

FEPOM

FROM CHKSM CHECK DIAG  
CZFPDRO

COPYRIGHT (c) 1982-84  
AH-T865A-MC  
FICHE 1 OF 1

APR 1984  
digital  
Made In USA

*[Faint, illegible text and graphics, likely bleed-through from the reverse side of the page.]*

10

.REM 6

IDENTIFICATION  
-----

PRODUCT CODE: AC-T864A-MC  
PRODUCT NAME: CZFPDAO PROM CHKSM CHECK DIAG  
PRODUCT DATE: MAY 1982  
MAINTAINER: CSS GNG DIAGNOSTIC ENGINEERING  
AUTHOR: DALE PROCTOR

COPYRIGHT (C) 1982,1984  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS 01754

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY TRANSFERED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC  
LSI

PDP  
DECUS  
VAX

UNIBUS  
DECTAPE

MASSBUS  
Q-BUS

## REVISION HISTORY

A.0 DALE PROCTOR 22-DEC-1983  
CHANGED THE NAME TO CZFPD-A0 AND RELEASED TO  
SDC. ALSO UPDATED ALL THE DOCUMENTATION, AND  
REMOVED THE HELP FILE QUESTIONS FROM THE  
INITIALIZATION SECTION.

5-NOV-1983 DALE PROCTOR V B.O  
ADDED CHANGES TO ALLOW CHECKING THE 11/23B PROCESSOR  
PROM. THE PRIMARY CHANGE IS THE PAGE ADDRESS TABLE  
FOR THE ADDRESSING THE PROM. THE CODE NOW RESTRICTS  
PROM CHECKING TO 512 WORDS.

MAY 1982 DALE PROCTOR INITIAL RELEASE

## TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
2.1	COMMANDS
2.2	SWITCHES
2.3	FLAGS
2.4	HARDWARE QUESTIONS
2.5	SOFTWARE QUESTIONS
2.6	EXTENDED P-TABLE DIALOGUE
2.7	QUICK STARTUP PROCEDURE
3.0	ERROR INFORMATION
4.0	PERFORMANCE AND PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	TEST SUMMARIES

## 1.0 GENERAL INFORMATION

### 1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC IS TO CHECK A CHECK SUM IN MEMORY, USUALLY A ROM OR PROM. IT READS THE MEMORY AND CALCULATES A CHECK SUM VALUE. IT THEN CHECKS THE LAST ADDRESS SPECIFIED AND COMPARES IT WITH THE CALCULATED CHECK SUM. ERROR CHECKING IS ALSO DONE FOR NON-EXISTANT MEMORY.

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 SECTION 2 OF THIS DOCUMENT.

### 1.2 SYSTEM REQUIREMENTS

THIS DIAGNOSTIC REQUIRES A MINIMUM OF 16K MEMORY, A CONSOLE TERMINAL, AND SOME LOAD MEDIUM.

### 1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USER'S MANUAL - CHQUS.

### 1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE CPU IS ASSUMED TO HAVE BEEN TESTED AND FOUND WORKING.

### 1.5 ASSUMPTIONS

THE ONLY ASSUMPTION IS THAT A CHECK SUM VALUE IS PROVIDED IN THE LAST ADDRESS OF THE MEMORY UNDER TEST.

## 2.0 OPERATING INSTRUCTIONS

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

### 2.1 COMMANDS

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

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ↑C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE

DROP	CONSIDERED TO BE ACTIVE AT START TIME
PRINT	DEACTIVATE A UNIT
	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED
	BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

## 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.
/PASS:DDDD	EXECUTE DDDDD PASSES (DDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/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
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	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)

\*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

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 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?"

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

THERE ARE TWO HARDWARE QUESTIONS.

- 1) STARTING ADDRESS TO CHECK
- 2) NUMBER OF WORDS OF MEMORY TO CHECK.

THERE ARE NO DEFAULTS

## 2.5 SOFTWARE QUESTIONS

THERE ARE NO SOFTWARE QUESTIONS.

## 2.6 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 (0) ? 160000<CR>  
SUB-DEVICE # (0) ? 0<CR>  
Q-FACTOR (0) 0 ? 1<CR>
```

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

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

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



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

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

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

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

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

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

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

◆ UNITS (0) ? 8<CR>

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

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

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

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE

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

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

# UNITS (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.

## 2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

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

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

## 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 (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

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

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

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBE" FLAGS ARE SET (SECTION 2.3). 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", "IBE" OR "IXE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

### 3.2 SPECIFIC ERROR MESSAGES

THERE ARE TWO ERROR MESSAGES PROVIDED. ONE IS THE RESULT OF THE CHECK SUM IF FOUND IN ERROR, AND THE SECOND IS FOR ANY NON-EXISTANT MEMORY ERRORS.

### 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. SECTION 2.2 DESCRIBES SWITCHES.

### 5.0 DEVICE INFORMATION TABLES

### 6.0 TEST SUMMARIES

