ;READ AND VERIFY THE RX CHARACTERS.
9336 033462 012737 021662 005224      MOV    #9138.,ERRNBR  ;SET ERROR NUMBER TO 9138.
9337                                     :+
9338                                     :+ THIS ROUTINE REPORTS THE ERROR WITH NUMBERS >>>> 9138 THRU 9141 <<<<.
9339                                     :-
9340 033470 004737 024230      JSR    PC,TXRREP      ;REPORT FINAL ERRORS FROM RX/RX.
9341 033474 020527 005006      CMP    R5,#DLP RTE    ;COMPARE DMA BAUDRATE TABLE PTR WITH TABLE END.
9342 033500 103700             BLO    12$            ;LOOP IF NOT ALL BAUDRATES DONE YET.
9343                                     :+
9344                                     :+ ALL DONE. HAVE EITHER RUN OUT OF ACTIVE LINES, OR COMPLETED THE TEST.
9345                                     :+ DISABLE INTERRUPTS.
9346                                     :+ CLEAR THE INTERRUPT VECTORS.
9347                                     :-
9348 033502             SETPRI #PRI07          ;DISABLE ALL INTERRUPTS.
9349 033502 012700 000340             MOV    #PRI07,RO
9350 033506 104441             TRAP  C$SPRI
9351 033510             CLRVEC TXVECA        ;RETURN TX INT VECTOR TO UNUSED POOL.
9352 033510 013700 002172             MOV    TXVECA,RO
9353 033514 104436             TRAP  C$CVEC
9354 033516             CLRVEC RXVECA        ;RETURN RX INT VECTOR TO UNUSED POOL.
9355 033516 013700 002170             MOV    RXVECA,RO
9356 033522 104436             TRAP  C$CVEC
9357
9358 033524 012737 021666 005224      MOV    #9142.,ERRNBR  ;SELECT NUMBER 9142 FOR THE NEXT ERROR REPORT.
9359 033532 004737 021664      JSR    PC,REPSMR      ;REPORT ERROR SUMMARIES IF CALLED FOR.
9360 033536             SETPRI #PRI07          ;DISABLE ALL INTERRUPTS.
9361 033536 012700 000340             MOV    #PRI07,RO
9362 033542 104441             TRAP  C$SPRI
9363 033544 005037 002222      CLR    CTRLCF        ;INDICATE THAT WE ARE NOT WITHIN A TEST.
9364 033550             ENDTST
9365 033550             L10035:
9366 033550 104401             TRAP  C$SETST

```

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 223
HARDWARE TEST - SPLSPD -

```

9367 .SBTTL HARDWARE TEST - SPLSPD -
9368 :*****
9369 :* - SPLIT SPEED TEST -
9370 :* THIS TEST IS USED TO VERIFY THE SPLIT SPEED CAPABILITIES OF THE DHV11,
9371 :* AND THE CORRECT OPERATION OF THE A & B BAUD RATE GROUP SELECTION.
9372 :* THE TEST USES THREE SETS OF BAUD RATES (38.4,50; 1200,75; 2000,2400).
9373 :* THIS TEST WILL ONLY EXECUTE IF THE STAGGARED LOOPBACK MODE IS SELECTED.
9374 :* THE SPECIAL STAGGARED LOOPBACK BERG CONNECTOR MUST BE FITTED.
9375 :*
9376 :*****
9377 033552 BGNST
9378 033552 T9::
9379 033552 123727 002176 000002 CMPB LOPBCK,#2 ;CHECK MODE SELECTED.
9380 033560 001402 BEQ 2$ ;DO NOT EXIT IF STAGGERD LOPBCK MODE SELECTED.
9381 033562 000137 034204 JMP 60$ ;EXIT THIS TEST.
9382 033566 2$: SETPRI #PRI05 ;ALLOW LTC INTERRUPTS.
9383 033566 012700 000240 MOV #PRI05,R0
9384 033572 104441 TRAP C$SPRI
9385 000011 TNUM == TNUM + 1 ;INCREMENT THE ASSEMBLY TIME TEST COUNTER.
9386 033574 012737 000011 002224 MOV #TNUM,TSTNUM ;SET UP THE TEST NUMBER. (92)
9387 033602 012737 177777 002222 MOV #-1,CTRLCF ;INDICATE THAT WE ARE IN A TEST.
9388 033610 012737 000001 005222 MOV #1,ERRTYP ;SET ERROR TYPE IN ERROR TABLE.
9389 033616 012737 021761 005224 MOV #9201,ERRNBR ;SET THE FIRST ERROR NUMBER IN ERROR TABLE.
9390 033624 012737 011476 005226 MOV #EM9201,ERRMSG ;SET ERROR MESSAGE ADDRESS IN ERROR TABLE.
9391 033632 005037 002406 CLR ERSMRF ;INITIALIZE THE 'REPORT ERROR SUMMARY' FLAGS.
9392 :+
9393 : RESET THE DUT TO A KNOWN STATE, REMOVE STATUS CODES FROM THE FIFO.
9394 : CLEAR TX AND RX INTERRUPT ENABLE BITS.
9395 : THIS SUBROUTINE REPORTS ERROR >>>> 9201 <<<<<.
9396 :-
9397 033636 004737 015502 JSR PC,CLNRST ;RESET THE DUT.
9398 033642 103160 BCC 60$ ;ABORT THE TEST IF FATAL ERROR FOUND IN RESET.
9399 :+
9400 : DISABLE ALL INTERRUPTS.
9401 : SET UP DMA TX AND RX INTERRUPT SERVICE ROUTINES.
9402 :-
9403 033644 SETPRI #PRI07 ;DISABLE ALL INTERRUPTS.
9404 033644 012700 000340 MOV #PRI07,R0
9405 033650 104441 TRAP C$SPRI
9406 033652 SETVEC TXVECA,#TXDMA,#PRI05 ;SELECT DMA TX INT SERVICE RTN.
9407 033652 012746 000240 MOV #PRI05,-(SP)
9408 033656 012746 025332 MOV #TXDMA,-(SP)
9409 033662 013746 002172 MOV TXVECA,-(SP)
9410 033666 012746 000003 MOV #3,-(SP)
9411 033672 104437 TRAP C$SVEC
9412 033674 062706 000010 ADD #10,SP
9413 033700 SETVEC RXVECA,#RXCHRS,#PRI05 ;SELECT RX INT SERVICE RTN.
9414 033700 012746 000240 MOV #PRI05,-(SP)
9415 033704 012746 025144 MOV #RXCHRS,-(SP)
9416 033710 013746 002170 MOV RXVECA,-(SP)
9417 033714 012746 000003 MOV #3,-(SP)
9418 033720 104437 TRAP C$SVEC
9419 033722 062706 000010 ADD #10,SP
9420 033726 SETPRI #PRI03 ;ALLOW INTERRUPTS.
9421 033726 012700 000140 MOV #PRI03,R0
9422 033732 104441 TRAP C$SPRI

```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 224
HARDWARE TEST - SPLSPD -

```

9423      ;+
9424      ;: ENABLE TRANSMITTERS ON ALL LINES.
9425      ;-
9426      033734 012705 000377      MOV      #MAPLNS,R5      ;PASS ACTIVE LINE BIT MAP.
9427      033740 004737 023412      JSR      PC,TXENBL      ;ENABLE TRANSMISSIONS ON ALL LINES.
9428
9429      ;+
9430      ;: CLEAR ERROR TABLE PRIOR TO PERFORMING TX/RX TEST.
9431      ;-
9432      033744 012700 003210      MOV      #ERCNTB,R0      ;GET THE BASE ADDRESS OF THE ERROR COUNTER TBL.
9433      033750 004737 015524      JSR      PC,CLR16W      ;CLEAR THE RX ERROR COUNTERS TABLE.
9434
9435      ;+
9436      ;: PERFORM SPLIT SPEED DMA TX AND RX ON ALL SELECTED LINES AT THE FOLLOWING
9437      ;: BAUD RATES.
9438      ;: 38.4K, 50 ; 1200, 75 ; 2000, 2400.
9439      ;-
9440      ;+
9441      ;: INITIALISE DMA TX/RX PARAMETERS IN THE CONTROL BLOCK FR EACH OF THE BAUD
9442      ;: RATES MENTIONED ABOVE.
9443      ;: 8 BITS/CHAR,1 STOP BITS,ODD PARITY.
9444      ;-
9445      033754 012705 005006      MOV      #SPLPRB,R5      ;GET BASE ADDRESS OF LPR PARAMETER TABLE.
9446      033760 012500      4$: MOV      (R5)+,R0      ;GET LPR CONTENTS FOR LINGRP II.
