

DZV 11

DZV-11 DIAG PART 2
CVDZBDO

COPYRIGHT (c) 1977-84
AH-A939D-MC
FICHE 01 OF 01

FEB 1985
digital
Made In USA

The microfiche card displays a grid of 100 frames of technical diagrams and data. The frames are arranged in 10 rows and 10 columns. The diagrams include various charts, tables, and flowcharts, all rendered in white on a dark background. A small white circular mark is visible at the bottom center of the card.

IDENTIFICATION

PRODUCT CODE: AC-A9380-MC
 PRODUCT NAME: CVDZB00 DZV11 DIAG PRT2
 DATE RELEASED: MARCH 1983
 MAINTAINER: DIAGNOSTIC ENGINEERING

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.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

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

COPYRIGHT (C) 1983,1984 DIGITAL EQUIPMENT CORPORATION

CVDZBO
CVDZBO.P11

21-AUG-84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 3 1

AN LSI11 CPU WITH MINIMUM 4K OF MEMORY.
ASR 33 (OR EQUIVALENT FOR CONSOLE)
DZV11 INTERFACE MODULE
H329 STAGGERED TURNAROUND CONNECTOR.
H325 CABLE TURNAROUND CONNECTOR.

NOTE: A STAGGERED TURNAROUND CONNECTOR IS NEEDED IN ORDER TO TEST THE
PARITY LOGIC.

CVDZBO
CVDZBO.P11

21-AUG-84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 4

PAGE 3

2.2 STORAGE

PROGRAM WILL USE ALL 4K OF MEMORY EXCEPT WHERE ABL AND BOOTSTRAP LOADER RESIDE. LOCATION 1500 THRU 1740 ARE ESPECIALLY TO BE NOTED AND TO BE UNTOUCHED BY OPERATOR AFTER PARAMETERS HAVE BEEN INPUT FROM CONSOLE (SM00=1); OR AFTER THE 'AUTO SIZING' HAS BEEN DONE. THESE LOCATIONS MAY BE CHANGED IF THE USER UNDERSTANDS THEIR MEANING AND DIFFERENT PARAMETERS ARE REQUIRED.

3. LOADING PROCEEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND ARE LOADED USING THE ABSOLUTE LOADER. NOTE: IF THE DIAGNOSTICS ARE ON A MEDIA SUCH AS DISK ,MAGTAPE,DECTAPE, OR CASSETTE; FOLLOW INSTRUCTIONS FOR THE MONITOR WHICH HAS BEEN PROVIDED ON THAT SPECIFIC MEDIA.

ABSOLUTE LOADER STARTING ADDRESS *500

MEMORY * SIZE

4K	17
8K	37
12K	57
16K	77
20K	117
24K	137
28K	157

3.1.1 STARTING THE PROCESSOR AT THE ABSOLUTE LOADER STARTING ADDRESS WILL LOAD THE DIAGNOSTIC INTO MEMORY.

4. STARTING PROCEDURE

- A. SET SWR TO ZERO FOR 'AUTO SIZING' OR SET SW00=1 FOR USER PARAMETER INPUT FROM CONSOLE TERMINAL. NOTE: LOC. 000176 IS USED AS A SOFTWARE SWITCH REGISTER IN ALL OF THE DZV11 DIAGNOSTICS. (SEE SEC. 4.1) ON THE FIRST STARTUP OF THE DIAGNOSTIC IF SW07=1 AND SW00=0 THE PROGRAM WILL ASSUME THAT THE STATUS TABLE HAS BEEN ALREADY BUILT FROM A PREVIOUS DZV11 DIAGNOSTIC RUN. NOTE: ANY DZV11 DIAGNOSTIC WILL OVERLAY THE STATUS TABLE WHEN LOADED TO PRESERVE ITS CONTENTS AND THUS WILL NOT ALTER A PREVIOUSLY BUILT TABLE.
- B. START THE DIAGNOSTIC AT LOC. 200(8). THE PROGRAM WILL TYPE MAINDEC AND PROGRAM NAMES (IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO THE FOLLOWING: (ON THE FIRST PROGRAM RUN OR IF PARAMETERS WERE CHANGED)

```
'MAP OF DZV11 STATUS'
1500 160100
1502 000300
1504 000017
1506 017470
1510 000000
```

THE ABOVE IS ONLY AN EXAMPLE! THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADD. 1500 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE USER IF AUTO SIZING IS DONE. FOR INFORMATION OF STATUS TABLE SEE SECTION 8.4 FOR HELP.

THE PROGRAM WILL TYPE "RUNNING" AND PROCEED TO RUN THE DIAGNOSTIC.

4.1 CONTROL SWITCH SETTINGS

NOTE: THIS PROGRAM UTILIZES A SOFTWARE SWITCH REGISTER WHICH MAY BE MODIFIED BY CHANGING LOC. 176 OR BY TYPING CONTROL "G" (+G) ON THE CONSOLE TERMINAL WHILE THE PROGRAM IS RUNNING.

```
SW 15 SET: HALT ON ERROR
SW 14 SET: LOOP ON CURRENT TEST
SW 13 SET: INHIBIT ERROR PRINT OUT
SW 12 SET: INHIBIT **ALL** TYPE OUT/BELL ON ERROR.
SW 11 SET: INHIBIT ITERATIONS. (QUICK PASS)
SW 10 SET: ESCAPE TO NEXT TEST
SW 09 SET: LOOP WITH CURRENT DATA
SW 08 SET: CATCH ERROR AND LOOP ON IT
SW 07 SET: NO AUTO SIZE. IF 1ST START OF PROGRAM AFTER LOADING AND
IF SW00=0 THEN THE PROGRAM WILL ASSUME THAT THE STATUS MAP
HAS BEEN BUILT FROM A PREVIOUS DZV11 DIAGNOSTIC RUN.

SW 06 SET: RESELECT DZV11'S DESIRED ACTIVE
SW 05 SET: RESERVED
SW 04 SET: SELECT DELAY PARAMETER (SEE SEC. 4.1.1)
SW 03 SET: EXTRA PARAMETER INPUT (SEE SEC. 4.1.1)
SW 02 SET: LOCK ON SELECTED TEST
SW 01 SET: RESTART PROGRAM AT SELECTED TEST
```

CVDZBD
CVDZBD.P11

21 AUG-84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 5-1

SM 00 SET: GET USERS PARAMETERS FROM CONSOLE

4.1.1 SWITCH REGISTER CONTROL OF PARAMETER INPUT FROM CONSOLE

- SW 00 GET USERS PARAMETERS FROM CONSOLE. SETTING THIS SWITCH AT START UP TIME ALLOWS THE USER TO INPUT AT THE CONSOLE TERMINAL THE FOLLOWING PARAMETERS: BASE DEVICE ADDRESS, BASE VECTOR ADDRESS, MODE OF OPERATION (EXTERNAL, INTERNAL, OR STAGGERED), AND THE NUMBER OF DZV11'S THAT ARE RUNNING. USING THIS SWITCH ALONE WILL DEFAULT THE FOLLOWING PARAMETERS: ALL 4 LINES ARE SET TO BE TESTED ON EACH DZV11, THE DEFAULT BAUD RATE IS SET AT 19.2 KBAUD AND THE CHARACTER LENGTH FOR THE MAJORITY OF TESTING IS SET AT EIGHT BITS PER CHARACTER WITH TWO STOP BITS.
- SW 03 EXTRA PARAMETER INPUT. SETTING THIS SWITCH AT START UP TIME PROVIDES THE USER WITH THE ABILITY TO SET THE LINES ACTIVE FOR TESTING AND TO SET THE DEFAULT BAUD RATE USED FOR THE MAJORITY OF THE DIAGNOSTIC TESTS. THE DELAY PARAMETER IS AUTOMATICALLY ADJUSTED TO THE BAUD RATE GIVEN BY THE USER.
- SW 04 SELECT DELAY PARAMETER. THE DELAY PARAMETER THIS SWITCH CONTROLS DETERMINES THE LENGTH OF TIME THE PROGRAM STALLS WAITING FOR A CHARACTER TO BE COMPLETELY TRANSMITTED OR RECEIVED. THIS DELAY COUNT IS AUTOMATICALLY SET TO PROVIDE ENOUGH DELAY TIME FOR THE DEFAULT BAUD RATE SPECIFIED WHEN RUNNING THE PROGRAM ON AN LSI11 WITH MOS MEMORY. WHEN RUNNING THIS PROGRAM ON A PROCESSOR WITH A FASTER MEMORY SPEED THIS DELAY COUNT SHOULD BE ADJUSTED PROPORTIONATELY HIGHER THAN THE FOLLOWING DEFAULTED VALUES:
- | | | |
|------|------------|------------|
| 2450 | ; TIME FOR | 50 BAUD |
| 1560 | ; TIME FOR | 75 BAUD |
| 1120 | ; TIME FOR | 110 BAUD |
| 0750 | ; TIME FOR | 134 BAUD |
| 0660 | ; TIME FOR | 150 BAUD |
| 0330 | ; TIME FOR | 300 BAUD |
| 0150 | ; TIME FOR | 600 BAUD |
| 0060 | ; TIME FOR | 1200 BAUD |
| 0040 | ; TIME FOR | 1800 BAUD |
| 0030 | ; TIME FOR | 2000 BAUD |
| 0020 | ; TIME FOR | 2400 BAUD |
| 0010 | ; TIME FOR | 3600 BAUD |
| 0001 | ; TIME FOR | 4800 BAUD |
| 0001 | ; TIME FOR | 7200 BAUD |
| 0001 | ; TIME FOR | 9600 BAUD |
| 0001 | ; TIME FOR | 19.2 KBAUD |