```

11          .TITLE PROGRAM HEADER AND TABLES
12          .SBTTL PROGRAM HEADER
38
40 000000          .ENABL ABS,AMA
41          002000          .          =          2000
43
44 002000          BGNMOD
45
46          ;++
47          ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
48          ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
49          ;--
50
51 002000          POINTER NONE
52
69
70 002000          HEADER CZFPD,A,0,1,0
          002000          L$NAME::          ;DIAGNOSTIC NAME
          002000          103          .ASCII /C/
          002001          132          .ASCII /Z/
          002002          106          .ASCII /F/
          002003          120          .ASCII /P/
          002004          104          .ASCII /D/
          002005          000          .BYTE 0
          002006          000          .BYTE 0
          002007          000          .BYTE 0
          002010          L$REV::          ;REVISION LEVEL
          002010          101          .ASCII /A/
          002011          L$DEPO::          ;0
          002011          060          .ASCII /O/
          002012          L$UNIT::          ;NUMBER OF UNITS
          002012          000000          .WORD 0
          002014          L$TIML::          ;LONGEST TEST TIME
          002014          000001          .WORD 1
          002016          L$HPCP::          ;POINTER TO H.W. QUES.
          002016          003456          .WORD L$HARD
          002020          L$SPCP::          ;POINTER TO S.W. QUES.
          002020          000000          .WORD 0
          002022          L$HPTP::          ;PTR. TO DEF. H.W. PTABLE
          002022          002130          .WORD L$HW
          002024          L$SPTP::          ;PTR. TO S.W. PTABLE
          002024          000000          .WORD 0
          002026          L$LADP::          ;DIAG. END ADDRESS
          002026          003644          .WORD L$LAST
          002030          L$STA::          ;RESERVED FOR APT STATS
          002030          000000          .WORD 0
          002032          L$CO::          .WORD 0
          002032          000000          .WORD 0
          002034          L$DTYP::          ;DIAGNOSTIC TYPE
          002034          000000          .WORD 0
          002036          L$APT::          ;APT EXPANSION
          002036          000000          .WORD 0
          002040          L$DTP::          ;PTR. TO DISPATCH TABLE
          002040          002124          .WORD L$DISPATCH
          002042          L$PRIO::          ;DIAGNOSTIC RUN PRIORITY
          002042          000000          .WORD 0
          002044          L$ENVI::          ;FLAGS DESCRIBE HOW IT WAS SETUP

```

PROGRAM HEADER

002044 000000  
002046 000000  
002050 003  
002051 003  
002052 000000  
002054 000000  
002056 000000  
002060 002162  
002062 000000  
002064 000000  
002066 000000  
002070 000000  
002072 000000  
002074 000000  
002076 002170  
002100 104035  
002102 000000  
002104 003116  
002106 003246  
002110 003242  
002112 003110  
002114 000000  
002116 000000  
002120 000000

.WORD 0 ;EXPANSION WORD  
L\$EXP1:: .WORD 0 ;SVC REV AND EDIT #  
L\$MREV:: .BYTE C\$REVISION  
          .BYTE C\$EDIT  
L\$EF:: .WORD 0 ;DIAG. EVENT FLAGS  
          .WORD 0  
L\$SPC:: .WORD 0  
L\$DEVP:: .WORD 0 ; POINTER TO DEVICE TYPE LIST  
L\$REPP:: .WORD L\$DVTYP ;PTR. TO REPORT CODE  
L\$EXP4:: .WORD 0  
L\$EXP5:: .WORD 0  
L\$AUT:: .WORD 0 ;PTR. TO ADD UNIT CODE  
L\$DUT:: .WORD 0 ;PTR. TO DROP UNIT CODE  
L\$LUN:: .WORD 0 ;LUN FOR EXERCISERS TO FILL  
L\$DESP:: .WORD 0 ;POINTER TO DIAG. DESCRIPTION  
L\$LOAD:: .WORD L\$DESC ;GENERATE SPECIAL AUTOLOAD EMT  
          EMT E\$LOAD  
L\$ETP:: .WORD 0 ;POINTER TO ERR\_TBL  
L\$ICP:: .WORD 0 ;PTR. TO INIT CODE  
L\$CCP:: .WORD L\$INIT ;PTR. TO CLEAN-UP CODE  
L\$ACP:: .WORD L\$CLEAN ;PTR. TO AUTO CODE  
L\$PRT:: .WORD L\$AUTO ;PTR. TO PROTECT TABLE  
L\$TEST:: .WORD L\$PROT ;TEST NUMBER  
L\$DLY:: .WORD 0 ;DELAY COUNT  
L\$HIME:: .WORD 0 ;PTR. TO HIGH MEM

DISPATCH TABLE

83  
 84  
 85  
 86  
 87  
 88  
 89  
 90 002122  
 002122 000001  
 002124  
 002124 003266  
 91

```
.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 1
.WORD 1
L#DISPATCH::
.WORD T1
```

DEFAULT HARDWARE P-TABLE

99  
100  
101  
102  
103  
104  
105  
106  
107

108 002126  
002126 000002  
002130  
002130

109  
110 002130 173000  
111 002132 000400

121  
122 002134  
002134

.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 173000 ;DEFAULT PROM STARTING ADDRESS  
.WORD 256. ;DEFAULT # OF WORDS IN ROM

ENDHW  
L10000:

SOFTWARE P-TABLE

124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
142  
143  
144  
145

002134  
002134 000000  
002136  
002136

002136  
002136

002136