9447      033762 012501      MOV      (R5)+,R1      ;GET LPR CONTENTS FOR LINGRP I.
9448      033764 004737 016746      JSR      PC,GETTIM      ;GET TIME-OUT BASED ON MINIMUM BAUDRATE IN USE.
9449      033770 012702 005062      MOV      #SDP2B,R2      ;SET UP THE START ADR OF THE DATA PATTERN.
9450      033774 012503      MOV      (R5)+,R3      ;GET NUMBER OF REPEAT TRANSMISSION ON LINGRP II.
9451      033776 012504      MOV      (R5)+,R4      ;GET NUMBER OF REPEAT TRANSMISSION ON LINGRP I.
9452      034000 004737 022524      JSR      PC,SPLSUP      ;SET UP CONTROL BLOCK ETC, FOR TX/RX.
9453      034004 012737 021762 005224      MOV      #9202.,ERRNBR ;SET THE ERROR NUMBER TO 9202.
9454
9455      ;+
9456      ;: THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9202 THRU 9204 <<<<.
9457      ;-
9458      034012 004737 020674      JSR      PC,PUFIFR      ;PURGE THE DUT RECEIVE CHARACTER FIFO.
9459      034016 103072      BCC     60$            ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
9460      034020 012737 021765 005224      MOV      #9205.,ERRNBR ;SET ERROR NUMBER TO 9205.
9461
9462      034026 004737 021064      JSR      PC,PURRXB      ;PURGE THE RX CHAR BUFFER IN MEMORY.
9463      034032 004737 017166      JSR      PC,INIDMA      ;SEND THE FIRST BATCH OF DATA PATTERNS.
9464
9465      ;+
9466      ;: THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9205 THRU 9210 <<<<.
9467      ;-
9468      034036 004737 021120      JSR      PC,RDCHRS      ;READ AND VERIFY THE RX CHARACTERS.
9469      034042 012737 021773 005224      MOV      #9211.,ERRNBR ;SET THE ERROR NUMBER TO 9211.
9470
9471      ;+
9472      ;: THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9211 THRU 9214 <<<<.
9473      ;-
9474      034050 004737 024230      JSR      PC,TXRREP      ;REPORT FINAL ERRORS FROM RX/RX.
9475      034054 012737 021777 005224      MOV      #9215.,ERRNBR ;SET ERROR NUMBER TO 9215.
9476
9477      ;+
9478      ;: SWAP PARAMETERS TO ALLOW FOR BOTH CHANNELS TO BE EXERCISED.
          ;-
          MOV      R2,-(SP) ;PUSH THE START ADDRESS ONTO THE STACK.
          MOV      R0,R2
          MOV      R1,R0

```

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 225
 CVDHCA.P11 12-JUL-83 11:44 HARDWARE TEST - SPLSPD -

```

9479 034070 010201          MOV    R2,R1          ;SWAP THE TWO SETS OF
9480 034072 010302          MOV    R3,R2          ;PARAMETERS OVER.
9481 034074 010403          MOV    R4,R3          ;
9482 034076 010204          MOV    R2,R4          ;
9483 034100 012602          MOV    (SP)+,R2       ;RESTORE THE START ADDRESS.
9484 034102 004737 022524  JSR    PC,SPLSUP      ;SET UP CONTROL BLOCK ETC, FOR TX/RX.
9485
9486
9487      :+
9488      :- THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9215 THRU 9217 <<<<.
9489 034106 004737 020674  JSR    PC,PUFIFR      ;PURGE THE DUT RECEIVE CHARACTER FIFO.
9490 034112 103034          BCC    60$           ;ABORT THIS TEST IF FIFO WOULD NOT PURGE.
9491 034114 012737 022002 005224  MOV    #9218.,ERRNBR ;SET ERROR NUMBER TO 9218.
9492
9493 034122 004737 021064  JSR    PC,PURRXB      ;PURGE THE RX CHAR BUFFER IN MEMORY.
9494 034126 004737 017166  JSR    PC,INIDMA      ;SEND THE FIRST BATCH OF DATA PATTERNS.
9495
9496      :+
9497      :- THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9218 THRU 9223 <<<<.
9498 034132 004737 021120  JSR    PC,RDCHRS      ;READ AND VERIFY THE RX CHARACTERS.
9499 034136 012737 022010 005224  MOV    #9224.,ERRNBR ;SET ERROR NUMBER TO 9224.
9500
9501      :+
9502      :- THIS ROUTINE REPORTS ERRORS WITH NUMBERS >>>> 9224 THRU 9227 <<<<.
9503 034144 004737 024230  JSR    PC,TXRREP      ;REPORT FINAL ERRORS FROM RX/RX.
9504 034150 020527 005036  CMP    R5,#SPLPRE    ;CHECK IF ALL PARAMETERS HAVE BEEN DONE.
9505 034154 103701          BLO    4$           ;IF NOT DONE LOOP TO SELECT THE NEXT PARAMETER.
9506
9507      :+
9508      :- DISABLE INTERRUPTS.
9509      :- CLEAR THE INTERRUPT VECTORS.
9510
9511 034156          SETPRI #PRI07        ;DISABLE ALL INTERRUPTS.
9512 034156 012700 000340  MOV    #PRI07,RO
9513 034162 104441          TRAP  C$SPRI
9514 034164          CLRVEC TXVECA      ;RETURN TX INT VECTOR TO UNUSED POOL.
9515 034170 013700 002172  MOV    TXVECA,RO
9516          TRAP  C$CVFC
9517 034172 012737 022014 005224  MOV    #9228.,ERRNBR ;SELECT NUMBER 9228 FOR THE NEXT ERROR REPORT.
9518 034200 004737 021664  JSR    PC,REPSMR      ;REPORT ERROR SUMMARIES IF CALLED FOR.
9519 034204 005037 002222 60$: CLR    CTRLCF       ;INDICATE THAT WE ARE NOT WITHIN A TEST.
9520 034210          ENDTST
9521 034210
9522 034210 104401          L10036: TRAP  C$ETST
    
```


CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 227
HARDWARE TEST - REPBMP -

9560
9561
9562
9563
9564
9565
9566
9567
9568
9569
9570
9571
9572
9573
9574
9575
9576
9577
9578
9579
9580
9581
9582
9583
9584
9585
9586
9587
9588
9589
9590
9591
9592
9593
9594
9595
9596
9597
9598
9599
9600
9601
9602
9603
9604
9605
9606
9607
9608
9609
9610
9611
9612
9613
9614
9615

.SBTTL HARDWARE PARAMETER CODING SECTION

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

BGNHRD

000027

.WORD L10040-LSHARD/2
LSHARD::

;DEVICE CSR ADDRESS QUESTION:
GPRMA HWPTQ1,0,0,160000,177776,YES

.WORD TSCODE
.WORD HWPTQ1
.WORD TSLOLIM
.WORD TSHILIM

;DEVICE INTERRUPT VECTOR QUESTION:
GPRMA HWPTQ2,2,0,40,776,YES

.WORD TSCODE
.WORD HWPTQ2
.WORD TSLOLIM
.WORD TSHILIM

;ACTIVE LINES BIT MAP QUESTION:
GPRMD HWPTQ3,4,0,MAPLNS,0,177777,YES

.WORD TSCODE
.WORD HWPTQ3
.WORD MAPLNS
.WORD TSLOLIM
.WORD TSHILIM

;TYPE OF LOOPBACK QUESTION:
GPRMD HWPTQ4,6,0,377,1,3,YES

.WORD TSCODE
.WORD HWPTQ4
.WORD 377
.WORD TSLOLIM
.WORD TSHILIM

;INTERRUPT BR LEVEL QUESTION:
GPRMD HWPTQ5,6,0,177400,0,6,YES

.WORD TSCODE
.WORD HWPTQ5
.WORD 177400
.WORD TSLOLIM
.WORD TSHILIM

ENDHRD

.EVEN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 229
HARDWARE PARAMETER CODING SECTION

9653
9654
9655
9656
9657
9658
9659
9660
9661
9662