CVDZBO
CVDZBO.P11 21-AUG-84 08:28

;;GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 7

PAGE 6

4.1.2 SWITCH REGISTER RESTRICTIONS

- SW 06 RESELECT DZV11'S DESIRED ACTIVE. A MESSAGE IS TYPED OUT ON THE CONSOLE TERMINAL ASKING THE OPERATOR TO TYPE A BIT MAP OF THE DZV'S DESIRED ACTIVE. USING THIS SWITCH ALLOWS LOCATION DZVACTV TO BE ALTERED (SEE SEC. 8.3 FOR A DESCRIPTION OF THIS LOCATION).
EXAMPLE:
IF THE DEVICES CORRESPONDING TO THE DZV11'S NUMBERED ZERO, TWO, AND FOUR IN THE DZV11 STATUS MAP (LOC. 1500 THROUGH 1740) ARE TO BE TESTED, TYPE IN: 25
THIS WILL SET BITS ZERO, TWO, AND FOUR IN LOCATION DZVACTV. ALL REMAINING DEVICES IN THE STATUS MAP WILL THEN NOT BE TESTED.
- SW 01 RESTART PROGRAM AT SELECTED TEST IT IS STRONGLY SUGGESTED THAT AT LEAST ONE PASS HAS BEEN MADE BEFORE TRYING TO SELECT A TEST THAT IS NOT IN THE ORDER OF SEQUENCE THE REASON BEING IS THAT THE PROGRAM HAS TO CLEAR AREAS AND SET UP PARAMETERS.
NOTE: IF RUNNING MULTIPLE DZV11'S, THE DZV11 YOU DESIRE TO BE UNDER TEST MUST BE SELECTED BY THE USE OF SW06 BEFORE LOCKING ON THE TEST. IN OTHER WORDS, EACH TIME THE PROGRAM IS STARTED, THE FIRST DZV11 WILL BE SELECTED TO BE UNDER TEST UNLESS SW06 IS USED TO SELECT ONLY ONE.
- SW 09 LOOP ON CURRENT DATA; THIS SWITCH WILL ONLY WORK IF CALL 'SCOPI' IS IN THAT TEST. THE REASON BEING THAT MOST TESTS DEAL WITH BLOCKS OF DIFFERENT DATA TO BE SENT OR RECEIVED ALL AT ONCE THUS IN BLOCK DATA, ONE PATTERN CAN'T BE SINGLED OUT.
THIS SWITCH IS DESIGNED TO PROVIDE AN AID FOR A TRAINED TROUBLE-SHOOTER TO SAMPLE VARIOUS SIGNALS ON THE MODULE AND IS NOT MEANT TO BE USED AS A GENERAL USER CONTROL SWITCH.
- SW 04 SELECT DELAY PARAMETER; THIS SWITCH SHOULD BE USED WITH CARE AS TOO SHORT A DELAY WILL CAUSE VALID TESTS TO FAIL.
(SEE SEC. 4.1.1)

CVDZ80
CVDZ80.P11 21-AUG-84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 8

PAGE 7

4.1.3 SWITCH REGISTER PRIORITIES

ERROR SWITCHES

1. SW 12 DELETE PRINT OUT/BELL ON ERROR.
2. SW 13 DELETE ERROR PRINTOUT.
3. SW 15 HALT ON THE ERROR.
4. SW 08 GO TO BEGINNING OF THE TEST(ON ERROR).
5. SW 10 GOTO NEXT TEST(ON ERROR).

SCOPE SWITCHES

1. SW 09 (IF ENABLED BY 'SCOPI'). IF AN '*' IS PRINTED IN FRONT OF THE TEST NO. ON AN ERROR REPORT (EX. *TEST NO. 10) SW09 IS INCORPORATED IN THAT TEST AND THEREFORE SW09 IS *USUALLY* THE BEST SWITCH FOR THE SCOPE LOOP (SW14=0, SW10=0, SW09=1, SW08=0) IF THE PROGRAM USER IS TECHNICALLY TRAINED TO ELECTRONICALLY ISOLATE SIGNAL PROBLEMS ON THE DZV11 MODULE. IF SW09 IS NOT ENABLED, AND THERE IS A *HARD* ERROR (CONSTANT), SW08 IS BEST.
2. FOR INTERMITTENT ERRORS EITHER START THE PROGRAM WITH SW01 AND SW02 SET WHICH WILL ALLOW THE USER TO LOCK ON A SELECTED TEST. OR ELSE SET SW14 AS AN ERROR IS BEING TYPED OUT ON THE TERMINAL. SW14 WILL CONTINUE TO LOOP ON THAT TEST REGARDLESS OF WHETHER AN ERROR OCCURS.
3. SW 14 LOOP ON CURRENT TEST.

4.2 STARTING ADDRESS

SA 200 - THE STARTING ADDRESS FOR ANY DZV11 DIAGNOSTIC IS LOC. 200

NOTE: IF ADDRESS 000042 IS NON-ZERO THE PROGRAM ASSUMES IT IS UNDER ACT11 OR XXDP CONTROL AND WILL ACT ACCORDINGLY. AFTER *ALL* AVAILABLE DZV11S ARE TESTED THE PROGRAM WILL RETURN TO 'XXDP' OR 'ACT-11'.

5 OPERATING PROCEDURE

WHEN THE PROGRAM IS INITIALLY STARTED, MESSAGES AS DESCRIBED IN SECTION FOUR WILL BE PRINTED AND THE DIAGNOSTIC WILL BEGIN RUNNING.

5 1 NORMAL START OF DIAGNOSTIC

ON THE FIRST START OF THE DIAGNOSTIC AT ADDRESS 200, IF SW00-1 THEN THE FOLLOWING QUESTIONS ARE ASKED AND MUST BE ANSWERED:

"1ST CSR ADDRESS (160000:167770): "
YOU MUST TYPE IN THE FIRST DZV11 CSR IN THE SYSTEM YOU WISH TESTING TO BEGIN AT. RANGE: 160000:167770

"1ST VECTOR ADDRESS (300:770): "
YOU MUST TYPE IN THE VECTOR OF THE FIRST DZV11 IN THE SYSTEM UNDER TEST. RANGE 300:770

"MAINTENANCE MODE

[EXTERNAL <M325> (E)]
[INTERNAL <DZCSR03=1>(I)]
[STAGGERED <M329> (S)] :

TYPE "E" OR "I" OR "S" DEPENDING ON WHICH MODE YOU WISH TO RUN IN. IF RUNNING "EXTERNAL", ALL SELECTED LINES MUST BE TERMINATED BY AN M325 TEST CONNECTOR.

"# OF DZV11'S <IN OCTAL> (1:20): "
TYPE TOTAL NUMBER OF DZV11'S TO BE TESTED IN THE SYSTEM. RANGE IS 1 THRU 20 IN OCTAL.

***** IF SW03=1 THEN THE FOLLOWING WILL BE PRINTED *****

"LINES ACTIVE BY BIT <IN OCTAL> (001:017):"
EACH BIT REPRESENTS A LINE AND ANY COMBINATION OF LINES MAY BE SELECTED (HOWEVER IN STAGGERED MODE TWO ADJACENT LINES MUST BE SELECTED (0-1, 2-3).

"DEFAULT BAUD RATE <IN OCTAL> (00:17): "
THIS GIVES THE USER A CHANCE TO CHANGE THE DEFAULT BAUD RATE USED IN APP. 90% OF THE TEST. BAUD RATE CHOICES ARE:
"00"(50 BAUD), "01"(75 BAUD), "02"(110 BAUD), "03"(134 BAUD),
"04"(150 BAUD), "05"(300 BAUD), "06"(600 BAUD), "07"(1200 BAUD),
"10"(1800 BAUD), "11"(2000 BAUD), "12"(2400 BAUD), "13"(3600 BAUD),
"14"(4800 BAUD), "15"(7200 BAUD), "16"(9600 BAUD), "17"(19.2 KBAUD)
LOW DEFAULT BAUD RATES ARE NOT SUGGESTED SINCE THEY LENGTHEN THE TIME TO COMPLETE A PROGRAM PASS DRAMATICALLY.

IT IS IMPORTANT TO NOTE THAT ALL DZV11'S IN THE SYSTEM MUST BE CONTIGIOUS FOR BOTH ADDRESS AND VECTORS. ALSO ALL THE EXTRA PARAMETERS OTHER THAN CSR AND VECTORS ARE GIVEN TO THE EXISTING DZV11'S IN THE SYSTEM.

IF THE MODE OF OPERATION IS DIFFERENT FOR EACH DZV11 THIS MUST BE PATCHED INTO THE CORRECT STATUS MAP ENTRY WHICH IS PRINTED AT START TIME. AN ALTERNATIVE IS TO PUT SW00-1 AT START TIME; ANSWER QUESTIONS ABOUT DZV11 UNDER TEST AND INDICATE ONE DZV11 IN THE SYSTEM. IF THE STATUS MAP IS TO BE "PATCHED" IT MUST BE DONE AFTER THE QUESTIONS ARE ANSWERED OR AFTER THE AUTO SIZE.

5.2 PROGRAM AND/OR OPERATOR ACTION

THE VARIETY OF PROGRAM CONTROL SWITCHES PROVIDED IN THIS DIAGNOSTIC PACKAGE IS DESIGNED TO PROVIDE THE USER WITH A WIDE RANGE OF TROUBLE-SHOOTING TECHNIQUES. BEFORE THE USER ATTEMPTS TO RUN THIS DIAGNOSTIC HE SHOULD BECOME FAMILIAR WITH THE USE OF THESE CONTROL SWITCHES AND THEIR RESTRICTIONS. (SEE SEC. 4.1, 4.1.1, 4.1.2, 4.1.3)

WHEN THE PROGRAM DETECTS AN ERROR THE TEST NUMBER AND PC WILL BE TYPED OUT AND POSSIBLY AN ERROR MESSAGE (DEPENDING ON THE PARTICULAR ERROR). IF IT IS NECESSARY TO KNOW MORE INFORMATION CONCERNING THE ERROR REPORT THEN LOOK IN THE PROGRAM LISTING FOR THAT TEST NUMBER AND THEN NOTE THE PC OF THE ERROR REPORT. THE REASON FOR THE ERROR REPORT WILL BECOME CLEARER WHEN READING THE COMMENTS IN THE PROGRAM LISTING.

6 ERRORS

AS DESCRIBED PREVIOUSLY THERE WILL ALWAYS BE A TEST NUMBER AND PC TYPED OUT AT THE TIME OF AN ERROR (PROVIDING SW 13=0 AND SW 12=0). IN MOST CASES ADDITIONAL INFORMATION WILL BE SUPPLIED TO THE THE ERROR MESSAGE WHICH IS TO GIVE THE OPERATOR AN INDICATION OF THE ERROR.