```

.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::

          ENDSW
L10001:

          ENDMOD

```



SOFTWARE P-TABLE

12  
13  
41  
51  
52 002136  
53  
54  
55  
56  
57  
58  
59 002136

.TITLE GLOBAL AREAS  
.SBTTL GLOBAL EQUATES SECTION

BGNMOD

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

EQUALS

;  
; BIT DIFINITIONS

100000	BIT15==	100000
040000	BIT14==	40000
020000	BIT13==	20000
010000	BIT12==	10000
004000	BIT11==	4000
002000	BIT10==	2000
001000	BIT09==	1000
000400	BIT08==	400
000200	BIT07==	200
000100	BIT06==	100
000040	BIT05==	40
000020	BIT04==	20
000010	BIT03==	10
000004	BIT02==	4
000002	BIT01==	2
000001	BIT00==	1

001000	BIT9==	BIT09
000400	BIT8==	BIT08
000200	BIT7==	BIT07
000100	BIT6==	BIT06
000040	BIT5==	BIT05
000020	BIT4==	BIT04
000010	BIT3==	BIT03
000004	BIT2==	BIT02
000002	BIT1==	BIT01
000001	BIT0==	BIT00

;  
; EVENT FLAG DEFINITIONS  
; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

000040	EF.START==	32.	; START COMMAND WAS ISSUED
000037	EF.RESTART==	31.	; RESTART COMMAND WAS ISSUED
000036	EF.CONTINUE==	30.	; CONTINUE COMMAND WAS ISSUED
000035	EF.NEW==	29.	; A NEW PASS HAS BEEN STARTED
000034	EF.PWR==	28.	; A POWER-FAIL/POWER-UP OCCURRED

;  
; PRIORITY LEVEL DEFINITIONS

000340	PRI07==	340
000300	PRI06==	300

GLOBAL EQUATES SECTION

	000240	PRI05== 240
	000200	PRI04== 200
	000140	PRI03== 140
	000100	PRI02== 100
	000040	PRI01== 40
	000000	PRI00== 0
		;
		;OPERATOR FLAG BITS
		;
	000004	EVL== 4
	000010	LOT== 10
	000020	ADR== 20
	000040	IDU== 40
	000100	ISR== 100
	000200	UAM== 200
	000400	BOE== 400
	001000	PNT== 1000
	002000	PRI== 2000
	004000	IXE== 4000
	010000	IBE== 10000
	020000	IER== 20000
	040000	LOE== 40000
	100000	MOE== 100000
		;
		;MISC EQUATES
		;
60		NO == 0
61		YES == 1
62		;
63	000000	;
64	000001	;
65		;
66		; PDP-11/23+ PROCESSOR PAGE CONTROL REGISTER ADDRESS
67		;
68	177520	CS#PCR == 177520
69		;



## GLOBAL TEXT SECTION