9663
9664
9665
9666
9667
9668
9669
9670
9671
9672
9673
9674
9675
9676
9677
9678
9679
9680
9681
9682
9683
9684
9685
9686
9687
9688
9689
9690
9691
9692
9693
9694
9695
9696
9697
9698
9699
9700
9701
9702
9703
9704
9705
9706
9707
9708

034630
034630 000010
034632

034632
034632 000130
034634 034652
034636 000020

034640
034640 001052
034642 034726
034644 177777
034646 000000
034650 177777

034652
034652

034652 042522 047520 052122
034660 052440 044516 020124
034666 052516 041115 051105
034674 040440 020123 040505
034702 044103 052440 044516
034710 020124 051511 052040
034716 051505 042524 035104
034724 000040
034726 052516 041115 051105
034734 047440 020106 047111
034742 044504 044526 052504
034750 046101 042040 052101
034756 020101 051105 047522
034764 051522 052040 020117
034772 042522 047520 052122
035000 047440 020116 020101
035006 044514 042516 020072
035014 000
035016

.SBTTL SOFTWARE PARAMETER CODING SECTION

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

BGNSFT

.WORD L10041-L\$SOFT/2
L\$SOFT::

:UNIT NUMBER PRINTOUT QUESTION:
GPRML SWPTQ1,0,20,YES

.WORD T\$CODE
.WORD SWPTQ1
.WORD 20

:NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE QUESTION:
GPRMD SWPTQ2,2,D,177777,0,177777,YES

.WORD T\$CODE
.WORD SWPTQ2
.WORD 177777
.WORD T\$LLOLIM
.WORD T\$HILIM

.EVEN

ENDSFT

.EVEN
L10041:

SWPTQ1: .ASCIZ /REPORT UNIT NUMBER AS EACH UNIT IS TESTED: /

SWPTQ2: .ASCIZ /NUMBER OF INDIVIDUAL DATA ERRORS TO REPORT ON A LINE: /

.EVEN

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 230
SOFTWARE PARAMETER CODING SECTION

```

9709
9710
9711 035016
9712 035016 000024
9713
9714
9715
9716
9717
9718 035066
9719
9720 035066 000000
9721 035070 000000
9722 035072
9723 035072
9724
9725
9726
9727
9728
9729
9730
9731 000001

```

SPATCH:: .BLKW 24

LASTAD .EVEN 0
.WORD 0
.WORD 0

LSLAST:: ENDMOD

.END

CVD
CVD
TXD
TXD
TXD
TXD
TXC
TXE
TXE
TXF
TXI
TXI
TXI
TXI
TXF
TXF
TXF
TXF
TXF
TXF
TXS
TXV
TSA

TSA
TSA
TSA
TSA
TSA
TSA
TSA
TSA
TSA

TSA
TSA

TSA
TSA

TS
TS
TS
TS

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 233
CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- USER SYMBOLS

CBOFSA	003046	G	1414#	6854																
CHCNTB	003350	G	1424#	2975	3400	3461	6475	6828*	7150	7566	7669									
CHKBMP	014446	G	3340#	8748	8895	8908														
CHKEXT	014516	G	3386#	3609																
CHKLOS	014616	G	3456#	3628																
CHMSK	014720	G	3522#	9066	9114															
CHRTOT	002404	G	1383#	3729*	5558*	6829*	6831*	6833*	7195*											
CKCHR	014756	G	3572#	5015																
CKFROR	015174	G	3712#	8266	8284	8410	8435													
CKINAC	015344	G	3796#	4975																
CKTRAP	015452	G	3867#	7832	7853	8100	8107	8118	8123											
CLKBRL	002262	G	1324#	7797*																
CLKCSR	002260	G	1323#	7796*	7831															
CLKHRZ	002266	G	1326#	3258*	7799*	7800	7812	7835*												
CLKINT	025074	G	7401#	7807																
CLKVEC	002264	G	1325#	7798*	7808															
CLNRST	015502	G	3906#	8205	8349	8490	8655	8810	9015	9184	9397									
CLR16W	015524	G	3940#	8236	8378	8904	9048	9216	9433											
CONMAP	015546	G	3969#	4275	6327	6990	7245													
CSRA	002202	G	1293#	3197*	4094*	4582*	5271	5804	5817*	5981*	6036*	6084*	6111*	6183	6528*					
			6584*	6711*	6738*	7542	7552	7600*	7640	7654	7686*	7703*	7890*	7905	8098					
			8106	8115	8691	8724	8847	8881												
CSRO =	000000	G	1183#																	
CTRLCF	002222	G	1305#	7985	8080*	8155*	8164*	8193*	8306*	8337*	8461*	8479*	8621*	8646*	8776*					
			8801*	8978*	9005*	9149*	9174*	9363*	9387*	9519*	9537*	9556*								
CSAU =	000052		995#	8060																
CSAUTO=	000061		995#	7971																
CSBRK =	000022		995#	5087	7412															
CSBSEG=	000004		995#																	
CSBSUB=	000002		995#																	
CSCEFG=	000045		995#																	
CSCLCK=	000062		995#	7794																
CSCLEA=	000012		995#	7999																
CSCLOS=	000035		995#																	
CSCLP1=	000006		995#																	
CSCVEC=	000036		995#	8300	8453	9145	9148	9353	9356	9515										
CSDCLN=	000044		995#	8157																
CSDODU=	000051		995#	8154																
CSDRPT=	000024		995#																	
CSDU =	000053		995#	8038																
CSEDIT=	000003		995#	1061																
CSERDF=	000055		995#	8147	8558	8614	9549													
CSERHR=	000056		995#																	
CSERRO=	000060		995#	3739	5010	5028	5163	5218	5238	5408	5423	5568	5659	5717	5766					
			5837	5893	5941	6413	8751	8768	8965	8974										
CSERSF=	000054		995#	5075																
CSERSO=	000057		995#																	
CSESCA=	000010		995#																	
CSSEEG=	000005		995#																	
CSSESUB=	000003		995#																	
CSSTST=	000001		995#	8167	8310	8464	8624	8779	8982	9153	9366	9522	9559							
CSEXIT=	000032		995#	7991																
CSGETB=	000026		995#																	
CSGETW=	000027		995#																	
CSGMAN=	000043		995#																	
CSGPHR=	000042		995#	7882																

CVD
CVD
WTW
WTW
XSA
XSF
XSO
XST
SPA

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 237
CROSS REFERENCE TABLE -- USER SYMBOLS

F\$INIT=	000006	995#	7760	7951											
F\$JMP =	000050	995#	7720	7991	8032	8052									
F\$MOD =	000000	995#	1006	9724											
F\$MSG =	000011	995#	2326	2355	2403	2414	2438	2462	2492	2559	2588	2617	2641	2653	
		2676	2701	2726	2766	2791	2847	2871	2900	2926	2998	3028	3062	3085	
		3139													
F\$PROT=	000^21	995#	7737	7744											
F\$PWR =	000^17	995#													
F\$RPT =	000012	995#	7717	7726											
F\$SEG =	000003	995#													
F\$SOFT=	000005	995#	9667	9686											
F\$SRV =	000010	995#													
F\$SUB =	000002	995#													
F\$SW =	000014	995#	1159	1167											
F\$TEST=	000001	995#	8078	8166	8185	8309	8329	8463	8478	8623	8641	8778	8796	8981	
		9000	9152	9169	9365	9379	9521	9535	9558						
GETCHR	016362	G	4310#	5531	5600										
GETLP1	016444	G	4363#	9057											
GETLP2	016616	G	4435#	9105											
GETPRM	026270		7784	7871	7875#	7886									