6.1 ERROR RECOVERY

IF FOR SOME REASON THE DZV11 SHOULD 'HANG THE BUS' (GAIN CONTROL OF BUS SO THAT CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT OR POWER DOWN/UP IS NECESSARY FOR OPERATOR TO REGAIN CONTROL OF CPU. IF THIS SHOULD HAPPEN, LOOK IN LOCATION '†TSTN†' (ADDRESS 1246) FOR THE NUMBER OF THE TEST THAT WAS RUNNING AT THE TIME OF THE CATASTROPHIC ERROR. IN THIS WAY THE OPERATOR WILL HAVE AN IDEA AS TO WHAT THE DZV11 WAS DOING AT THE TIME OF THE ERROR.

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

SEE SECTION 4.1.2
THE STATUS TABLE SHOULD BE VERIFIED REGARDLESS OF HOW THE PROGRAM WAS STARTED. ALSO IT IS IMPORTANT TO USE THIS LISTING ALONG WITH THE INFORMATION PRINTED ON THE TTY TO COMPLETELY ISOLATE PROBLEMS.

CVDZBO
CVDZBO.P11 21 AUG 84 08:28

;;GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 11

PAGE 10

7.2 OPERATING RESTRICTIONS

PARAMETER MUST BE INPUT FROM USER OR APT IF "AUTO SIZING" IS NOT USED.

8 MISCELLANEOUS

8.1 EXECUTION TIME

ALL DZV11 DEVICE DIAGNOSTICS WILL GIVE AN 'END PASS' MESSAGE (PROVIDING NO ERRORS AND SW12=0) WITHIN 2 MIN. THIS IS ASSUMING SW11=1 (INHIBIT ITERATIONS) IS SET TO GIVE THE FASTEST POSSIBLE EXECUTION.

8.2 PASS COMPLETE

NOTE: *EVERY* TIME THE PROGRAM IS STARTED, THE TESTS WILL RUN AS IF SW11 (DELETE ITERATIONS) WAS UP (=1). THIS IS TO 'VERIFY NO *HARD* ERRORS' AS SOON AS POSSIBLE. THEREFORE THE FIRST PASS -EACH TIME PROGRAM IS STARTED- WILL BE A 'QUICK PASS' UNTIL ALL DZV11'S IN SYSTEM ARE TESTED. WHEN THE DIAGNOSTIC HAS COMPLETED A PASS THE FOLLOWING IS AN EXAMPLE OF THE PRINT OUT TO BE EXPECTED.

END PASS DVDZB-D CSR: 160100 VEC: 300 PASSES: 000001 ERRORS: 000000

NOTE: THE NUMBERS FOR CSR AND VEC ARE NOT NECESSARILY THE VALUES FOR THE DEVICE. THEY ARE ONLY FOR THIS EXAMPLE.

CVDZBO
CVDZBO.P11 21-AUG 84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 12

PAGE 11

8.3 KEY LOCATIONS

1LPADR (1252) CONTAINS THE ADDRESS WHERE PROGRAM WILL RETURN WHEN ITERATION COUNT IS REACHED OR IF LOOP ON TEST IS ASSERTED.

NEXT (1362) CONTAINS THE ADDRESS OF THE NEXT TEST TO BE PERFORMED.

1TS/NM (1246) CONTAINS THE NUMBER OF THE TEST NOW BEING PERFORMED.

RUN (1412) THE BIT IN 'RUN' ALWAYS POINTS ONE PAST THE DZV11 CURRENTLY BEING TESTED. EXAMPLE: (RUN) 1412/0000000001000000 MEANS THAT DZV11 NO.5 IS THE DZV11 NOW RUNNING.

STATUS MAP (1500)-(1740) THESE LOCATIONS CONTAIN THE INFORMATION NEEDED TO TEST UP TO 16 (DECIMAL) DZV11S SEQUENTIALY. THEY CONTAIN THE CSR, VECTOR AND STATUS CONCERNING THE CONFIGURATION OF EACH DZV11.

DZVACTV(1406) EACH BIT SET IN THIS LOCATION INDICATES THAT THE ASSOCIATED DZV11 WILL BE TESTED IN TURN. EXAMPLE: (DZVACTV) 1406/0000000000011111 MEANS THAT DZV11 NO. 00,01,02,03,04 WILL BE TESTED. EXAMPLE: (DZVACTV) 1406/0000000000010001 MEANS THAT DZ11 NO. 00,04 WILL BE TESTED.

1BASE (1174) CONTAINS THE RECEIVER CSR OF THE CURRENT DZV11 UNDER TEST.

CVDZBO
CVDZBO.P11

21-AUG-84 08:28

PAGE 12

8 4 MORE ON THAT 'STATUS TABLE' (1500-1740)

'MAP OF DZV11 STATUS'	
1500	160100
1502	000300
1504	000017
1506	017470
1510	000000

THE ABOVE INFORMATION WILL BE REPEATED FOR EACH OF UP TO 16 DZV11'S IN THE SYSTEM (THESE WILL FOLLOW UNDER THIS TABLE). EXPLANATION:

1500	160100	THIS IS THE SYSTEM CONTROL REGISTER FOR THE 1ST DZV11 IN THE SYSTEM.
1502	000300	THIS IS VECTOR 'A' FOR THE FIRST DZV11 IN THE SYSTEM.
1504	000017	THIS IS THE BINARY REPRESENTATION OF WHAT LINES ARE TO BE TESTED.
1506	017470	THIS IS THE PARAMETER LOCATION USED IN MOST OF THE TESTS. IT INDICATES PARAMETERS OF: RX ON, SPEED SELECT 17 (19.2K BAUD) EIGHT BITS PER CHAR, AND TWO STOP BITS. THE USER MAY ALTER THE STOP BITS AND THE SPEED, BUT THE REMAINING PARAMETERS SHOULD BE LEFT ALONE. THIS LOCATION IS USED TO LOAD THE DZV11 LINE PARAMETER REGISTER FOR EACH LINE. THE MEANING OF THE BITS SET IN THIS LOCATION IS THE SAME AS THE FUNCTION OF THE RELATED BITS IN THE DEVICE LINE PARAMETER REGISTER.
1510	000000	THIS LOCATION WILL CONTAIN EITHER ALL ZEROS INDICATING THAT INTERNAL LOOP WAS SELECTED AS MODE OF OPERATION OR IT WILL CONTAIN 100000 INDICATING THAT "STAGGERED MODE" WAS SELECTED OR IT WILL CONTAIN 000200 INDICATING THAT "EXTERNAL" WAS THE MODE SELECTED.

THE ABOVE IS REPEATED FOR EACH DZV11 IN THE SYSTEM. THE TABLE IS FILLED BY AUTO SIZING OR BY THE MANUAL PARAMETER INPUT PROGRAM AS DESCRIBED PREVIOUSLY. ALSO IF DESIRED BY USER, THE LOCATIONS MAY BE ALTERED BY HAND TO SUIT THE SPECIFIC CONFIGURATION.

CVDZBO
CVDZBO.P11 21-AUG-84 08:28

PAGE 13

8.5 *** METHOD OF AUTO SIZING ***

8.5.1 FINDING THE CONTROL STATUS REGISTER.

THE PROGRAM WILL START AT ADDRESS 160000 AND START 'REFERENCING' THE ADDRESS IN THE POINTER. IF A NON-EX MEMORY TRAP OCCURS, THE POINTER (HOLDING 160000) IS UPDATED BY 10 AND THE ABOVE IS REPEATED UNTIL ADDRESS 167770 IS REACHED. IF A 'BUS REPLY' RESPONSE WAS ISSUED BY THE DZV11 (OR ANY OTHER DEVICE) (NO NON TRAP), "MASTER SCAN ENABLE" IS ATTEMPTED TO BE SET AND THE TCR BITS FOR ALL FOUR LINES ARE SET. "TRDY" IS THEN TESTED TO BE SET AND "MASTER SCAN ENABLE" IS TESTED TO BE STILL SET. THE DIAGNOSTIC WILL THEN CHECK THAT AT LEAST ONE TCR BIT IS STILL SET. IF ALL OF THE ABOVE WORKED, THIS DEVICE IS ASSUMED TO BE A DZV11. IF ANY OF THE ABOVE FAILED, UPDATING OF THE POINTER IS DONE AND THE SEQUENCE IS REPEATED.

NOTE: IF THE PROGRAM DOES NOT FIND YOUR DZV11, SOMETHING IS WRONG AND AUTO SIZING SHOULD NOT BE DONE.

8.5.2 FINDING THE VECTOR

THE VECTOR AREA (ADDRESS 300-776) IS FILLED WITH THE INSTRUCTION IOT AND '.+2' (NEXT ADDRESS). BIT14 AND BITS (TX INTERRUPT ENABLE AND MSTSCAN ENABLE) ARE SET INTO THE DZVCSR. ALL TCR BITS ARE SET, A DELAY OCCURS, AND IF NO INTERRUPT OCCURS (BECAUSE OF A BAD DZV11) THE PROGRAM ASSUMES VECTOR ADDRESS 300 AND THE PROBLEM SHOULD BE FIXED IN THE DIAGNOSTIC. ONCE THE PROBLEM IS FIXED, THE PROGRAM SHOULD BE SETUP AGAIN TO SET THE CORRECT VECTOR. IF AN INTERRUPT OCCURRED, THE ADDRESS TO WHICH THE DZV11 INTERRUPTED TO IS PICKED UP AND REPORTED AS THE VECTOR. NOTE: IF THE VECTOR REPORTED IS NOT THE VECTOR SET UP BY YOU, THERE IS A PROBLEM AND AUTO SIZING SHOULD NOT BE DONE.

8.5.3 PARAMETER ASSUMPTIONS.

SINCE TOO MUCH HARDWARE WOULD NEED TO BE TURNED ON TO SIZE THE REST OF THE PARAMETERS, THE PROGRAM MUST ASSUME THE REMAINING VARIATIONS. THE RESULT IF NOT TO YOUR SPECIFIC CONFIGURATION MAY BE ALTERED BY HAND. IN THIS WAY 95% OF THE PARAMETER SETUP WAS DONE BY THE PROGRAM AND 5% BY YOU.