```

91          .SBTTL GLOBAL TEXT SECTION
92
93          ;**
94          ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
95          ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
96          ; MORE THAN ONE TEST.
97          ;--
98
99          ;
100         ; NAMES OF DEVICES SUPPORTED BY PROGRAM
101         ;
102         002162          DEVTYP <PROM>
103         002162          L$DVTYP::
104         002162          120    122    117    .ASCIZ /PROM/
105         002165          115    000
106         .EVEN
107
108         ; TEST DESCRIPTION
109         ;
110         002170          DESCRIPT    <ROM OR PROM CHECKSUM CHECK>
111         002170          L$DESC::
112         002170          122    117    115    .ASCIZ /ROM OR PROM CHECKSUM CHECK/
113         002173          040    117    122
114         002176          040    120    122
115         002201          117    115    040
116         002204          103    110    105
117         002207          103    113    123
118         002212          125    115    040
119         002215          103    110    105
120         002220          103    113    000
121         .EVEN
122         .EVEN
123
124         ;
125         ; FORMAT STATEMENTS USED IN PRINT CALLS
126         ;
127         002224          045    116    045    FOR1:: .ASCII /#N#ACALCULATED WAS #06/
128         002227          101    103    101
129         002232          114    103    125
130         002235          114    101    124
131         002240          105    104    040
132         002243          127    101    123
133         002246          040    040    040
134         002251          045    117    066
135         122 002254          045    116    045    .ASCII /#N#AVALUE IN ROM WAS #06/
136         002257          101    126    101
137         002262          114    125    105
138         002265          040    111    116
139         002270          040    122    117
140         002273          115    040    127
141         002276          101    123    040
142         002301          045    117    066
143         123 002304          045    116    045    .ASCIZ /#N#ACHECKSUM ADDRESS #06/
144         002307          101    103    110

```

GLOBAL TEXT SECTION

	002312	105	103	113	
	002315	123	125	115	
	002320	040	101	104	
	002323	104	122	105	
	002326	123	123	040	
	002331	045	117	066	
	002334	000			
124	002335	045	116	045	FOR2:: .ASCIZ /NUNABLE TO READ ADDRESS #06/
	002340	101	125	116	
	002343	101	102	114	
	002346	105	040	124	
	002351	117	040	122	
	002354	105	101	104	
	002357	040	101	104	
	002362	104	122	105	
	002365	123	123	040	
	002370	045	117	066	
	002373	000			
125					.EVEN
136					
137					
138	002374	111	116	103	MSG1:: .ASCIZ /INCORRECT CHECK SUM VALUE/
	002377	117	122	122	
	002402	105	103	124	
	002405	040	103	110	
	002410	105	103	113	
	002413	040	123	125	
	002416	115	040	126	
	002421	101	114	125	
	002424	105	000		
139	002426	124	122	101	MSG2:: .ASCIZ /TRAP 4 TIME OUT/
	002431	120	040	064	
	002434	040	124	111	
	002437	115	105	040	
	002442	117	125	124	
	002445	000			
140	002446	104	117	040	HELP1:: .ASCIZ /DO YOU WANT THE DIAGNOSTIC SUPERVISOR HELP FILE PRINTED/
	002451	131	117	125	
	002454	040	127	101	
	002457	116	124	040	
	002462	124	110	105	
	002465	040	104	111	
	002470	101	107	116	
	002473	117	123	124	
	002476	111	103	040	
	002501	123	125	120	
	002504	105	122	126	
	002507	111	123	117	
	002512	122	040	110	
	002515	105	114	120	
	002520	040	106	111	
	002523	114	105	040	
	002526	120	122	111	
	002531	116	124	105	
	002534	104	000		
141	002536	104	117	040	HELP2:: .ASCIZ /DO YOU WANT THE PROM DIAGNOSTIC HELP FILE PRINTED/
	002541	131	117	125	

GLOBAL TEXT SECTION

002544	040	127	101
002547	116	124	040
002552	124	110	105
002555	040	120	122
002560	117	115	040
002563	104	111	101
002566	107	116	117
002571	123	124	111
002574	103	040	110
002577	105	114	120
002602	040	106	111
002605	114	105	040
002610	120	122	111
002613	116	124	105
002616	104	000	

142

.EVEN

GLOBAL ERROR REPORT SECTION

151  
 152  
 153  
 154  
 155  
 156  
 157  
 158  
 159 002620  
 002620  
 160 002620  
 002620 013746 002136  
 002624 013746 002144  
 002630 013746 002142  
 002634 012746 002224  
 002640 012746 000004  
 002644 010600  
 002646 104414  
 002650 062706 000012  
 161 002654  
 002654  
 002654 104423  
 162  
 163 002656  
 002656  
 164 002656  
 002656 013746 002136  
 002662 012746 002335  
 002666 012746 000002  
 002672 010600  
 002674 104414  
 002676 062706 000006  
 165 002702  
 002702  
 002702 104423  
 166  
 182

```

.SBTTL GLOBAL ERROR REPORT SECTION

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

ERR1:: BGNMSG ERR1
PRINTB #FOR1,VAL,CKSUM,ADD
MOV ADD,-(SP)
MOV CKSUM,-(SP)
MOV VAL,-(SP)
MOV #FOR1,-(SP)
MOV #4,-(SP)
MOV SP,RO
TRAP C#PNTB
ADD #12,SP
ENDMSG
L10002: TRAP C#MSG

ERR2:: BGNMSG ERR2
PRINTB #FOR2,ADD
MOV ADD,-(SP)
MOV #FOR2,-(SP)
MOV #2,-(SP)
MOV SP,RO
TRAP C#PNTB
ADD #6,SP
ENDMSG
L10003: TRAP C#MSG

```

GLOBAL SUBROUTINES SECTION