GETT. #	016746	G	4490#	8245	8385	9061	9109	9218	9290	9448					
GPRSOB	002372	G	1374#	6369	9054*	9102*									
G\$CNTO=	000200		995#												
G\$DELM=	000372		995#												
G\$DISP=	000003		995#												
G\$EXCP=	000400		995#												
G\$HILI=	000002		995#												
G\$LOLI=	000001		995#												
G\$NO =	000000		995#												
G\$OFFS=	000400		995#	9581	9587	9593	9600	9607	9672	9677					
G\$OFSI=	000376		995#	9581	9587	9593	9600	9607	9672	9677					
G\$PRMA=	000001		995#	9581	9587										
G\$PRMD=	000002		995#	9593	9600	9607	9677								
G\$PRML=	000000		995#	9672											
G\$RADA=	000140		995#												
G\$RADB=	000000		995#												
G\$RADD=	000040		995#	9677											
G\$RADL=	000120		995#	9672											
G\$RADO=	000020		995#	9581	9587	9593	9600	9607							
G\$XFER=	000004		995#												
G\$YES =	000010		995#	9581	9587	9593	9600	9607	9672	9677					
HELP =	000000		1#	988	995	1000	1014	1102	1123	1147	1168	1171	1199	1269	1638
			1824	1841	1853	1854	1860	1862	2293	3142	7707	7729	7745	7953	7968
			7972	7984	7994	8000	8011	8039	8051	9560	9579	9618	9653	9670	9690
			9709	9714											
HOE =	100000	G	1267#												
HUPTQ1	034352		9582	9619#											
HUPTQ2	034370		9588	9622#											
HUPTQ3	034423		9594	9627#											
HUPTQ4	034451		9601	9631#											
HUPTQ5	034603		9608	9647#											
IBE =	010000	G	1264#												
IBM =	002226	G	1307#	3532*	4589	5505	7693	9223*	9295*	9322*					
IDU =	000040	G	1257#												
IER =	020000	G	1265#												
IESTAT	002234	G	1310#	3187	4093	4577	5274	5844*	5976	6031	6083*	6084	6109*	6110*	6111

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 239
 CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- USER SYMBOLS

LSDU	026540	G	1078	8010#		
LSDUT	002072	G	1077#			
LSDVTY	005272	G	1068	1836#		
LSEF	002052	G	1062#			
LSENV1	002044	G	1055#			
LSERRT	005222	G	1086	1639#		
LSETP	002102	G	1085#			
LSEXP1	002046	G	1057#			
LSEXP4	002064	G	1071#			
LSEXP5	002066	G	1073#			
LSHARD	034274	G	1034	9576	9577#	
LSHIME	002120	G	1099#			
LSHPCP	002016	G	1033#			
LSHPTP	002022	G	1037#			
LSHW	002152	G	1038	1135	1136#	
LSICP	002104	G	1087#			
LSINIT	025666	G	1088	7760#		
LSLADP	002026	G	1041#			
LSLAST	035072	G	1042	9722#		
LSLOAD	002100	G	1083#			
LSLUN	002074	G	1079#			
LSMREV	002050	G	1059#			
LSNAME	002000	G	1016#			
LSPRIO	002042	G	1053#			
LSPROT	025660	G	1094	7737#		
LSPRT	002112	G	1093#			
LSREPP	002062	G	1069#			
LSREV	002010	G	1025#			
LSRPT	025652	G	1070	7717#		
LSOFT	034632	G	1036	9667	9668#	
LSGPC	002056	G	1065#			
LSGPCP	002020	G	1035#			
LSPTP	002024	G	1039#			
LSSTA	002030	G	1043#			
LSW	002164	G	1040	1159	1160#	
LSTEST	002114	G	1095#			
LSTIML	002014	G	1031#			
LSUNIT	002012	G	1029#	7877	7931	
L10000	002162		1135	1146#		
L10001	002170		1159	1167#		
L10002	012014		2355#			
L10003	012252		2414#			
L10004	012332		2462#			
L10005	012560		2559#			
L10006	012656		2617#			
L10007	012704		2653#			
L10010	012774		2701#			
L10011	013142		2766#			
L10012	013324		2847#			
L10013	013426		2900#			
L10014	013654		2998#			
L10015	013766		3062#			
L10016	014146		3139#			
L10017	025656		7721	7726#		
L10021	026516		7951#			
L10022	026520		7970#			

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 243
CROSS REFERENCE TABLE -- USER SYMBOLS

3035	3036	3037	3038	3044	3045	3046	3047	3048	3049	3054	3055	3056
3057	3058	3063	3090	3091	3092	3093	3094	3095	3117	3118	3119	3120
3121	3126	3127	3128	3129	3130	3131	3132	3133	3140	3739	4088	4089
4091	4092	4102	4103	5010	5028	5075	5076	5077	5078	5081	5082	5083
5084	5085	5087	5163	5218	5238	5279	5280	5281	5282	5283	5284	5285
5408	5423	5546	5547	5549	5550	5554	5555	5568	5659	5717	5766	5837
5893	5941	6078	6079	6081	6082	6086	6087	6413	6705	6706	6708	6709
6713	6714	6908	6909	6911	6912	6944	6945	7412	7720	7721	7727	7763
7764	7766	7769	7770	7772	7775	7776	7778	7781	7782	7784	7788	7793
7794	7795	7806	7807	7808	7809	7810	7811	7816	7817	7861	7881	7882
7883	7885	7934	7935	7936	7937	7938	7939	7947	7948	7952	7971	7988
7991	7992	7999	8012	8013	8014	8015	8016	8017	8032	8033	8038	8052
8053	8060	8147	8148	8149	8150	8153	8154	8157	8167	8191	8192	8212
8213	8215	8216	8217	8218	8219	8220	8222	8223	8296	8297	8299	8300
8310	8335	8336	8356	8357	8359	8360	8361	8362	8363	8364	8366	8367
8449	8450	8452	8453	8464	8558	8559	8560	8561	8614	8615	8616	8617
8624	8642	8643	8751	8768	8779	8797	8798	8965	8974	8982	9001	9002
9024	9025	9027	9028	9029	9030	9031	9032	9034	9035	9036	9037	9038
9039	9041	9042	9141	9142	9144	9145	9147	9148	9153	9170	9171	9191
9192	9194	9195	9196	9197	9198	9199	9201	9202	9203	9204	9205	9206
9208	9209	9349	9350	9352	9353	9355	9356	9361	9362	9366	9383	9384
9404	9405	9407	9408	9409	9410	9411	9412	9414	9415	9416	9417	9418
9419	9421	9422	9511	9512	9514	9515	9522	9549	9550	9551	9552	9559
9576	9581	9582	9583	9584	9587	9588	9589	9590	9593	9594	9595	9596
9597	9600	9601	9602	9603	9604	9607	9608	9609	9610	9611	9615	9667
9672	9673	9674	9677	9678	9679	9680	9681	9686	9719	9720	9721	

SVCSUB= 000001
SVCTAG= 000001

SVCTST= 000001
SWAPO 023024 G
SWPTQ1 034652
SWPTQ2 034726
S&LSYM= 010000

TIMER1 002270 G
TIMER2 002272 G
TIMER3 002274 G
TNUM = 000012

TP4FLG 002256 G
TP4RTN 025310 G
TP4VEC 002254 G
TSABRT 023074 G
TSTNUM 002224 G
TXAD1A 002214 G
TXAD10= 000012 G
TXAD2A 002216 G
TXAD20= 000014 G
TXBFCA 002220 G
TXBFCA= 000016 G
TXCHA 002204 G
TXCHRO= 000002 G
TXCNTB 003410 G

995#	997#											
995#	999#	1146	1167	2355	2414	2462	2559	2617	2653	2701	2766	2847
2900	2998	3062	3139	7726	7951	7970	7998	8037	8059	8166	8309	8463
8623	8778	8981	9152	9365	9521	9558	9616	9687				
995#	996#	8077	8184	8328	8477	8640	8795	8999	9168	9378	9534	
6357#	9056	9058	9104	9106								
9673	9690#											
9678	9698#											
995#	1147#	1168#	2356#	2415#	2463#	2560#	2618#	2654#	2702#	2767#	2848#	2901#
2999#	3063#	3140#	7727#	7952#	7971#	7999#	8038#	8060#	8167#	8310#	8464#	8624#
8779#	8982#	9153#	9366#	9522#	9559#	9617#	9688#					
1327#	3247*	3248	3265	7401	7403*							