THEREFORE:

- 1) ALL FOUR LINES ARE ASSUMED TO BE TESTED.
- 2) DEFAULT BAUD RATE IS SET TO 17 (19.2 KBAUD).
- 3) MODE OF OPERATION IS "INTERNAL MODE".

FOR ALL PARAMETER ADJUSTMENTS PLEASE REFER TO SECTION 8.4 FOR GREATER DETAIL.

9.0 RUNNING THE DZV11 DIAGNOSTIC UNDER APT

9.1.1 THE APT INTERFACE

THE DZV DIAGNOSTICS HAVE BEEN DESIGNED TO BE COMPATIBLE WITH THE APT (AUTOMATED PRODUCT TEST) SYSTEM. THE DZV LOGIC TEST DIAGNOSTICS (DVDZA, AND DVDZB) CAN BE RUN AS STANDALONE DIAGNOSTICS OR IN EITHER OF THE APT MODES. DVDZC, HOWEVER IS DESIGNED AS A STANDALONE DIAGNOSTIC ONLY AND REQUIRES DIRECT OPERATOR PARTICIPATION.

9.1.2 SETTING UP THE DIAGNOSTIC USING APT

THE DIAGNOSTIC USES SEVERAL VARIABLES IN THE REGION SUBTITLED " APT MAILBOX-ETABLE". THESE VARIABLES ARE:

#SMREG -(1142)	USED AS THE SOFTWARE SWITCH REGISTER WHILE RUNNING UNDER APT.
#VECT1 -(1170)	USED TO SPECIFY THE FIRST VECTOR ADDRESS
#BASE -(1174)	USED TO INDICATE BOTTOM ADDRESS OF DZV11 UNDER TEST
#DEVH -(1176)	A BIT MAP REPRESENTING WHICH DZV11'S WILL BE TESTED
#CDW1 -(1200)	USED TO INDICATE WHICH LINES TO RUN ON ALL DZV11'S
#CDW2 -(1202)	USED TO INDICATE THE DEFAULT TEST MODE. SET TO 0 FOR INTERNAL TESTING, 200 FOR EXTERNAL LOOP BACK (H325 INSTALLED), OR SET TO 100000 FOR STAGGERED LOOP BACK TESTING (H329 INSTALLED).
#DDW0 -(1204)	EACH OF THE #DDW WORDS DESCRIBES THE PARAMETERS (LPR) FOR A PARTICULAR DZV11, GOING UP TO 16 DZV11'S

9.1.3 RUNNING UNDER APT

ALL OF THE VARIABLES MENTIONED IN SECTION 9.1.2 SHOULD BE SET UP PRIOR TO RUNNING THE DIAGNOSTIC UNDER APT.

NOTE

BE SURE #BASE POINTS TO THE FIRST DZV11 BEFORE RUNNING

BASED ON THESE VALUES, THE DIAGNOSTIC WILL SET UP THE STATUS TABLE. THE USER IS THEN FREE TO MONITOR UNDER APT AS NORMAL.

CVDZBO
CVDZBO P11 21-AUG-84 08:28

10.0

PROGRAM DESCRIPTION.

THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC PACKAGE (MAINDEC-11-DZBAC-C3), JAN 19, 1977.

46 INITIAL ADDRESS OF THE STACK POINTER *** 1120 ***

51 MISCELLANEOUS DEFINITIONS

63 GENERAL PURPOSE REGISTER DEFINITIONS

75 PRIORITY LEVEL DEFINITIONS

85 "SWITCH REGISTER" SWITCH DEFINITIONS

113 DATA BIT DEFINITIONS (BIT00 TO BIT15)

141 BASIC "CPU" TRAP VECTOR ADDRESSES

358 BITS 15-11=CPU TYPE
11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
11/70=06,PDQ=07,Q=10
BIT 10-REAL TIME CLOCK
BIT 9-FLOATING POINT PROCESSOR
BIT 8-MEMORY MANAGEMENT

366 MEM.TYPE BYTE -- (HIGH BYTE)
900 NSEC CORE=001
300 NSEC BIPOLAR=002
300 NSEC MOS=003

371 MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF "TYPE" A80

410 THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS USED IN THE PROGRAM.

462 THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR. THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN LOCATION #ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
NOTE1: IF #ITEMB IS 0 THE ONLY PERTINENT DATA IS (#ERRPC).
NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

468 EM ::POINTS TO THE ERROR MESSAGE
DH ::POINTS TO THE DATA HEADER
DT ::POINTS TO THE DATA
DF ::POINTS TO THE DATA FORMAT

CVDZ80
CVDZ80.P11 21-AUG-84 08:28

11GPA MACY11 30A(1052) 21-AUG-84 08:31 PAGE 17

1010 INCREMENT THE PASS NUMBER (#PASS)
IF THERES A MONITOR GO TO IT
IF THERE ISN'T JUMP TO CYCLE

1072 THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
AND LOAD THE TEST NUMBER(#TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
AND LOAD THE ERROR FLAG (#ERFLG) INTO DISPLAY<15:08>
THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
SW14=1 LOOP ON TEST
SW11=1 INHIBIT ITERATIONS
CALL
SCOPE ;:SCOPE=IOT

1147 ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
NOTE1: #NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
NOTE2: #FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
NOTE3: #FILLC CONTAINS THE CHARACTER TO FILL AFTER.

CALL:
1) USING A TRAP INSTRUCTION
TYPE ,MESADR ;:MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
OR
TYPE
MESADR

1931 ROUTINE USED TO SET UP THE DIAGNOSTIC VIA APT.
IF BIT7 IN THE ENVIRONMENT MODE (#ENVM) BYTE IS SET,
THE PROGRAM WILL LOAD ITS PARAMETERS FROM THE ETABLE.

1963 ROUTINE USED TO "AUTO SIZE" THE DZV11
CSR AND VECTOR.
NOTE: THE CSR MAY BE ANY WHERE IN THE FLOATING
ADDRESS RANGE (160000;167770)
AND THE VECTOR MAY BE ANY WHERE IN THE
FLOATING VECTOR RANGE (300;770)

2071 ***** TEST 1 *****
THIS TEST VERIFIES OVERRUN AND SILO ALARM
ONE LINE AT A TIME - BASED UPON VALID LINES
AS EACH OF THE FIRST 16 CHARS ARE SENT, SILO ALARM IS
TESTED TO BE CLEARED. ON THE 16TH CHAR THE PROGRAM THEN
EXPECTS SILO ALARM TO SET, THEN THE ENTIRE
SILO IS FILLED AND AN OVERRUN IS EXPECTED ON THE 65TH
CHAR PULLED OUT OF THE SILO.
ERROR PRINTOUTS WILL REPORT TRANSMITTING LINE NO.
USING SWITCH NINE FOR THIS TEST SENDS 20. CHARACTERS
ON DZV LINE PREVIOUSLY SELECTED CONTINUOUSLY WHILE SW09=1.
USED TO SCOPE SILO ALARM PULSES, ETC.

- 2192 ***** TEST 2 *****
THIS TEST THAT "SILO ENABLE" WILL INHIBIT
RECEIVER INTERRUPTS AND THAT ON THE
16TH CHAR THAT "SILO ALARM" WILL CAUSE AN
INTERRUPT WITH "RIE" SET.
THIS WILL DO ALL SELECTED LINES ONE AT A TIME.
ERROR PRINTOUTS WILL REPORT TRANSMITTING LINE NO.

- 2264 ***** TEST 3 *****
THIS TEST RUNS ALL LINES FULL BORE
BASED UPON QUALIFIED LINES
..THIS IS AN INTERRUPT TEST ON THE RECEIVER AND
TRANSMITTER

- 2397 ***** TEST 4 *****
DZV11 RELATIVE TIMING TEST.
EACH SELECTED LINE WILL IN TURN RUN 16. CHARS
AT ALL BAUD RATES AND THEN THE HIGHEST BAUD
WITH ALL CHAR LENGTHS. EACH NEW PARAMETER SHOULD
DECREASE IN TIME FROM THE PREVIOUS PARAMETERS SELECTED.
THE TIME IS CHECKED AGAINST THE LAST PARAMETER USED
AND A LOWER TIME IS EXPECTED ON THE CURRENT PARAMETER.
PARAMETERS ARE:
EIGHT BITS/PER/CHAR - TWO STOP BITS AT
50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000
2400, 3600, 4800, 7200, 9600 BAUD.
19.2 K BAUD - TWO STOP BITS AT
SEVEN, SIX, FIVE BITS/PER/CHAR.
AFTER EACH LINE HAS FINISHED ALL THE ABOVE PARAMETERS
THE NEXT SELECTED LINE IS THEN TESTED.
WHEN RUNNING UNDER THE APT MANUFACTURING SYSTEM
THIS TEST IS ONLY RUN THE FIRST PASS

- 2491 ***** TEST 5 *****
THE MAIN FUNCTION OF THIS TEST IS TO VERIFY
THAT "PE" (PARITY ERROR) CAN BE FLAGGED BY
THE UARTS. THIS TEST WILL NOT BE DONE UNLESS
YOU ARE IN "STAGGERED" MODE.
40(8) CHARS ARE USED FOR THIS TEST.
ALL SELECTED LINES WILL BE ENABLED AT THE SAME TIME.
THIS TEST FIRST CHECKS EVEN PARITY FOR ODD LINES AND
ODD PARITY FOR EVEN LINES, THEN IT CHECKS THE REVERSE.