```

184          .SBTTL GLOBAL SUBROUTINES SECTION
185
186          ;**
187          ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
188          ; THAT ARE USED IN MORE THAN ONE TEST.
189          ;--
190
191 002704     STARS
                ;*****
192          ;**
193          ; FUNCTIONAL DESCRIPTION:
194          ;     SUBROUTINE TO PRINT OUT A TEXT OR HELP FILE
195          ; INPUTS:
196          ;     NONE
197          ; OUTPUTS:
198          ;     CONTENTS OF EITHER DRS.HLP OR DIAGNOSTIC DEVICE HELP FILE
199          ;     PRINTED ON CONSOLE.
200          ;     CONTENTS OF R1 AND R2 ARE DESTROYED.
201          ; CALLING SEQUENCE:
202          ;     USES MACRO HLPMAC; FORMAT IS HLPMAC 'FILENAME.EXT'
203          ;     MACRO IS DEFINED AT END OF SUBROUTINE.
204          ;     MACRO USES FORMAT JSR R5,HLP.
205          ;--
206 002704     STARS
                ;*****
207
208 002704     $HLP::
209 002704     010537 003100     MOV     R5,#STORE           ;SAVE RETURN ADDRESS
210 002710     010500           OPEN     R5                     ;OPEN FILE NAME
211 002712     104434           MOV     R5,R0
212          ;FIRST PRINT CR/LF
213 002714     012702 000015     TRAP   C#OPEN
214 002720     004537 003002     MOV     #15,R2           ;CARRIAGE RETURN TO R2
215 002724     012702 000012     JSR    R5,60#           ;GO PRINT IT
216 002730     004537 003002     MOV     #12,R2           ;LINE FEED TO R2
217 002736     005001           JSR    R5,60#           ;GO PRINT IT
218 002742     103011           CLR     R1               ;CLEAR POSITION COUNTER
219 002744     022702 000000     10# : GETBYTE R2         ;GET BYTE COUNT OF RECORD
220 002750     001406           TRAP   C#GETB
221 002752     022702 000011     MOVB   R0,R2
222 002756     001437           BNCOMPLETE 50#         ;BRANCH IF END OF FILE
223 002760     004537 003002     BCC    50#
224 002764     000764           CMP     #0,R2           ;END OF RECORDS?
225          ;END OF FILE OR LAST RECORD
226 002766     002766           BEQ    50#             ;BRANCH IF END
227 002766     104435           30# : CMP     #11,R2     ;IS IT TAB CHARACTER?
228 002770     013705 003100     BEQ    80#             ;IF YES HANDLE TAB
229 002774     062705 000012     40# : JSR    R5,60#     ;GO PRINT IT
230 003000     000205           BR     10#             ;GO PROCESS NEXT CHARACTER
231          ;PRINT AND COUNT ROUTINE
232 003002     010237 003076     50# : CLOSE
233          ;CLOSE FILE
234          ;RESTORE RETURN ADDRESS
235          ;SKIP PAST FILE NAME
236          ;GET CHARACTER TO PRINT IN R2
                CLOSE
                TRAP   C#CLOS
                MOV     #STORE,R5
                ADD     #10.,R5
                RTS     R5
                ;PRINT AND COUNT ROUTINE
                60# : MOV     R2,#MESLNE

```



GLOBAL SUBROUTINES SECTION

```

233 003006          PRINTB  $$HLPFOR          ;GO PRINT LINE
      003006 012746 003074      MOV  $$HLPFOR, -(SP)
      003012 012746 000001      MOV  #1, -(SP)
      003016 010600          MOV  SP, R0
      003020 104414          TRAP  C#PNTB
      003022 062706 000004      ADD  #4, SP
234 003026 005201          INC  R1          ;INCREMENT POSITION COUNTER
235 003030 022701 000010      CMP  #8, R1          ;IS IT MAXIMUM?
236 003034 001406          BEQ  65$          ;IF MAXIMUM, BRANCH
237 003036 022702 000012      CMP  #12, R2         ;IS IT A LINE FEED?
238 003042 001403          BEQ  65$          ;IF LINE FEED, BRANCH
239 003044 022702 000015      CMP  #15, R2        ;IS IT A CARRIAGE RETURN?
240 003050 001001          BNE  70$          ;IF NOT, BRANCH
241 003052 005001          65$: CLR  R1          ;CLEAR POSITION COUNTER
242 003054 000205          70$: RTS  R5          ;FROM PRINT AND COUNT ROUTINE
243          ;TAB ROUTINE
244 003056          80$:
245 003056 012702 000040      MOV  #40, R2         ;MOVE SPACE CHARACTERS
246 003062 004537 003002      JSR  R5, 60$        ;GO PRINT SPACE
247 003066 005701          TST  R1          ;DONE YET?
248 003070 001372          BNE  80$          ;IF NOT DONE, AGAIN
249 003072 000721          BR   10$          ;RETURN FROM TAB ROUTINE
250          ;
251          ;STORAGE, FORMAT AND MESSAGE LINE
252          ;
253 003074          $HLPFOR::
254 003074          045      101      .ASCII  /#A/
255 003076          $MESLNE:
256 003076          000          .BYTE  0          ;RESERVED FOR MESSAGE LINE
257 003077          000          .BYTE  0
258          .EVEN
259 003100 000000          $STORE: .WORD  0          ;STORES MAIN SUBROUTINE RETURN ADDRESS
260          ;
261          ;MACRO TO INTERFACE WITH THIS SUBROUTINE
262          ;
263          .MACRO  HLPMAC  N
264          .LIST  MEB
265          .NCHR  $SYM, N
266          JSR  R5, $HLP          ;JUMP TO SUBROUTINE
267          .ASCII  /N/          ;NAME OF FILE
268          $TEMP = 10. - $SYM
269          .REPT  $TEMP
270          .BYTE  0
271          .ENDR
272          .ENDM

```

GLOBAL SUBROUTINES SECTION

274 003102

STARS

::\*\*\*\*\*

275

::++

276

:

INTERRUPT SERVICE ROUTINE

277

:

ADDRESS TRAP SERVICE ROUTINE

278

::--

279 003102

STARS

::\*\*\*\*\*

280

281 003102

BGNSRV TRAP4

003102

TRAP4::

INC TRPFLG

;SET TRAP FLAG

282 003102 005237 002146

283 003106

L10004:

ENDSRV

003106

003106 000002

RTI

284

285 003110

ENDMOD

286

GLOBAL SUBROUTINES SECTION