1328#	7404	7406*										
1329#	7407*	7409*										
8078#	8079	8194#	8195	8338#	8339	8480#	8481	8644#	8645	8799#	8800	9003#
9004	9172#	9173	9385#	9386	9535#	9536						
1319#	3869*	3871	7509*	7828*	7847*	8097*						
7506#	7824	7846	8085									
1318#	7508	7823*	7833	7845*	7854	8084*	8142					
6408#	8620	8775	8977									
1306#	6139	8079*	8195*	8339*	8481*	8645*	8800*	9004*	9173*	9386*	9536*	
1298#	4099*											
1189#												
1299#	4095	4100*	6520	6576								
1190#												
1300#	4098*	6186										
1191#												
1294#	4591*	7695*										
1185#												
1425#	4592*	4668*	5933	6674*	6843*	7566	7591*	7669	7697*			

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 244
CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- USER SYMBOLS

TXDBLF	002414	G	1387#	5506*	5537	5552*	7466	7473*							
TXDMA	025332	G	7540#	8216	8360	9195	9408								
TXDONE	023206	G	5607	6448#											
TXDONF	002410	G	1385#	3721	5523	5926	6247*	6457	6485	7197*	7568*	7671*	8227*	8371*	8417*
TXDSBL	023316	G	6516#	6914	7470										
TXENBL	023412	G	5551	6572#	6810	6942	8514	9427							
TXENBM	002250	G	1316#	5542	7471*										
TXFRPR	023506	G	6637#	8261	8279										
TXIEO	023602	G	6487	6703#	8292	8442	8497								
TXIE1	023642	G	3715	5508	6736#										
TXINTF	002252	G	1317#	4663*	5648	6669*	7196*	7585*	8229*	8373*	8419*				
TXPTRB	003250	G	1422#	4583	4588*	4654	6660	6844*	7576	7687	7692*				
TXRINI	023666	G	4220	4234	4252	4263	6268	6287	6305	6315	6797#	7223	7234		
TXROFF	024142	G	5008	5026	5715	6905#									
TXRON	024202	G	5011	5029	5718	6939#									
TXRREP	024230	G	6981#	8293	8415	8446	9079	9127	9242	9314	9340	9471	9503		
TXRXLB	005142	G	1594#	2496	2972	3399	3460	3815	3971	5004	5021	5044	6853*	7128	
TXRXLE	005202	G	1611#												
TXSCHR	025464	G	7638#	9028											
TXVECA	002172	G	1283#	7894*	8217	8299	8361	8452	9029	9144	9196	9352	9409	9514	
TSARGC=	000002		1017#	1018#	1019#	1020#	1021#	1022#	2333#	2337	2341#	2345	2347#	2351	2406#
			2411	2443#	2448	2452#	2457	2500#	2506	2519#	2525	2539#	2545	2548#	2553
			2595#	2600	2605#	2610	2644#	2650	2679#	2684	2686#	2691	2693#	2698	2731#
			2737	2739#	2745	2749#	2755	2758#	2763	2795#	2800	2804#	2809	2837#	2842
			2874#	2879	2887#	2893	2931#	2936	2945#	2949	2954#	2959	2979#	2988	3033#
			3038	3044#	3049	3054#	3058	3090#	3095	3117#	3121	3126#	3133	5081#	5085
			5279#	5285	7934#	7939	8012#	8017							
			9581#	9587#	9593#	9600#	9607#	9672#	9677#						
TSERRN=	022125		995#	5076#	8148#	8559#	8615#	9550#							
TSEXCP=	000000		9581#	9585	9587#	9591	9593#	9598	9600#	9605	9607#	9612	9677#	9682	
TSFLAG=	000050		7720#	7722	7991#	8032#	8034	8052#	8054						
TSGMAN=	000000		995#												
TSHILI=	177777		9581#	9584	9587#	9590	9593#	9597	9600#	9604	9607#	9611	9677#	9681	
TSLAST=	000001		995#	9720#											
TSLOLI=	000000		9581#	9583	9587#	9589	9593#	9596	9600#	9603	9607#	9610	9677#	9680	
TSLSYM=	010000		995#	1147	1168	2356	2415	2463	2560	2618	2654	2702	2767	2848	2901
			2999	3063	3140	7727	7952	7971	7999	8038	8060	8167	8310	8464	8624
			8779	8982	9153	9366	9522	9559	9617	9688					
TSLTNO=	000012		9723#												
TSNEST=	177777		995#	1006#	1135#	1146#	1159#	1167#	2326#	2355#	2403#	2414#	2438#	2462#	2492#
			2559#	2588#	2617#	2641#	2653#	2676#	2701#	2726#	2766#	2791#	2847#	2871#	2900#
			2926#	2998#	3028#	3062#	3085#	3139#	7717#	7726#	7737#	7744#	7760#	7951#	7966#
			7970#	7982#	7998#	8010#	8037#	8050#	8059#	8078#	8166#	8185#	8309#	8329#	8463#
			8478#	8623#	8641#	8778#	8796#	8981#	9000#	9152#	9169#	9365#	9379#	9521#	9535#
			9558#	9576#	9615#	9667#	9686#	9724#							
TSNSO =	000000		1006#	9724											
TSNS1 =	000005		1135#	1146	1159#	1167	2326#	2355	2403#	2414	2438#	2462	2492#	2559	2588#
			2617	2641#	2653	2676#	2701	2726#	2766	2791#	2847	2871#	2900	2926#	2998
			3028#	3062	3085#	3139	7717#	7726	7737#	7744	7760#	7951	7966#	7970	7982#
			7998	8010#	8037	8050#	8059	8078#	8166	8185#	8309	8329#	8463	8478#	8623
			8641#	8778	8796#	8981	9000#	9152	9169#	9365	9379#	9521	9535#	9558	9576#
			9615	9667#	9686										
TSPTNU=	000000		995#												
TSSAVL=	177777		995#												
TSSEGL=	177777		995#												
TSUBN=	000000		995#	8077#	8184#	8328#	8477#	8640#	8795#	8999#	9168#	9378#	9534#		

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 246
CROSS REFERENCE TABLE -- USER SYMBOLS

WTWLNC	025014	G	6807	7321#	8501	8534	8545	8576	8590	8676	8702	8831	8858				
WTWLPR	025044	G	6809	7361#	8509	8678	8704	8833	8860								
XSALWA=	000000		995#														
XSFALS=	000040		995#														
XSOFFS=	000400		995#														
XSTRUE=	000020		995#														
SPATCH	035016	G	9711#														
.	= 035072		1003#	1392#	1401#	1418#	1419#	1420#	1421#	1422#	1423#	1424#	1425#	1426#	1431#		
			1432#	1433#	1435#	1440#	1441#	1442#	1443#	1444#	1445#	1446#	1447#	1448#	1449#		
			1450#	1451#	1452#	1453#	1454#	1455#	1839#	2291#	7721	7992	8028#	8033	8053		
			8656	8663	8811	8818	8839	8849	8873	9016	9708#	9712#					

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 249
 CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- MACRO NAMES

ENDSW	1#	995#	1166														
ENDTST	1#	995#	8165	8308	8462	8622	8777	8980	9151	9364	9520	9557					
EQUALS	1#	995#	1199														
ERRDF	1#	995#	8146	8557	8613	9548											
ERRHRD	1#	995#															
ERROR	1#	995#	3738	5009	5027	5162	5217	5237	5407	5422	5567	5658	5716	5765	5836		
	5892	5940	6412	8750	8767	8964	8973										
ERRSF	1#	995#	5074														
ERRSOF	1#	995#															
ERRTBL	1#	995#	1638														
ESCAPE	1#	995#															
EXIT	1#	995#	7719	7990	8031	8051											
FEQUAL	1#	995#															
GETBYT	1#	995#															
GETPRI	1#	995#	4087	5545	6077	6704	6907										
GETWOR	1#	995#															
GMANIA	1#	995#															
GMANID	1#	995#															
GMANIL	1#	995#															
GPHARD	1#	995#	7880														
GPRMA	1#	995#	9580	9586													
GPRMD	1#	995#	9592	9599	9606	9676											
GPRML	1#	995#	9671														