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26
CVDZBD.P11 21-AUG-84 08:28 BASIC DEFINITIONS

(2)	177776	PS=	177776	::	PROCESSOR STATUS WORD
(2)	177776	PSW=	PS		
(2)	177774	STKLMT=	177774	::	STACK LIMIT REGISTER
(2)	177772	PIRQ=	177772	::	PROGRAM INTERRUPT REQUEST REGISTER
(2)	177570	DSMR=	177570	::	HARDWARE SWITCH REGISTER
(2)	177570	DDISP=	177570	::	HARDWARE DISPLAY REGISTER

;; GENERAL PURPOSE REGISTER DEFINITIONS

(2)	000000	R0=	#0	::	GENERAL REGISTER
(2)	000001	R1=	#1	::	GENERAL REGISTER
(2)	000002	R2=	#2	::	GENERAL REGISTER
(2)	000003	R3=	#3	::	GENERAL REGISTER
(2)	000004	R4=	#4	::	GENERAL REGISTER
(2)	000005	R5=	#5	::	GENERAL REGISTER
(2)	000006	R6=	#6	::	GENERAL REGISTER
(2)	000007	R7=	#7	::	GENERAL REGISTER
(2)	000006	SP=	#6	::	STACK POINTER
(2)	000007	PC=	#7	::	PROGRAM COUNTER

;; PRIORITY LEVEL DEFINITIONS

(2)	000300	PR0=	0	::	PRIORITY LEVEL 0
(2)	000040	PR1=	40	::	PRIORITY LEVEL 1
(2)	000100	PR2=	100	::	PRIORITY LEVEL 2
(2)	000140	PR3=	140	::	PRIORITY LEVEL 3
(2)	000200	PR4=	200	::	PRIORITY LEVEL 4
(2)	000240	PR5=	240	::	PRIORITY LEVEL 5
(2)	000300	PR6=	300	::	PRIORITY LEVEL 6
(2)	000340	PR7=	340	::	PRIORITY LEVEL 7

;; "SWITCH REGISTER" SWITCH DEFINITIONS

(2)	100000	SW15=	100000		
(2)	040000	SW14=	40000		
(2)	020000	SW13=	20000		
(2)	010000	SW12=	10000		
(2)	004000	SW11=	4000		
(2)	002000	SW10=	2000		
(2)	001000	SW09=	1000		
(2)	000400	SW08=	400		
(2)	000200	SW07=	200		
(2)	000100	SW06=	100		
(2)	000040	SW05=	40		
(2)	000020	SW04=	20		
(2)	000010	SW03=	10		
(2)	000004	SW02=	4		
(2)	000002	SW01=	2		
(2)	000001	SW00=	1		
(2)	001000	SW9=	SW09		
(2)	000400	SW8=	SW08		
(2)	000200	SW7=	SW07		
(2)	000100	SW6=	SW06		
(2)	000040	SW5=	SW05		
(2)	000020	SW4=	SW04		
(2)	000010	SW3=	SW03		
(2)	000004	SW2=	SW02		
(2)	000002	SW1=	SW01		
(2)	000001	SW0=	SW00		

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-1
 CVDZBD.P11 21-AUG-84 08:28 BASIC DEFINITIONS

```

(2)
(2)          100000      ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
(2)          040000      BIT15= 100000
(2)          020000      BIT14= 40000
(2)          010000      BIT13= 20000
(2)          004000      BIT12= 10000
(2)          002000      BIT11= 4000
(2)          001000      BIT10= 2000
(2)          000400      BIT09= 1000
(2)          000200      BIT08= 400
(2)          000100      BIT07= 200
(2)          000040      BIT06= 100
(2)          000020      BIT05= 40
(2)          000010      BIT04= 20
(2)          000004      BIT03= 10
(2)          000002      BIT02= 4
(2)          000001      BIT01= 2
(2)          000001      BIT00= 1
(2)          001000      BIT9=  BIT09
(2)          000400      BIT8=  BIT08
(2)          000200      BIT7=  BIT07
(2)          000100      BIT6=  BIT06
(2)          000040      BIT5=  BIT05
(2)          000020      BIT4=  BIT04
(2)          000010      BIT3=  BIT03
(2)          000004      BIT2=  BIT02
(2)          000002      BIT1=  BIT01
(2)          000001      BIT0=  BIT00

(2)          ;*BASIC "CPU" TRAP VECTOR ADDRESSES
(2)          000004      ERRVEC= 4          ;:TIME OUT AND OTHER ERRORS
(2)          000010      RESVEC= 10         ;:RESERVED AND ILLEGAL INSTRUCTIONS
(2)          000014      TBITVEC=14         ;:"T" BIT
(2)          000014      TRTVEC= 14         ;:TRACE TRAP
(2)          000014      BPTVEC= 14         ;:BREAKPOINT TRAP (BPT)
(2)          000020      IOTVEC= 20         ;:INPUT/OUTPUT TRAP (IOT) **SCOPE**
(2)          000024      PWRVEC= 24         ;:POWER FAIL
(2)          000030      EMTVEC= 30         ;:EMULATOR TRAP (EMT) **ERROR**
(2)          000034      TRAPVEC=34         ;:"TRAP" TRAP
(2)          000060      TKVEC= 60          ;:TTY KEYBOARD VECTOR
(2)          000064      TPVEC= 64          ;:TTY PRINTER VECTOR
(2)          000240      PIRQVEC=240        ;:PROGRAM INTERRUPT REQUEST VECTOR

(1)
(1)          ;INSTRUCTION DEFINITIONS
(1)          ;-----
(1)          005746      PUSH1SP=5746      ;DECREMENT PROCESSOR STACK 1 WORD
(1)          005726      POP1SP=5726       ;INCREMENT PROCESSOR STACK 1 WORD
(1)          010046      PUSHRO=10046       ;SAVE R0 ON STACK
(1)          012600      POPRO=12600        ;RESTORE R0 FROM STACK
(1)          024646      PUSH2SP=24646     ;DECREMENT STACK TWICE
(1)          022626      POP2SP=22626     ;INCREMENT STACK TWICE
(1)          000200      MASK=BIT7         ;SET INTERRUPT MASK (INHIBIT FURTHER INTERRUPTS)
(1)          000000      CLEAR=0           ;ALLOW INTERRUPTS (CLEAR PROCESSOR STATUS)
(1)

```


CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-4
CVDZBD.P11 21-AUG-84 08:28

GENERAL DEFINITIONS AND EQUIVALENCES

(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)

TABLE OF LOOP AROUND FUNCTIONS (H325)

I	↑
V	↑
REC	TRANS
DATA	DATA

I	↑
V	↑
CO	RTS

I	↑
V	↑
RING	DTR

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-5
 CVDZBD.P11 21 AUG-84 08:28 TRAPCATCHER FOR UNEXPECTED INTERRUPTS

```

(1) ;:*****
(1) ;-----
(1) ; TRAPCATCHER FOR ILLEGAL INTERRUPTS
(1) ; THE STANDARD "TRAP CATCHER" IS PLACED
(1) ; BETWEEN ADDRESS 0 TO ADDRESS 776.
(1) ; IT LOOKS LIKE "PC+2 HALT".
(1) ;-----
(1) ;:*****
(1) ;
(1) ;=0
(1) 000000 ;MUST NOT SET UP EMT TRAP CATCHER SINCE IT IS USED FOR UFD MONITOR LINKAGE
(1) ;=34 ;SKIP OVER EMT TRAP
(1) 000034 ;STANDARD INTERRUPT VECTORS
(1) ;-----
(1) ;=20
(1) 000020 ;SCOPE
(1) 000020 004464 ;SCOPE LOOP HANDLER
(1) 000022 000200 ;HANDLE AT PRIORITY 7
(1) 000024 007566 ;POWER FAIL HANDLER
(1) 000026 000340 ;SERVICE AT PRIORITY LEVEL 7
(1) ;=34
(1) 000034 ;TRPSRV
(1) 000036 000340 ;GENERAL HANDLER DISPATCH SERVICE
(1) ;SERVICE AT PRIORITY LEVEL 7
(2) ;SBTTL ACT11 HOOKS
(2) ;
(3) ;:*****
(2) ;HOOKS REQUIRED BY ACT11
(2) 000040 ;SVPC= ;SAVE PC
(2) 000046 ;=46
(2) 000046 004420 ;ENDAD ;1)SET LOC.46 TO ADDRESS OF $ENDAD IN $EOP
(2) 000052 ;=52
(2) 000052 000000 ;WORD 0 ;2)SET LOC.52 TO ZERO
(2) 000040 ;=$SVPC ; RESTORE PC
(1) ;=174
(1) 000174 000000 DISPREG:0 ;SOFTWARE DISPLAY REGISTER FOR SWITCHLESS 11S
(1) 000176 000000 SMREG: 0 ;SOFTWARE SWITCH REGISTER FOR SWITCHLESS 11S
(1) 000200 000200 ;=200
(1) 000200 000137 002116 JMP .START ;GO TO START OF PROGRAM
(2) ;=1000
(2) 001000 005200 053103 055104 ;HTITLE: .ASCIZ <200><12>/CVDZBD/<200>/FOUR LINE ASYNC MUX TESTS, PART 2 OF 2/<200>
(2)
    
```


CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-8
 CVDZBO.P11 21-AUG-84 08:28 COMMON TAGS

```

(3)          .SBTTL COMMON TAGS
(3)
(4)          ;*****
(3)          ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
(3)          ;*USED IN THE PROGRAM.
(3)
(3)          $CHTAG:          ;:START OF COMMON TAGS
(3) 001244          000000          .WORD          0          ;:CONTAINS THE TEST NUMBER
(3) 001244          000          .BYTE          0          ;:CONTAINS ERROR FLAG
(3) 001246          000          .BYTE          0          ;:CONTAINS SUBTEST ITERATION COUNT
(3) 001247          000          .BYTE          0          ;:CONTAINS SCOPE LOOP ADDRESS
(3) 001250          000000          .WORD          0          ;:CONTAINS SCOPE RETURN FOR ERRORS
(3) 001252          000000          .WORD          0          ;:CONTAINS TOTAL ERRORS DETECTED
(3) 001254          000000          .WORD          0          ;:CONTAINS ITEM CONTROL BYTE
(3) 001256          000000          .WORD          0          ;:CONTAINS MAX. ERRORS PER TEST
(3) 001260          000          .BYTE          0          ;:CONTAINS PC OF LAST ERROR INSTRUCTION
(3) 001261          001          .BYTE          1          ;:CONTAINS ADDRESS OF 'GOOD' DATA
(3) 001262          000000          .WORD          0          ;:CONTAINS ADDRESS OF 'BAD' DATA
(3) 001264          000000          .WORD          0          ;:CONTAINS 'GOOD' DATA
(3) 001266          000000          .WORD          0          ;:CONTAINS 'BAD' DATA
(3) 001270          000000          .WORD          0          ;:RESERVED--NOT TO BE USED
(3) 001272          000000          .WORD          0
(3) 001274          000000          .WORD          0
(3) 001276          000000          .WORD          0
(3) 001300          000          .BYTE          0          ;:AUTOMATIC MODE INDICATOR
(3) 001301          000          .BYTE          0          ;:INTERRUPT MODE INDICATOR
(3) 001302          000000          .WORD          0
(3) 001304          177570          .WORD          DSWR          ;:ADDRESS OF SWITCH REGISTER
(3) 001306          177570          .WORD          DDISP          ;:ADDRESS OF DISPLAY REGISTER
(3) 001310          177560          .WORD          0          ;:TTY KBD STATUS
(3) 001312          177562          .WORD          0          ;:TTY KBD BUFFER
(3) 001314          177564          .WORD          0          ;:TTY PRINTER STATUS REG. ADDRESS
(3) 001316          177566          .WORD          0          ;:TTY PRINTER BUFFER REG. ADDRESS
(3) 001320          000          .BYTE          0          ;:CONTAINS NULL CHARACTER FOR FILLS
(3) 001321          002          .BYTE          2          ;:CONTAINS # OF FILLER CHARACTERS REQUIRED
(3) 001322          012          .BYTE          12          ;:INSERT FILL CHARS. AFTER A "LINE FEED"
(3) 001323          000          .BYTE          0          ;:"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
(3) 001324          000000          .WORD          0          ;:CONTAINS THE ADDRESS FROM
(3)          ;:WHICH (#REGO) WAS OBTAINED
(5) 001326          000000          .WORD          0          ;:CONTAINS ((#REGAD)+0)
(5) 001330          000000          .WORD          0          ;:CONTAINS ((#REGAD)+2)
(5) 001332          000000          .WORD          0          ;:CONTAINS ((#REGAD)+4)
(5) 001334          000000          .WORD          0          ;:CONTAINS ((#REGAD)+6)
(5) 001336          000000          .WORD          0          ;:CONTAINS ((#REGAD)+10)
(5) 001340          000000          .WORD          0          ;:CONTAINS ((#REGAD)+12)
(5) 001342          000000          .WORD          0          ;:USER DEFINED
(5) 001344          000000          .WORD          0          ;:USER DEFINED
(5) 001346          000000          .WORD          0          ;:USER DEFINED
(5) 001350          000000          .WORD          0          ;:USER DEFINED
(5) 001352          000000          .WORD          0          ;:USER DEFINED
(3) 001354          000000          .WORD          0          ;:MAX. NUMBER OF ITERATIONS
(3) 001356          077          .ASCII          /?/          ;:QUESTION MARK
(3) 001357          015          .ASCII          <15>          ;:CARRIAGE RETURN
(3) 001360          000012          .ASCII          <12>          ;:LINE FEED
    
```

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-9
 CVDZBO P11 21-AUG-84 08:28 ERROR POINTER TABLE

```

(3) .SBTTL ERROR POINTER TABLE
(3)
(3) ;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
(3) ;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
(3) ;*LOCATION #ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
(3) ;*NOTE1: IF #ITEMB IS 0 THE ONLY PERTINENT DATA IS (#ERRPC).
(3) ;*NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:
(3)
(3) ;* EM ;:POINTS TO THE ERROR MESSAGE
(3) ;* DM ;:POINTS TO THE DATA HEADER
(3) ;* DT ;:POINTS TO THE DATA
(3) ;* DF ;:POINTS TO THE DATA FORMAT
(3)
(3) #ERRTB:
(3) ;PROGRAM CONTROL PARAMETERS
(3) ;-----
(3)
(3) 001362 000000 NEXT: 0 ;ADDRESS OF NEXT TEST TO BE EXECUTED
(3) 001364 000000 LOCK: 0 ;ADDRESS FOR LOCK ON CURRENT TEST,TIGHT LOOP
(3)
(3) ;PROGRAM VARIABLES
(3) ;-----
(3)
(3) 001366 000017 LINE: 17 ;DEFAULT ALL FOUR LINES RUNNING
(3) 001370 017470 PAR: 17470 ;PARAMETERS: 8 BITS/CHAR,2 STOP BITS,19200 BAUD,NO PARITY
(3) 001372 000000 MODE: 0 ;DEFAULT MAINTENANCE MODE
(3) 001374 000000 SAVLIN: 0 ;LINE NUMBER
(3) 001376 000000 XPTLIN: 0 ;TRANSMISSION LINE NUMBER
(3) 001400 000000 XPTCNT: 0 ;COUNT OF WORDS IN A TRANSMISSION PATTERN
(3) 001402 000000 REGIST: 0 ;DEVICE ADDRESS STORAGE LOCATION
(3) 001404 000000 SAVPC: 0 ;PROGRAM COUNTER STORAGE
(3) 001406 000001 DZVACTV: .BLKW 1 ;*DZV11'S SELECTED ACTIVE.
(3) 001410 000001 SAVACTV: .B_LKW 1 ;*A BIT MAP OF DZV11'S IN THE SYSTEM
(3) 001412 000001 RUN: 1 ;*POINTER ONE PAST RUNNING DEVICE.
(3) 001414 000001 DZVNUM: .BLKB 1 ;*OCTAL NUMBER OF DZV11'S IN THE SYSTEM.
(3) 001415 001 SAVNUM: .BYTE 1 ;*WORKABLE NUMBER.
(3) 001416 000001 SAVNO: .BLKB 1 ;*OCTAL NUMBER OF DZV11'S BEING TESTED
(3) 001420 001420 .EVEN
(3) 001420 001500 ACTIVE: DZV.MAP ;TABLE POINTER.

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-10
 CVDZB0.P11 21-AUG-84 08:28 ERROR POINTER TABLE

```

(2)
(2)
(2)
(2)
(2) 001422 000
(2) 001423 000
(2) 001424 000
(2) 001425 000
(2)
(2)
(2) 001426 000000
(2) 001430 000000
(2) 001432 000000
(2) 001434 000000
(2) 001436 000000
(2) 001440 000000
(2) 001442 000000
(2) 001444 000000
(2) 001446
(2)
(2)
(3)
(2)
(3)
(2) 001446
(2) 000024
(2) 000024 000200
(2) 000044
(2) 000044 001446
(2) 001446
(3)
(2)
(2)
(2)
(2) 001446
(2) 001446 000000
(2) 001450 001120
(2) 001452 000132
(2) 001454 000137
(2) 001456 000000
(2) 001460 000052
(1)
(1)
(1)
(1) 001500
(1) 001500
(3)
(3) 001500 000001
(3) 001502 000001
(3) 001504 000001
(3) 001506 000001
(3) 001510 000001
(3)
(3) 001512 000001
(3) 001514 000001
(3) 001516 000001

```

```

;PROGRAM CONTROL FLAGS
-----
INIFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG
HDRFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG FOR HEADER MAP
MNTFLG: .BYTE 0 ;MAINTENANCE BIT SET FLAG
DNMFLG: .BYTE 0 ;TRANSMISSION COMPLETION FLAG
.EVEN
;DATA VARIABLES
TD0: .WORD 0
TD1: .WORD 0
TD2: .WORD 0
TD3: .WORD 0
TR0: .WORD 0
TR1: .WORD 0
TR2: .WORD 0
TR3: .WORD 0
STOP:
.SBTTL APT PARAMETER BLOCK

;*****
;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
;*****
.X= ;SAVE CURRENT LOCATION
=24 ;SET POWER FAIL TO POINT TO START OF PROGRAM
200 ;FOR APT START UP
=44 ;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR ;POINT TO APT HEADER BLOCK
=.X ;RESET LOCATION COUNTER
;*****
;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-POP11 DIAGNOSTIC
;INTERFACE SPEC.

$APTHD:
$HIBTS: .WORD 0 ;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBAADR: .WORD $MAIL ;ADDRESS OF APT MAILBOX (BITS 0-15)
$TSTM: .WORD 90. ;RUN TIM OF LONGEST TEST
$PASTM: .WORD 95. ;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD 0. ;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
.WORD $ETEND-$MAIL/2 ;LENGTH MAILBOX-ETABLE(WORDS)
;DZV11 STATUS TABLE AND ADDRESS ASSIGNMENTS
-----

.=1500
DZV.MAP:
DZCR0: .BLKW 1 ;CONTROL STATUS REGISTER FOR DZV11 NUMBER 0
DZVC0: .BLKW 1 ;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 0
LINE0: .BLKW 1 ;ALL LINES SELECTED
PAR0: .BLKW 1 ;PARAMETERS
MANT0: .BLKW 1 ;MAINTENANCE MODE FOR THIS DEVICE

DZCR1: .BLKW 1 ;CONTROL STATUS REGISTER FOR DZV11 NUMBER 1
DZVC1: .BLKW 1 ;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 1
LINE1: .BLKW 1 ;ALL LINES SELECTED

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-11
 CVDZBO.P11 21-AUG-84 08:28 APT PARAMETER BLOCK

(3)	001520	000001	PAR1:	.BLKW	1	;PARAMETERS
(3)	001522	000001	MANT1:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001524	000001	DZCR2:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 2
(3)	001526	000001	DZVC2:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 2
(3)	001530	000001	LINE2:	.BLKW	1	;ALL LINES SELECTED
(3)	001532	000001	PAR2:	.BLKW	1	;PARAMETERS
(3)	001534	000001	MANT2:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001536	000001	DZCR3:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 3
(3)	001540	000001	DZVC3:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 3
(3)	001542	000001	LINE3:	.BLKW	1	;ALL LINES SELECTED
(3)	001544	000001	PAR3:	.BLKW	1	;PARAMETERS
(3)	001546	000001	MANT3:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001550	000001	DZCR4:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 4
(3)	001552	000001	DZVC4:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 4
(3)	001554	000001	LINE4:	.BLKW	1	;ALL LINES SELECTED
(3)	001556	000001	PAR4:	.BLKW	1	;PARAMETERS
(3)	001560	000001	MANT4:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001562	000001	DZCR5:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 5
(3)	001564	000001	DZVC5:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 5
(3)	001566	000001	LINE5:	.BLKW	1	;ALL LINES SELECTED
(3)	001570	000001	PAR5:	.BLKW	1	;PARAMETERS
(3)	001572	000001	MANT5:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001574	000001	DZCR6:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 6
(3)	001576	000001	DZVC6:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 6
(3)	001600	000001	LINE6:	.BLKW	1	;ALL LINES SELECTED
(3)	001602	000001	PAR6:	.BLKW	1	;PARAMETERS
(3)	001604	000001	MANT6:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001606	000001	DZCR7:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 7
(3)	001610	000001	DZVC7:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 7
(3)	001612	000001	LINE7:	.BLKW	1	;ALL LINES SELECTED
(3)	001614	000001	PAR7:	.BLKW	1	;PARAMETERS
(3)	001616	000001	MANT7:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001620	000001	DZCR10:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 10
(3)	001622	000001	DZVC10:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 10
(3)	001624	000001	LINE10:	.BLKW	1	;ALL LINES SELECTED
(3)	001626	000001	PAR10:	.BLKW	1	;PARAMETERS
(3)	001630	000001	MANT10:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001632	000001	DZCR11:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 11
(3)	001634	000001	DZVC11:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 11
(3)	001636	000001	LINE11:	.BLKW	1	;ALL LINES SELECTED
(3)	001640	000001	PAR11:	.BLKW	1	;PARAMETERS
(3)	001642	000001	MANT11:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)						
(3)	001644	000001	DZCR12:	.BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 12
(3)	001646	000001	DZVC12:	.BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 12
(3)	001650	000001	LINE12:	.BLKW	1	;ALL LINES SELECTED
(3)	001652	000001	PAR12:	.BLKW	1	;PARAMETERS
(3)	001654	000001	MANT12:	.BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE

CVDZB D MACY11 30A(1052) 21 AUG-84 08:31 PAGE 26-12
 CVDZBD.P11 21-AUG-84 08:28 APT PARAMETER BLOCK

(3)					
(3)	001656	000001	DZCR13: .BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 13
(3)	001660	000001	DZVC13: .BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 13
(3)	001662	000001	LINE13: .BLKW	1	;ALL LINES SELECTED
(3)	001664	000001	PAR13: .BLKW	1	;PARAMETERS
(3)	001666	0000C1	MANT13: .BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)					
(3)	001670	000001	DZCR14: .BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 14
(3)	001672	000001	DZVC14: .BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 14
(3)	001674	000001	LINE14: .BLKW	1	;ALL LINES SELECTED
(3)	001676	000001	PAR14: .BLKW	1	;PARAMETERS
(3)	001700	000001	MANT14: .BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)					
(3)	001702	000001	DZCR15: .BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 15
(3)	001704	000001	DZVC15: .BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 15
(3)	001706	000001	LINE15: .BLKW	1	;ALL LINES SELECTED
(3)	001710	000001	PAR15: .BLKW	1	;PARAMETERS
(3)	001712	000001	MANT15: .BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)					
(3)	001714	000001	DZCR16: .BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 16
(3)	001716	000001	DZVC16: .BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 16
(3)	001720	000001	LINE16: .BLKW	1	;ALL LINES SELECTED
(3)	001722	000001	PAR16: .BLKW	1	;PARAMETERS
(3)	001724	000001	MANT16: .BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(3)					
(3)	001726	000001	DZCR17: .BLKW	1	;CONTROL STATUS REGISTER FOR DZV11 NUMBER 17
(3)	001730	000001	DZVC17: .BLKW	1	;RECEIVER AND BASE VECTOR FOR DZV11 NUMBER 17
(3)	001732	000001	LINE17: .BLKW	1	;ALL LINES SELECTED
(3)	001734	000001	PAR17: .BLKW	1	;PARAMETERS
(3)	001736	000001	MANT17: .BLKW	1	;MAINTENANCE MODE FOR THIS DEVICE
(1)					
(1)	001740	177777	DZV.END:	177777	

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-14
 CVDZBO.P11 21-AUG-84 08:28 APT PARAMETER BLOCK

```

(1)                                     ;DZV11 VECTOR AND REGISTER INDIRECT POINTERS
(1)                                     ;WORKING AREA
(1)
(1) 002010 160040 DZVCSR: 160040 ;R/W
(1) 002012 160041 HDZVCSR:160041 ;R/W
(1) 002014 160042 DZVRBUF:160042 ;READ ONLY
(1) 002016 160043 HDZVRBUF:160043 ;READ ONLY
(1) 002020 160042 DZVLPR: 160042 ;WRITE ONLY
(1) 002022 160043 HDZVLPR:160043 ;WRITE ONLY
(1) 002024 160044 DZVTCR: 160044 ;R/W
(1) 002026 160045 HDZVTCR:160045 ;R/W
(1) 002030 160046 DZVMSR: 160046 ;READ ONLY
(1) 002032 160047 HDZVMSR:160047 ;READ ONLY
(1) 002034 160046 DZVTDR: 160046 ;WRITE ONLY
(1) 002036 160047 HDZVTDR:160047 ;WRITE ONLY
(1)
(1)                                     ;DEFAULT DZV VECTORS
(1)
(1) 002040 000300 DZVRIV: 300 ;REC INTR VECTOR
(1) 002042 000302 DZVRIS: 302 ;REC INTR STATUS
(1) 002044 000304 DZVTIV: 304 ;XMIT INTR VECTOR
(1) 002046 000306 DZVTIS: 306 ;XMIT INTR STATUS
(1)
(1)

```

(VDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-15
(VDZBO P11 21-AUG-84 08:28 APT PARAMETER BLOCK

			TIME TABLE FOR RELATIVE TIMING TESTS	

(1)			TMTBL:	
(1)	002050	000000	T50:	0
(1)	002052	000000	T75:	0
(1)	002054	000000	T110:	0
(1)	002056	000000	T134:	0
(1)	002060	000000	T150:	0
(1)	002062	000000	T300:	0
(1)	002064	000000	T600:	0
(1)	002066	000000	T1200:	0
(1)	002070	000000	T1800:	0
(1)	002072	000000	T2000:	0
(1)	002074	000000	T2400:	0
(1)	002076	000000	T3600:	0
(1)	002100	000000	T4800:	0
(1)	002102	000000	T7200:	0
(1)	002104	000000	T9600:	0
(1)	002106	000000	TEIGHT:	0
(1)	002110	000000	TSEVEN:	0
(1)	002112	000000	TSIX:	0
(1)	002114	000000	TFIVE:	0

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-16
CVDZB0.P11 21-AUG-84 08:28 PROGRAM INITIALIZATION AND START UP.

```

(1)                                     ;PROGRAM INITIALIZATION
(1)                                     ;LOCK OUT INTERRUPTS
(1)                                     ;SET UP PROCESSOR STACK
(1)                                     ;SET UP POWER FAIL VECTOR
(1)                                     ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
(1)                                     ;TYPE TITLE MESSAGE
(1)                                     UFDSET = 1
(1)
(1) 002116 032737 000040 000052 .START: BIT    @BITS, @#52    ;ARE WE UNDER UFD ?
(1) 002124 001403                BEQ      1#          ;NO, THEN SKIP THE INSTRUCTIONS
(1) 002126 104042                EMT     42          ;GET DSRERR ADDRESS
(1) 002130 005060 000042                CLR     42(R0)     ;INITIALIZE IT TO NO ERROR
(1) 002134
(2)                                     1#:
(2)                                     ;;LCP/ORION ROUTINE TO SAVE EMTULATOR AND PRIORITY
(2) 002134 013737 000030 002152 EHTSAV: MOV     30, SAV30    ;;SAVE EMULATOR ADDRESS
(2) 002142 013737 000032 002154      MOV     32, SAV32    ;;SAVE EMULATOR PRIORITY LEVEL
(2) 002150 000402                BR      VMKOR      ;;GET AROUND TAG AREA
(2) 002152 000000      SAV30: .WORD 0    ;;PUT EMULATOR INFO HERE
(2) 002154 000000      SAV32: .WORD 0    ;;PUT PRIORITY LOCATION HERE
(2) 002156      VMKOR:
(2)                                     ;*****
(1) 002156 012737 006664 000030      MOV     @ERROR, @#30    ;SET UP EMT VECTOR TO ERROR ROUTINE
(1) 002164 012737 000340 000032      MOV     @340, @#32
(1) 002172 012706 001120                MOV     @STACK, SP    ;SET UP STACK
(1) 002176 106427 000200      MTPS   @MASK        ;LOCK OUT INTERRUPTS
(1) 002202 012737 007566 000024      MOV     @PWRDN, @#24  ;SET UP POWER FAIL VECTOR
(1) 002210 005037 001126      CLR     @PASS        ;CLEAR PASS COUNT
(1) 002214 105037 001247      CLRB   @ERFLG       ;CLEAR ERROR FLAG
(1) 002220 012737 001500 001420      MOV     @DZV.MAP, ACTIVE ;GET MAP POINTER.
(1) 002226 012737 000001 001412      MOV     @1, RUN      ;POINT POINTER TO FIRST DEVICE.
(1) 002234 005037 001256      CLR     @ERTTL       ;CLEAR ERROR COUNT
(1) 002240 005037 001262      CLR     @ERRPC       ;CLEAR LAST ERROR POINTER
(1) 002244 005037 001246      CLR     @TSTNM       ;SET UP FOR TEST 1
(1) 002250 012737 002116 001252      MOV     @.START, @LPADR ;SET UP FOR POWER FAIL BEFORE
(1)                                     ;TESTING STARTS
(1)                                     ;SET UP FOR SMALL 11 SWITCH REGISTER COMPATIBILITY
(1) 002256 012737 000176 001304      MOV     @SMREG, SMR   ;POINT TO SOFTWARE SMR
(1) 002264 012737 000174 001306      MOV     @DISPREG, DISPLAY ;POINT TO SOFTWARE DISPLAY REGISTER
(1) 002272 004737 017332      CALL   FALCON        ; CHECK FOR FALCON (KXT11)    ;;GPA
(1) 002276 001402                BEQ     1000#        ; BR IF NOT                    ;;GPA
(1) 002300 004737 000570      CALL   FALCINI       ; YES, INIT FOR FALCON.      ;;GPA
(1) 002304      1000#:
(1) 002304 105737 001422      TSTB   INIFLG        ;HAVE WE ALREADY BEEN HERE TODAY?
(1) 002310 001014                BNE     10#          ;IF SO, SKIP PRINTING THE TITLE
(1) 002312 023727 000042 004420      CMP     @#42, @#ENDAD ;IF RUNNING UNDER ACT
(1) 002320 001406                BEQ     1#          ;DON'T PRINT TITLE
(1) 002322 032737 000040 000052      BIT     @BITS, @#52  ;ARE WE UNDER UFD ?
(1) 002330 001002                BNE     1#          ;YES, THEN SKIP TYPEOUT
(1) 002332 101032 001000      TYPE   ,MYTITLE     ;PRINT THE DIAGNOSTIC'S TITLE
(1) 002336 105037 001422      1#:  DECB   INIFLG    ;SET THE ONCE ONLY FLAG
(1) 002342 105737 001141      10#: TSTB   @ENVM       ;DETERMINE WHETHER APT SIZING SHOULD BE DONE
(1) 002346 100004                BPL     15#         ;IF NOT, GO CHECK FOR AUTO-SIZING
(1) 002350 004737 011436      JSR    PC, SETAPT   ;OTHERWISE, GO DO APT SIZING FROM ETABLE

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-17
 CVDZBD.P11 21-AUG-84 08:28 PROGRAM INITIALIZATION AND START UP.