```

12 003110          BGNMOD
13                .TITLE MISCELLANEOUS SECTIONS
14                .SBTTL REPORT CODING SECTION
42
43                .SBTTL PROTECTION TABLE
44
45                ;**
46                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
47                ; TO PROTECT THE LOAD MEDIA.
48                ;--
49
50 003110          BGNPROT
003110          L$PROT::
51
52 003110 177777   -1          ;OFFSET INTO P-TABLE FOR CSR ADDRESS
53 003112 177777   -1          ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
54 003114 177777   -1          ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
55
56 003116          ENDPROT
57

```

INITIALIZE SECTION

```

72          .SBTTL INITIALIZE SECTION
73
74          ;**
75          ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
76          ; AT THE BEGINNING OF EACH PASS.
77          ;**
78
79 003116    BGNINIT
003116    L$INIT::
80
81 003116    SETPRI @PRI00                ; INITIALIZE WITH CPI ZERO PRIORITY
003116    012700    MOV @PRI00,RO
003122    104441    TRAP C$SPRI
82 003124    READEF @EF.CONTINUE         ; WAS THERE A CONTINUE COMMAND
003124    012700    MOV @EF.CONTINUE,RO
003130    104447    TRAP C$REFG
83 003132    BCOMPLETE ENDI              ; IF YES, DON'T SET UP NEW P-TABLE
003132    103440    BCS ENDI
84 003134    READEF @EF.NEW              ; NEW PASS OR SUB PASS?
003134    012700    MOV @EF.NEW,RO
003140    104447    TRAP C$REFG
85 003142    BNCOMPLETE NEXT             ; IF NOT NEW PASS, SKIP SETUP
003142    103003    BCC NEXT
86          ; TST HP1                    ; WAS HELP FILE PRINTED
87          ; BNE 1$                      ; BRANCH IF ALREADY PRINTED
88          ; INC HP1                      ; SET HELP FILE QUESTION FLAG
89          ; GMANIL HELP1,VAL,1,YES      ; ASK IF HELP FILE WANTED
90          ; CMP @NO,VAL                  ; DO WE WANT TO PRINT HELP FILE?
91          ; BEQ 1$                        ; IF NO, BRANCH
92          ; HLPMAC DRS.HLP              ; PRINT OUT SUPERVISOR HELP FILE
93          ;1$: TST HP2                    ; WAS HELP FILE PRINTED
94          ; BNE SETUP                    ; BRANCH IF ALREADY PRINTED
95          ; INC HP2                      ; SET HELP FILE QUESTION FLAG
96          ;2$: GMANIL HELP2,VAL,1,YES   ; ASK IF DIAGNOSTIC HELP FILE WANTED
97          ; CMP @NO,VAL                  ; DO WE WANT TO PRINT HELP FILE
98          ; BEQ SETUP                    ; IF NO, BRANCH
99          ; HLPMAC Z0794.HLP            ; PRINT OUT DIAGNOSTIC HELP FILE
100 003144    012737    177777    002156    SETUP: MOV @-1,LOGUNIT ; INITIALIZE LOGICAL UNIT COUNTER
101 003152    005237    002156    NEXT:     INC LOGUNIT          ; POINT TO NEXT LOGICAL UNIT
102 003156    023737    002156    002012    CMP LOGUNIT,L$UNIT    ; HAVE WE PASSED MAXIMUM
103 003164    001422    BEQ ABORT          ; GO IF MAXIMUM PASSED
104 003166    GPHARD LOGUNIT,PLOC         ; GET P-TABLE ADDRESS
003166    013700    002156    MOV LOGUNIT,RO
003172    104442    TRAP C$GPHRD
003174    010037    002150    MOV RO,PLOC
105 003200    BNCOMPLETE NEXT             ; IF NOT AVAILABLE, GET NEXT UNIT
003200    103364    BCC NEXT
106 003202    013737    002156    002074    MOV LOGUNIT,L$LUN    ; STORE CURRENT LOGICAL UNIT IN HEADER
107          ; GET ALL HARDWARE TABLE ENTRIES IN STORAGE
108 003210    013700    002150    MOV PLOC,RO          ; PLACE HEADER POINTER IN RO
109 003214    016037    000000    002136    MOV 0(RO),ADD      ; BASE ADDRESS OF PROM
110 003222    016037    000002    002140    MOV 2(RO),WORDS    ; NUMBER OF WORDS IN THE ROM
111 003230    000401    BR ENDI           ; SKIP ABORT INSTRUCTION
112 003232    ABORT: DOCLN                 ; DO CLEANUP AND ABORT THE PASS
003232    104444    TRAP C$DCLN
113 003234    ENDI: EXIT INIT             ; THAT'S ALL TO INITIALIZE PASS
003234    104432    TRAP C$EXIT

```

INITIALIZE SECTION

137 003236 000002  
 138  
 150  
 151  
 152  
 153 003240  
 003240  
 003240 104411

.WORD L10006 .  
  
 .EVEN  
 ENDINIT  
 L10006: TRAP C\$INIT

AUTODROP SECTION

155  
 156  
 157  
 158  
 159  
 160  
 161  
 162  
 163  
 164 003242  
       003242  
 165  
 166 003242 000240  
 173  
 174 003244  
       003244  
       003244 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
L$AUTO::

          NOP                ;NO SPECIAL CODE NEEDED

          ENDAUTO
L10007:  TRAP  C$AUTO
```