HEADER	1#	995#	1015														
INLOOP	1#	995#															
IOSETU	1#	995#															
IOSTAR	1#	995#															
KT11	1#	995#															
LASTAD	1#	995#	9718														
MANUAL	1#	995#															
MEMORY	1#	995#															
MSBYTE	1#	995#	1016#	1022	1023	1024											
MSCHEC	1#	995#	7720#	7991#	8032#	8052#											
MSCNTO	1#	995#	9581#	9587#	9593#	9600#	9607#	9672#	9677#								
MSCOUN	1#	995#	2333#	2341#	2347#	2406#	2443#	2452#	2500#	2519#	2539#	2548#	2595#	2605#	2644#		
	2679#	2686#	2693#	2731#	2739#	2749#	2758#	2795#	2804#	2837#	2874#	2887#	2931#	2945#	2954#		
MSDATA	1#	995#	1016#	1025	1027	1029	1031	1033	1035	1037	1039	1041	1043	1045	1047		
	1049	1051	1053	1055#	1057	1059	1062	1065	1067	1069	1071	1073	1075	1077	1079		
	1081	1083	1085	1087	1089	1091	1093	1095	1097	1099	1836#	1845#					
MSDECR	1#	995#	1146#	1167#	2355#	2414#	2462#	2559#	2617#	2653#	2701#	2766#	2847#	2900#	2998#		
	3062#	3139#	7726#	7744#	7951#	7970#	7998#	8037#	8059#	8166#	8309#	8463#	8623#	8778#	8981#		
	9152#	9365#	9521#	9558#	9615#	9686#	9724#										
MSDEFA	1#	995#	9581#	9587#	9593#	9600#	9607#	9672#	9677#								
MSENDE	1#	995#	1146#	1167#	2355#	2414#	2462#	2559#	2617#	2653#	2701#	2766#	2847#	2900#	2998#		
	3062#	3139#	7726#	7951#	7970#	7998#	8037#	8059#	8166#	8309#	8463#	8623#	8778#	8981#	9152#		
	9365#	9521#	9558#	9615#	9686#	9724#											
MSERRI	1#	995#	5075#	8147#	8558#	8614#	9549#										
MSESCA	1#	995#															
MSESCS	1#	995#															
MSEXCP	1#	995#	9581#	9587#	9593#	9600#	9607#	9672#	9677#								
MSEXIT	1#	995#	7720#	7991#	7992	8032#	8052#										
MSEXSE	1#	995#	7720#	7991#	8032#	8052#											
MSEXTJ	1#	995#	7720#	7721	7991#	8032#	8033	8052#	8053								
MSGEN	1#	995#	1016#	1025#	1027#	1029#	1031#	1033#	1035#	1037#	1039#	1041#	1043#	1045#	1047#		
	1049#	1051#	1053#	1055#	1057#	1059#	1062#	1065#	1067#	1069#	1071#	1073#	1075#	1077#	1079#		

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 250
CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- MACRO NAMES

	1081#	1083#	1085#	1087#	1089#	1091#	1093#	1095#	1097#	1099#	1111#	1136#	1137#	1146#	1160#
	1161#	1167#	1639#	1836#	1845#	2326#	2355#	2403#	2414#	2438#	2462#	2492#	2559#	2588#	2617#
	2641#	2653#	2676#	2701#	2726#	2766#	2791#	2847#	2871#	2900#	2926#	2998#	3028#	3062#	3085#
	3139#	7717#	7726#	7737#	7760#	7951#	7966#	7970#	7982#	7998#	8010#	8037#	8050#	8059#	8077#
	8166#	8184#	8309#	8328#	8463#	8477#	8623#	8640#	8778#	8795#	8981#	8999#	9152#	9168#	9365#
	9777#	9521#	9534#	9558#	9577#	9616#	9668#	9687#	9722#						
MSGENB	1#	995#													
MSGETS	1#	995#	1146#	1167#	2355#	2414#	2462#	2559#	2617#	2653#	2701#	2766#	2847#	2900#	2998#
	3062#	3139#	7726#	7744#	7951#	7970#	7998#	8037#	8059#	8166#	8309#	8463#	8623#	8778#	8981#
	9152#	9365#	9521#	9558#	9615#	9686#	9724#								
MSGETT	1#	995#	7720#	7991#	8032#	8052#									
MSGNGB	1#	995#	1006#	1016#	1025#	1027#	1029#	1031#	1033#	1035#	1037#	1039#	1041#	1043#	1045#
	1047#	1049#	1051#	1053#	1055#	1057#	1059#	1062#	1065#	1067#	1069#	1071#	1073#	1075#	1077#
	1079#	1081#	1083#	1085#	1087#	1089#	1091#	1093#	1095#	1097#	1099#	1110#	1111	1135#	1136
	1137	1159#	1160	1161	1639#	1836#	1845#	2326#	2403#	2438#	2492#	2588#	2641#	2676#	2726#
	2791#	2871#	2926#	3028#	3085#	7717#	7737#	7760#	7966#	7982#	8010#	8050#	9576#	9577	9667#
	9668	9719#	9722												
MSGNIN	1#	995#	1016#	1017	1018	1019	1020	1021	1022#	1023#	1024#	1025#	1026	1027#	1028
	1029#	1030	1031#	1032	1033#	1034	1035#	1036	1037#	1038	1039#	1040	1041#	1042	1043#
	1044	1045#	1046	1047#	1048	1049#	1050	1051#	1052	1053#	1054	1055#	1056	1057#	10
	1059#	1060	1061	1062#	1063	1064#	1065#	1066	1067#	1068	1069#	1070	1071#	1072	1075#
	1074	1075#	1076	1077#	1078	1079#	1080	1081#	1082	1083#	1084	1085#	1086	1087#	1088
	1089#	1090	1091#	1092	1093#	1094	1095#	1096	1097#	1098	1099#	1100	1110#	1112#	1113#
	1114#	1115#	1116#	1117#	1118#	1119#	1120#	1121#	1135#	1159#	1836#	1837	1839	1845#	1846
	1850	2333#	2334#	2335	2336#	2337	2341#	2342#	2343	2344#	2345	2347#	2348#	2349	2350#
	2351	2356#	2406#	2407#	2408#	2409	2410#	2411	2415#	2443#	2444#	2445#	2446	2447#	2448
	2452#	2453#	2454#	2455	2456#	2457	2463#	2500#	2501#	2502#	2503#	2504	2505#	2506	2519#
	2520#	2521#	2522#	2523	2524#	2525	2539#	2540#	2541#	2542#	2543	2544#	2545	2548#	2549#
	2550#	2551	2552#	2553	2560#	2595#	2596#	2597#	2598	2599#	2600	2605#	2606#	2607#	2608
	2609#	2610	2618#	2644#	2645#	2646#	2647#	2648	2649#	2650	2654#	2679#	2680#	2681#	2682
	2683#	2684	2686#	2687#	2688#	2689	2690#	2691	2693#	2694#	2695#	2696	2697#	2698	2702#
	2731#	2732#	2733#	2734#	2735	2736#	2737	2739#	2740#	2741#	2742#	2743	2744#	2745	2749#
	2750#	2751#	2752#	2753	2754#	2755	2758#	2759#	2760#	2761	2762#	2763	2767#	2795#	2796#
	2797#	2798	2799#	2800	2804#	2805#	2806#	2807	2808#	2809	2837#	2838#	2839#	2840	2841#
	2842	2848#	2874#	2875#	2876#	2877	2878#	2879	2887#	2888#	2889#	2890#	2891	2892#	2893
	2901#	2931#	2932#	2933#	2934	2935#	2936	2945#	2946#	2947	2948#	2949	2954#	2955#	2956#
	2957	2958#	2959	2979#	2980#	2981#	2982#	2983#	2984#	2985#	2986	2987#	2988	2999#	3033#
	3034#	3035#	3036	3037#	3038	3044#	3045#	3046#	3047	3048#	3049	3054#	3055#	3056	3057#
	3058	3063#	3090#	3091#	3092#	3093	3094#	3095	3117#	3118#	3119	3120#	3121	3126#	3127#
	3128#	3129#	3130#	3131	3132#	3133	3140#	3739#	4088#	4089#	4091#	4092#	4102#	4103#	5010#
	5028#	5075#	5076#	5077#	5078#	5081#	5082#	5083	5084#	5085	5087#	5163#	5218#	5238#	5279#
	5280#	5281#	5282#	5283	5284#	5285	5408#	5423#	5546#	5547#	5549#	5550#	5551#	5555#	5568#
	5659#	5717#	5766#	5837#	5893#	5941#	6078#	6079#	6081#	6082#	6086#	6087#	6413#	6705#	6706#
	6708#	6709#	6713#	6714#	6908#	6909#	6911#	6912#	6944#	6945#	7412#	7720#	7721#	7727#	7763#
	7764#	7766#	7769#	7770#	7772#	7775#	7776#	7778#	7781#	7782#	7784#	7788#	7793#	7794#	7795#
	7806#	7807#	7808#	7809#	7810#	7811	7816#	7817#	7861#	7881#	7882#	7883#	7885#	7934#	7935#
	7936#	7937	7938#	7939	7947#	7948#	7952#	7971#	7988#	7991#	7992#	7999#	8012#	8013#	8014#