CLEANUP CODING SECTION

176  
 177  
 178  
 179  
 180  
 181  
 182  
 183 003246  
 003246  
 184  
 185 003246 012700 000004  
 003246 104436  
 003252  
 194  
 195  
 207  
 208  
 209 003254  
 003254  
 003254 104412

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

          BGNCLN
L$CLEAN:
          CLRVEC  #4           ;RESTORE TRAP VECTOR
          MOV     #4,R0
          TRAP   C$CVEC

          ENDCLN
L10010:  TRAP   C$CLEAN
```

DROP UNIT SECTION

```

211          .SBTTL  DROP UNIT SECTION
212
213          ;++
214          ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
215          ; TO NO LONGER BE TESTED.
216          ;--
217
218 003256          BGNDU
003256          L$DU::
219
220 003256 000240          NOP                      ;NO SPECIAL CODE NEEDED
229
230
242
243
244 003260          ENDDU
003260          L10011:
003260 104453          TRAP  C$DU

```



## ADD UNIT SECTION

```
246          .SBTTL  ADD UNIT SECTION
247
248          ;**
249          ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
250          ; TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
251          ; TO THE TEST CYCLE.
252          ;--
253
254 003262          BGNAU
003262          L#AU::
255
256 003262 000240          NOP                      ;NO SPECIAL CODE NEEDED
265
266
267
268
269
270 003264          ENDAU
003264          L10012:
003264 104452          TRAP  C#AU
281
282 003266          ENDMOD
283
```

H3

ADD UNIT SECTION

1  
2  
13  
49  
50 003266

.TITLE HARDWARE TESTS  
.ENABL AMA

BGNMOD

## TEST 1:

```

52          .SBTTL TEST 1:
53 003266   STARS
           ;*****
54          ;**
55          ;TEST DESCRIPTION
56          ; DOES A CHECKSUM CHECK ON A PROM
57          ; EXPECTS A CHECK SUM VALUE TO BE IN THE LAST ADDRESS OF THE PROM
58          ;TEST STEPS
59          ; SET UP FOR BAD ADDRESS TRAP, ABORT IF BAD ADDRESS OCCURS
60          ; READ THE PROM FROM STARTING ADDRESS EXCEPT LAST ADDRESS
61          ; SUM EACH PROM VALUE TO PREVIOUS TOTAL FOR CALCULATING CHECK SUM
62          ; COMPARE CHECK SUM VALUE WITH CONTENTS OF LAST ADDRESS
63          ;--
64 003266   STARS
           ;*****
65
66 003266   BGNTST
67 003266   T1::
           SETVEC  #4,#TRAP4,#PRI07      ; SET FOR BAD ADDRESS TRAP
           MOV     #PRI07,-(SP)
           MOV     #TRAP4,-(SP)
           MOV     #4,-(SP)
           MOV     #3,-(SP)
           TRAP   C#SVEC
           ADD     #10,SP
68          ;
69          ; FIRST SET UP PROM MAPPING.
70          ; IF ITS A 11/23A WE PROBABLY WILL GET A BAD ADDRESS
71          ; TRAP AND IGNORE IT.
72          ;
73 003314   CLR     TRPFLG                ; CLEAR TRAP INDICATOR
74 003320   MOV     #400,#CS#PCR         ; MAP TO PROM PAGES 0 AND 1
75 003326   MOV     TRPFLG,CPUTYP       ; STORE RESULT OF BAD ADDRESS TRAP
76          ;
77          ; 0 = LSI 11/23B
78          ; 1 = LSI 11/23A
79          ;
80          ; SET UP TO READ PROM
81          ;
82          MOV     ADD,R1                ; GET STARTING ADDRESS IN R1
83          MOV     WORDS,R2             ; NUMBER OF WORDS TO CHECK IN R2
84          DEC     R2                   ; SKIP LAST ROM ADDRESS
85          CLR     R3                   ; CLEAR R3 AS ACCUMULATOR
86          CLR     TRPFLG               ; CLEAR TRAP INDICATOR
87 003354   LOOP:  MOV     (R1)+,R4       ; READ PROM
88          TST     TRPFLG               ; DID INTERRUPT OCCUR?
89          BNE     BADADD                ; IF YES, BRANCH
90          ADD     R4,R3                 ; ADD VALUE TO ACCUMULATOR
91          DEC     R2                   ; SUBTRACT COUNTER
92          BNE     LOOP                 ; IF NOT DONE, LOOP
93          ;
94          ; NOW READ CHECK SUM VALUE
95          ;
96 003372   MOV     (R1),CKSUM           ; MOVE ACCUMULATED CHECK SUM TO VAL
97 003376   MOV     R3,VAL               ; ARE THEY THE SAME?
98 003402   CMP     CKSUM,VAL
99 003410   BEQ     END                  ; IF YES, END TEST

```

TEST 1:

```

100
101
102
103 003412 010137 002136
104 003416 104456
    003420 000001
    003422 002374
    003424 002620
105 003426
    003426 104432
    003430 000022
106
107
108
109 003432
110 003432 162701 000002
111 003436 010137 002136
112 003442
    003442 104455
    003444 000002
    003446 002426
    003450 002656
113 003452
    003452
    003452 104401
114
115 003454
116

```

```

;
; ERROR, CKSUM IS NOT RIGHT
;
MOV R1,ADD ; ADDRESS OF CHECK SUM IN ADD
ERRHRD 1,MSG1,ERR1 ; TELL OPERATOR ERROR
TRAP C#ERRHRD
.WORD 1
.WORD MSG1
.WORD ERR1
EXIT TST ; END OF TEST
TRAP C#EXIT
.WORD L10013-.