	8015	8016#	8017	8032#	8033#	8038#	8052#	8053#	8060#	8147#	8148#	8149#	8150#	8153#	8154#
	8157#	8167#	8191#	8192#	8212#	8213#	8215#	8216#	8217#	8218#	8219#	8220	8222#	8223#	8296#
	8297#	8299#	8300#	8310#	8335#	8336#	8356#	8357#	8359#	8360#	8361#	8362#	8363#	8364	8366#
	8367#	8449#	8450#	8452#	8453#	8464#	8558#	8559#	8560#	8561#	8614#	8615#	8616#	8617#	8624#
	8642#	8643#	8751#	8768#	8779#	8797#	8798#	8965#	8974#	8982#	9001#	9002#	9024#	9025#	9027#
	9028#	9029#	9030#	9031#	9032	9034#	9035#	9036#	9037#	9038#	9039	9041#	9042#	9141#	9142#
	9144#	9145#	9147#	9148#	9153#	9170#	9171#	9191#	9192#	9194#	9195#	9196#	9197#	9198#	9199
	9201#	9202#	9203#	9204#	9205#	9206	9208#	9209#	9349#	9350#	9352#	9353#	9355#	9356#	9361#
	9362#	9366#	9383#	9384#	9404#	9405#	9407#	9408#	9409#	9410#	9411#	9412	9414#	9415#	9416#

CVDHCAO DHV-11 FUNC TST PART1
CVDHCA.P11 12-JUL-83 11:44

MACY11 30A(1052) 12-JUL-83 13:56 PAGE 251
CROSS REFERENCE TABLE -- MACRO NAMES

	9417#	9418#	9419	9421#	9422#	9511#	9512#	9514#	9515#	9522#	9549#	9550#	9551#	9552#	9559#
	9576#	9581#	9582	9583	9584	9587#	9588	9589	9590	9593#	9594	9595	9596	9597	9600#
	9601	9602	9603	9604	9607#	9608	9609	9610	9611	9615#	9667#	9672#	9673	9674	9677#
	9678	9679	9680	9681	9686#	9719#	9720#	9721#							
MSGNLS	1#	995#													
MSGNSU	1#	995#													
MSGNTA	1#	995#	1146#	1167#	2355#	2414#	2462#	2559#	2617#	2653#	2701#	2766#	2847#	2900#	2998#
	3062#	3139#	7726#	7951#	7970#	7998#	8037#	8059#	8166#	8309#	8463#	8623#	8778#	8981#	9152#
	9365#	9521#	9558#	9615#	9616	9686#	9687								
MSGNTE	1#	995#	8077#	8184#	8328#	8477#	8640#	8795#	8999#	9168#	9378#	9534#			
MSHAPT	1#	995#	1016#												
MSHNAP	1#	995#	1016#	1055											
MSINCR	1#	995#	1006#	1135#	1159#	2326#	2336#	2344#	2350#	2356#	2403#	2410#	2415#	2438#	2447#
	2456#	2463#	2492#	2505#	2524#	2544#	2552#	2560#	2588#	2599#	2609#	2618#	2641#	2649#	2654#
	2676#	2683#	2690#	2697#	2702#	2726#	2736#	2744#	2754#	2762#	2767#	2791#	2799#	2808#	2841#
	2848#	2871#	2878#	2892#	2901#	2926#	2935#	2948#	2958#	2987#	2999#	3028#	3037#	3048#	3057#
	3063#	3085#	3094#	3120#	3132#	3140#	3739#	4088#	4092#	4103#	5010#	5028#	5075#	5084#	5087#
	5163#	5218#	5238#	5284#	5408#	5423#	5546#	5550#	5555#	5568#	5659#	5717#	5766#	5837#	5893#
	5941#	6078#	6082#	6087#	6413#	6705#	6709#	6714#	6908#	6912#	6945#	7412#	7717#	7727#	7737#
	7760#	7764#	7770#	7776#	7782#	7788#	7794#	7810#	7817#	7861#	7882#	7938#	7948#	7952#	7966#
	7971#	7982#	7988#	7991#	7999#	8010#	8016#	8038#	8050#	8060#	8077#	8078#	8147#	8154#	8157#
	8167#	8184#	8185#	8192#	8213#	8219#	8223#	8297#	8300#	8310#	8328#	8329#	8336#	8357#	8363#
	8367#	8450#	8453#	8464#	8477#	8478#	8558#	8614#	8624#	8640#	8641#	8643#	8751#	8768#	8779#
	8795#	8796#	8798#	8965#	8974#	8982#	8999#	9000#	9002#	9025#	9031#	9038#	9042#	9142#	9145#
	9148#	9153#	9168#	9169#	9171#	9192#	9198#	9205#	9209#	9350#	9353#	9356#	9362#	9366#	9378#
	9379#	9384#	9405#	9411#	9418#	9422#	9512#	9515#	9522#	9534#	9535#	9549#	9559#	9576#	9667#
MSIOSE	1#	995#													
MSLDRO	1#	995#	4091#	4102#	5549#	5554#	6081#	6086#	6708#	6713#	6911#	6944#	7763#	7769#	7775#
	7781#	7793#	7816#	7881#	7947#	8153#	8191#	8212#	8222#	8296#	8299#	8335#	8356#	8366#	8449#
	8452#	8642#	8797#	9001#	9024#	9041#	9141#	9144#	9147#	9170#	9191#	9208#	9349#	9352#	9355#
	9361#	9383#	9404#	9421#	9511#	9514#									
MSMASK	1#	995#													
MSMCHI	1#	995#													
MSMCLO	1#	995#													
MSMSK1	1#	995#													
MSPOP	1#	995#	1146#	1167#	2355#	2414#	2462#	2559#	2617#	2653#	2701#	2766#	2847#	2900#	2998#
	3062#	3139#	7726#	7744#	7951#	7970#	7998#	8037#	8059#	8166#	8309#	8463#	8623#	8778#	8981#
	9152#	9365#	9521#	9558#	9615#	9686#	9724#								
MSPRIN	1#	995#	2333#	2341#	2347#	2406#	2443#	2452#	2500#	2519#	2539#	2548#	2595#	2605#	2644#
	2679#	2686#	2693#	2731#	2739#	2749#	2758#	2795#	2804#	2837#	2874#	2887#	2931#	2945#	2954#
	2979#	3033#	3044#	3054#	3090#	3117#	3126#	5081#	5279#	7934#	8012#				
MSPUSH	1#	995#	1006#	1135#	1159#	2326#	2403#	2438#	2492#	2588#	2641#	2676#	2726#	2791#	2871#
	2926#	3028#	3085#	7717#	7737#	7760#	7966#	7982#	8010#	8050#	8077#	8078	8184#	8185	8328#
	8329	8477#	8478	8640#	8641	8795#	8796	8999#	9000	9168#	9169	9378#	9379	9534#	9535
	9576#	9667#													
MSPUT	1#	995#	2333#	2341#	2347#	2406#	2443#	2452#	2500#	2519#	2539#	2548#	2595#	2605#	2644#
	2679#	2686#	2693#	2731#	2739#	2749#	2753#	2795#	2804#	2837#	2874#	2887#	2931#	2945#	2954#
	2979#	3033#	3044#	3054#	3090#	3117#	3126#	5081#	5279#	7806#	7934#	8012#	8215#	8359#	9027#
	9034#	9194#	9201#	9407#	9414#										
MSPUT1	1#	995#	2333#	2334	2341#	2342	2347#	2348	2406#	2407	2408	2443#	2444	2445	2452#
	2453	2454	2500#	2501	2502	2503	2519#	2520	2521	2522	2539#	2540	2541	2542	2548#
	2549	2550	2595#	2596	2597	2605#	2606	2607	2644#	2645	2646	2647	2679#	2680	2681
	2686#	2687	2688	2693#	2694	2695	2731#	2732	2733	2734	2739#	2740	2741	2742	2749#
	2750	2751	2752	2758#	2759	2760	2795#	2796	2797	2804#	2805	2806	2837#	2838	2839
	2874#	2875	2876	2887#	2888	2889	2890	2931#	2932	2933	2945#	2946	2954#	2955	2956
	2979#	2980	2981	2982	2983	2984	2985	3033#	3034	3035	3044#	3045	3046	3054#	3055

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 252
CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- MACRO NAMES

	3090#	3091	3092	3117#	3118	3126#	3127	3128	3129	3130	5081#	5082	5279#	5280	5281
	5282	7806#	7807	7808	7809	7934#	7935	7936	8012#	8013	8014	8215#	8216	8217	8218
	8359#	8360	8361	8362	9027#	9028	9029	9030	9034#	9035	9036	9037	9194#	9195	9196
	9197	9201#	9202	9203	9204	9407#	9408	9409	9410	9414#	9415	9416	9417		
MSRADI	1#	995#	9581#	9587#	9593#	9600#	9607#	9672#	9677#						
MSRBRO	1#	995#													
MSRNRO	1#	995#	4088#	4089	5546#	5547	6078#	6079	6705#	6706	6908#	6909	7793#	7795	7881#
	7883														
MSSETS	1#	995#	1006#	1135#	1159#	2326#	2403#	2438#	2492#	2588#	2641#	2676#	2726#	2791#	2871#
	2926#	3028#	3085#	7717#	7737#	7760#	7966#	7982#	8010#	8050#	8078#	8185#	8329#	8478#	8641#
	8796#	9000#	9169#	9379#	9535#	9576#	9667#								