;
; ERROR, NON EXISTENT MEMORY
;
BADADD:
SUB #2,R1 ; CORRECT ADDRESS
MOV R1,ADD ; PLACE ADDRESS IN MEMORY FOR MESSAGE
ERRDF 2,MSG2,ERR2 ; TELL OPERATOR BAD ADDRESS
TRAP C#ERRDF
.WORD 2
.WORD MSG2
.WORD ERR2
END:
L10013: TRAP C#ETST
ENDMOD

```

TEST 1:

1  
12  
13  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59

003454  
003454 000011  
003456  
003456 000031  
003460 003500  
003462 000000  
003464 177776  
003466 001052  
003470 003550  
003472 177777  
003474 000000  
003476 000400  
003500  
127 110 101  
124 040 111  
123 040 124  
110 105 040  
123 124 101  
122 124 111  
116 107 040  
101 104 104  
122 105 123  
123 040 117  
106 040 124  
110 105 040  
122 117 115  
000  
116 125 115  
102 105 122  
040 117 106  
040 127 117  
122 104 123  
040 111 116  
040 124 110  
105 040 122  
117 115 040  
040 040 040

```
.TITLE PARAMETER CODING
.SBTTL  HARDWARE PARAMETER CODING SECTION
      BGNMOD

; **
; 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
      .WORD L10014-L$HARD/2
L$HARD::
      GPRMA  G1,0,0,0,177776,YES      ;BASE ADDRESS
      .WORD  T$CODE
      .WORD  G1
      .WORD  T$LOLIM
      .WORD  T$HILIM
      GPRMD  G2,2,D,-1,0,256.,YES    ;NUMBER OF WORDS
      .WORD  T$CODE
      .WORD  G2
      .WORD  -1
      .WORD  T$LOLIM
      .WORD  T$HILIM
      ENDRD
      .EVEN

L10014:
G1::  .ASCIZ  /WHAT IS THE STARTING ADDRESS OF THE ROM/

G2::  .ASCIZ  /NUMBER OF WORDS IN THE ROM /
```

HARDWARE PARAMETER CODING SECTION

	003606	040	040	040	
	003611	040	040	040	
	003614	040	040	040	
	003617	000			
60					.EVEN
61					
68					
69					
70	003620				\$PATCH::
71	003620				.BLKW    10
72					
79					
80	003640				LASTAD
					.EVEN
	003640	000000			.WORD    0
	003642	000000			.WORD    0
	003644				L\$LAST::
81	003644				ENDMOD

M3

PARAMETER CODING      MACRO M1200    09-JAN-84 12:23    PAGE 30

SEQ 0038

HARDWARE PARAMETER CODING SECTION

83  
84  
97

000001

.END





SYMBOL TABLE

T\$LOLI= 000000	T\$SUBN= 000000	T\$\$CLE= 010010	T\$\$SW = 010001	X\$OFFS= 000400
T\$LSYM= 010000	T\$TAGL= 177777	T\$\$DU = 010011	T\$\$TES= 010013	X\$TRUE= 000020
T\$LTNO= 000001	T\$TAGN= 010015	T\$\$HAR= 010014	T1 = 003266 G	YES = 000001 G
T\$NEST= 177777	T\$TEMP= 000000	T\$\$HW = 010000	UAM = 000200 G	\$HLP 002704 G
T\$NSO = 000000	T\$TEST= 000001	T\$\$INI= 010006	VAL = 002142 G	\$HLPFO 003074 G
T\$NS1 = 000004	T\$TSTM= 177777	T\$\$MSG= 010003	WORDS 002140 G	\$MESLN 003076
T\$PTNU= 000000	T\$TSTS= 000001	T\$\$PRO= 010005	X\$ALWA= 000000	\$PATCH 003620 G
T\$SAVL= 177777	T\$\$AU = 010012	T\$\$SRV= 010004	X\$FALS= 000040	\$STORE 003100
T\$SEGL= 177777	T\$\$AUT= 010007			

. ABS. 003644 000  
 000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 28504 WORDS ( 112 PAGES)

DYNAMIC MEMORY: 20060 WORDS ( 77 PAGES)

ELAPSED TIME: 00:01:36

ZFPDA0.BIC,CZFPDA0/-SP=SVC/ML,ZFPDA01,ZFPDA02,ZFPDA03,ZFPDA04,ZFPDA05,ZFPDA06