MSSTAR	1#	995#													
MS SVC	1#	995#	2333#	2336	2341#	2344	2347#	2350	2355#	2356	2406#	2410	2414#	2415	2443#
	2447	2452#	2456	2462#	2463	2500#	2505	2519#	2524	2539#	2544	2548#	2552	2559#	2560
	2595#	2599	2605#	2609	2617#	2618	2644#	2649	2653#	2654	2679#	2683	2686#	2690	2693#
	2697	2701#	2702	2731#	2736	2739#	2744	2749#	2754	2758#	2762	2766#	2767	2795#	2799
	2804#	2808	2837#	2841	2847#	2848	2874#	2878	2887#	2892	2900#	2901	2931#	2935	2945#
	2948	2954#	2958	2979#	2987	2998#	2999	3033#	3037	3044#	3048	3054#	3057	3062#	3063
	3090#	3094	3117#	3120	3126#	3132	3139#	3140	3739#	4088#	4091#	4092	4102#	4103	5010#
	5028#	5075	5081#	5084	5087#	5163#	5218#	5238#	5279#	5284	5408#	5423#	5546#	5549#	5550
	5554#	5555	5568#	5659#	5717#	5766#	5837#	5893#	5941#	6078#	6081#	6082	6086#	6087	6413#
	6705#	6708#	6709	6713#	6714	6908#	6911#	6912	6944#	6945	7412#	7720#	7726#	7727	7763#
	7764	7769#	7770	7775#	7776	7781#	7782	7788#	7793#	7794	7806#	7810	7816#	7817	7861#
	7881#	7882	7934#	7938	7947#	7948	7951#	7952	7970#	7971	7988#	7991#	7998#	7999	8012#
	8016	8032#	8037#	8038	8052#	8059#	8060	8147	8153#	8154	8157#	8166#	8167	8191#	8192
	8212#	8213	8215#	8219	8222#	8223	8296#	8297	8299#	8300	8309#	8310	8335#	8336	8356#
	8357	8359#	8363	8366#	8367	8449#	8450	8452#	8453	8463#	8464	8558	8614	8623#	8624
	8642#	8643	8751#	8768#	8778#	8779	8797#	8798	8965#	8974#	8981#	8982	9001#	9002	9024#
	9025	9027#	9031	9034#	9038	9041#	9042	9141#	9142	9144#	9145	9147#	9148	9152#	9153
	9170#	9171	9191#	9192	9194#	9198	9201#	9205	9208#	9209	9349#	9350	9352#	9353	9355#
	9356	9361#	9362	9365#	9366	9383#	9384	9404#	9405	9407#	9411	9414#	9418	9421#	9422
	9511#	9512	9514#	9515	9521#	9522	9549	9558#	9559						
MSLAB	1#	995#	2336#	2344#	2350#	2356#	2410#	2415#	2447#	2456#	2463#	2505#	2524#	2544#	2552#
	2560#	2599#	2609#	2618#	2649#	2654#	2683#	2690#	2697#	2702#	2736#	2744#	2754#	2762#	2767#
	2799#	2808#	2841#	2848#	2878#	2892#	2901#	2935#	2948#	2958#	2987#	2999#	3037#	3048#	3057#
	3063#	3094#	3120#	3132#	3140#	3739#	4088#	4092#	4103#	5010#	5028#	5075#	5084#	5087#	5163#
	5218#	5238#	5284#	5408#	5423#	5546#	5550#	5555#	5568#	5659#	5717#	5766#	5837#	5893#	5941#
	6078#	6082#	6087#	6413#	6705#	6709#	6714#	6908#	6912#	6945#	7412#	7727#	7764#	7770#	7776#
	7782#	7788#	7794#	7810#	7817#	7861#	7882#	7938#	7948#	7952#	7971#	7988#	7991#	7999#	8016#
	8038#	8060#	8147#	8154#	8157#	8167#	8192#	8213#	8219#	8223#	8297#	8300#	8310#	8336#	8357#
	8363#	8367#	8450#	8453#	8464#	8558#	8614#	8624#	8643#	8751#	8768#	8779#	8798#	8965#	8974#
	8982#	9002#	9025#	9031#	9038#	9042#	9142#	9145#	9148#	9153#	9171#	9192#	9198#	9205#	9209#
	9350#	9353#	9356#	9362#	9366#	9384#	9405#	9411#	9418#	9422#	9512#	9515#	9522#	9549#	9559#
MS STL	1#	995#	2336#	2344#	2350#	2356#	2410#	2415#	2447#	2456#	2463#	2505#	2524#	2544#	2552#
	2560#	2599#	2609#	2618#	2649#	2654#	2683#	2690#	2697#	2702#	2736#	2744#	2754#	2762#	2767#
	2799#	2808#	2841#	2848#	2878#	2892#	2901#	2935#	2948#	2958#	2987#	2999#	3037#	3048#	3057#
	3063#	3094#	3120#	3132#	3140#	3739#	4088#	4092#	4103#	5010#	5028#	5075#	5084#	5087#	5163#
	5218#	5238#	5284#	5408#	5423#	5546#	5550#	5555#	5568#	5659#	5717#	5766#	5837#	5893#	5941#
	6078#	6082#	6087#	6413#	6705#	6709#	6714#	6908#	6912#	6945#	7412#	7727#	7764#	7770#	7776#
	7782#	7788#	7794#	7810#	7817#	7861#	7882#	7938#	7948#	7952#	7971#	7988#	7991#	7999#	8016#
	8038#	8060#	8147#	8154#	8157#	8167#	8192#	8213#	8219#	8223#	8297#	8300#	8310#	8336#	8357#
	8363#	8367#	8450#	8453#	8464#	8558#	8614#	8624#	8643#	8751#	8768#	8779#	8798#	8965#	8974#
	8982#	9002#	9025#	9031#	9038#	9042#	9142#	9145#	9148#	9153#	9171#	9192#	9198#	9205#	9209#
	9350#	9353#	9356#	9362#	9366#	9384#	9405#	9411#	9418#	9422#	9512#	9515#	9522#	9549#	9559#
MSWORD	1#	995#	1055#	1064	1110#	1112	1113	1114	1115	1116	1117	1118	1119	1120	1121
	5075#	5076	5077	5078	7720#	7991#	8032#	8052#	8147#	8148	8149	8150	8558#	8559	8560

CVDHCAO DHV-11 FUNC TST PART1 MACY11 30A(1052) 12-JUL-83 13:56 PAGE 253
 CVDHCA.P11 12-JUL-83 11:44 CROSS REFERENCE TABLE -- MACRO NAMES

	8561	8614#	8615	8616	8617	9549#	9550	9551	9552	9581#	9587#	9593#	9600#	9607#	9672#
MSXFER	9677#	9720	9721												
OPEN	1#	995#													
PASS	1758#	2352	2459	2556	2614	2995	3059	3135	3206	3314	3351	3423	3496	3534	3674
	3764	3838	3875	3920	3946	3984	4016	4106	4147	4278	4327	4397	4464	4507	4539
	4598	4676	4708	4740	4823	4876	4921	5047	5089	5166	5243	5286	5336	5430	5466
	5622	5661	5724	5768	5847	5896	5944	5997	6052	6148	6191	6330	6425	6489	6544
	6600	6682	6880	6916	6946	6995	7091	7155	7247	7293	7335	7375	7601	7704	
POINTE	1#	995#	1012												
PRINTB	1#	995#	2332	2340	2405	2442	2451	2499	2594	2604	2643	2678	2730	2794	2873
	2930	3032	3043	3053	3089										
PRINTF	1#	995#	5080	7933	8011										
PRINTS	1#	995#													
PRINTX	1#	995#	2346	2518	2538	2547	2685	2692	2738	2748	2757	2803	2836	2886	2944
	2953	2978	3116	3125	5278										
READBU	1#	995#													
REDEF	1#	995#	7762	7768	7774	7780									
RFLAGS	1#	995#													
SAVE	1717#	2327	2439	2493	2589	2927	3029	3086	3176	3237	3340	3386	3456	3522	3572
	3712	3796	3867	3906	3940	3969	4007	4052	4128	4195	4310	4363	4435	4490	4529
	4575	4632	4703	4728	4782	4867	4901	4966	5071	5139	5196	5269	5308	5368	5456
	5502	5646	5686	5753	5796	5876	5923	5970	6025	6136	6171	6243	6408	6448	6516
	6572	6637	6797	6905	6939	6981	7021	7126	7193	7278	7321	7361	7540	7638	
SETPRI	1#	995#	4090	4101	5548	5553	6080	6085	6707	6712	6910	6943	7815	7946	8190
	8211	8221	8295	8334	8355	8365	8448	8641	8796	9000	9023	9040	9140	9169	9190
	9207	9348	9360	9382	9403	9420	9510								
SETVEC	1#	995#	7805	8214	8358	9026	9033	9193	9200	9406	9413				
SLASH	1#	995#													
STARS	1#	995#													
SVC	1#	993#	994												
XFER	1#	995#	7720#	7991#	8032#	8052#									
XFERF	1#	995#													
XFERT	1#	995#													

. ABS. 035072 000

ERRORS DETECTED: 0

CVDHCA.BIC,CVDHCA.LST/CRF/SOL/NL:TOC=SVC34R.MLB,CVDHCA.P11
 RUN-TIME: 25 36 2 SECONDS
 RUN-TIME RATIO: 96/64=1.4
 CORE USED: 16K (31 PAGES)