```

(1) 002354 000137 003710          JMP      1054      ;GO PRINT DZV STATUS TABLE
(1) 002360 032737 000040 000052 154: BIT      @BITS,@#52 ;ARE WE UNDER UFD ?
(1) 002366 001015          BNE     174      ;YES, THEN GO AUTOSIZE THE SETUP
(1) 002370 005737 000042          TST     @#42     ; CHAINED UNDER XXDP ??      ;:GPA
(1) 002374 001410          BE      164      ; OR IF NOT                  ;:GPA
(1) 002376 032737 000040 000052  BIT      @BITS,@#52 ;ARE WE UNDER UFD ?
(1) 002404 001006          BNE     174      ;YES THEN GO CHECK THE SWITCHES
(1) 002406 004737 011436          CALL   SETAPT   ; YES, SET-UP FROM STABLE    ;:GPA
(1) 002412 000137 003710          JMP     1054     ; AND PROCEED?              ;:GPA
(1) 002416 004737 007350          164: CALL   GETSMR   ; GET AN INITIAL SMR        ;:GPA
(1) 002422 032777 000001 176654 174: BIT      @SM00,BSMR ;RESELECT ?
(1) 002430 001002          BNE     204     ;IF YES, GO SET UP THE INFORMATION
(1) 002432 000137 002734          JMP     554     ;IF NO, SKIP THE INTERROGATION
(1) 002436 012700 001500          204: MOV     @DZV,MAP,RO ;POINT TO THE BEGINNING OF THE MAP TABLE
(1) 002442 105037 001423          CLR    HDRFLG  ;MAKE SURE A MAP GETS PRINTED
(1) 002446 005020          254: CLR    (RO)    ;CLEAR A TABLE LOCATION
(1) 002450 020027 001740          CMP    RO,@DZV.END ;HAVE THE TABLE BOUNDARIES BEEN EXCEEDED?
(1) 002454 001374          BNE     254     ;IF NOT ,CLEAR THE NEXT LOCATION IN THE TABLE
(1) 002456 105337 001422          DECB   INIFLG  ;INSURE NO AUTO SIZING IF QUESTIONS ANSWERED'
    
```

;THE FOLLOWING ARE PARAMETERS USED TO FILL IN THE MAP
 ;TABLE AND SET UP THE DIAGNOSTIC.

```

(1) ;GET THE BASE ADDRESS OF THE DZV11'S
(1) GETCSR= . ; POINTER FOR FALCON TWEAKER. ;:GPA
(2) 002462 104403 INSTR ;CALL THE STRING INPUT ROUTINE
(2) 002464 003154 914 ;POINTER TO MESSAGE TO BE PRINTED
(2) 002466 104405 PARAM ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
(2) 002470 160000 160000 ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 002472 167770 167770 ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 002474 001500 DZCRO ;POINTER TO MAP LOCATION TO BE FILLED
(2) 002476 007 .BYTE 7 ;MASK OF INVALID BITS FOR THIS PARAMETER
(2) 002477 001 .BYTE 1 ;NUMBER OF PARAMETERS TO STORE
(1) 002500 013737 001500 001174 MOV     DZCRO,@BASE ;COPY BASE ADDRESS TO ETABLE
(1) ;GET THE BASE VECTOR ADDRESS
(1) GETVEC= . ; POINTER FOR FALCON TWEAKER. ;:GPA
(2) 002506 104403 INSTR ;CALL THE STRING INPUT ROUTINE
(2) 002510 003220 924 ;POINTER TO MESSAGE TO BE PRINTED
(2) 002512 104405 PARAM ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
(2) 002514 000300 300 ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 002516 000776 776 ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 002520 001502 DZVCO ;POINTER TO MAP LOCATION TO BE FILLED
(2) 002522 003 .BYTE 3 ;MASK OF INVALID BITS FOR THIS PARAMETER
(2) 002523 001 .BYTE 1 ;NUMBER OF PARAMETERS TO STORE
(1) 002524 013737 001502 001170 MOV     DZVCO,@VECT1 ;COPY VECTOR TO ETABLE
(1) ;GET THE MODE OF OPERATION (E,I,S)
(2) 002532 104403 INSTR ;CALL THE STRING INPUT ROUTINE
(2) 002534 003447 964 ;POINTER TO THE MESSAGE TO BE PRINTED
(2) 002536 104406 SETFLG ;CALL THE MAINTENANCE FLAG SETUP ROUTINE
(2) 002540 001510 MANTO ;THIS IS THE FLAG BEING SETUP
(1) ;GET THE NUMBER OF DZV11'S RUNNING
(2) 002542 104403 INSTR ;CALL THE STRING INPUT ROUTINE
(2) 002544 003404 954 ;POINTER TO MESSAGE TO BE PRINTED
    
```


CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-20
 CVDZBD.P11 21-AUG-84 08:28 PROGRAM INITIALIZATION AND START UP.

```

(1) 003620 042600 052116 051105 974: .ASCIZ <200>/ENTER DELAY PARAMETER: /
(1) 003652 003652 .EVEN
(1) 003652 122737 000377 001422 1004: CMPB #377,INIFLG ;ONLY DO AUTO SIZE ON 1ST START
(1) 003640 001013 BNE 1054 ;
(1) 003662 032777 000200 175414 BIT #BIT7,BSMR ;BIT7-1??
(1) 003670 001007 BNE 1054 ;BR IF NO AUTO SIZE
(1) 003672 005737 017346 TST KXTFLAG ; KXT11 ?? ;GPA
(1) 003676 001402 BEQ 10014 ; SKIP NEXT IF NOT ;GPA
(1) 003700 000137 002436 JMP 204 ; YES, DON'T AUTO-SIZE ;GPA
(1) 003704 10014: JSR PC,AUTO.SIZE ;GO DO THE AUTO SIZE
(1) 003704 004737 011564 TSTB HDRFLG ;HAS THE TABLE BEEN TYPED YET?
(1) 003710 105737 001423 BNE 1204 ;IF SO, DON'T TYPE IT AGAIN
(1) 003714 001021 DECB HDRFLG ;INDICATE THAT THE TABLE WILL BE TYPED
(1) 003716 105337 001423 TYPE ,XHEAD ;TYPE MAP HEADER
(1) 003722 104402 010322 MOV #DZV,MAP,RO ;SET POINTER
(1) 003726 012700 001500 MOV RO,#TMP1 ;POINT TO THE MAP LOCATION
(1) 003732 010037 001344 MOV (RO),#TMP2 ;SET DATA
(1) 003736 012037 001346 CMP #0-1,#TMP2 ;END OF LIST?
(1) 003742 022737 177777 001346 BEQ 1204 ;BR IF YES
(1) 003750 001403 1154: CONVRT ;CALL THE OCTAL TO ASCII CONVERSION ROUTINE
(1) 003752 104411 XSTATQ ;CONVERT THE DATA AT THIS ADDRESS
(1) 003754 010412 BR 1104 ;GO PRINT THE NEXT PARAMETER
(1) 003756 000765 MOV SAVACTV,DZVACTV ;COPY BIT MAP OF ACTIVE DEVICES
(1) 003760 013737 001410 001406 MOV# DZVNUM,SAVNO ;COPY NO. OF DEVICES IN THE SYSTEM
(1) 003766 113737 001414 001416 BIT #SM06,BSMR ;DESELECT SPECIFIC DEVICES??
(1) 003774 032777 000100 175302 BEQ 1354 ;BR IF NO.
(1) 004002 001431 1214: INSTR ;CALL THE STRING INPUT ROUTINE
(2) 004004 104403 MNEW ;POINTER TO MESSAGE TO BE PRINTED
(2) 004006 010240 PARAM ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
(2) 004010 104405 1 ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 004012 000001 177777 ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 004014 177777 DZVACTV ;POINTER TO MAP LOCATION TO BE FILLED
(2) 004016 001406 .BYTE 0 ;MASK OF INVALID BITS FOR THIS PARAMETER
(2) 004020 000 .BYTE 1 ;NUMBER OF PARAMETERS TO STORE
(2) 004021 001 CMP DZVACTV,SAVACTV ;IS VALUE VALID?
(1) 004022 023737 001406 001410 BLOS 1224 ;IF YES BRANCH
(1) 004030 101403 TYPE ,MERR3 ;IF NOT TYPE ERROR
(1) 004032 104402 010112 BR 1214 ;THEN REASK QUESTION
(1) 004036 000762 CLR# SAVNO ;INITIALIZE NO. OF ACTIVE DEVICES
(1) 004040 105037 001416 1224: MOV DZVACTV,#TMP1 ;COPY BIT MAP OF ACTIVE DEVICES
(1) 004044 013737 001406 001344 1264: ASR #TMP1 ;ROTATE OUT AN ACTIVE BIT
(1) 004052 006237 001344 BCC 1274 ;IF NOT ACTIVE SKIP RECORDING IT
(1) 004056 105002 INCB SAVNO ;INCREMENT NO. OF ACTIVE DEVICES
(1) 004060 105237 001416 1274: BNE 1264 ;IF NOT DONE GO CONTINUE
(1) 004064 001372 1354: BIT #SM04,BSMR ;CHECK TO SEE IF DELAY COUNT CHANGES
(1) 004066 032777 000020 175210 BEQ 1404 ;IF NOT, GO CLEAR VECTOR AREA
(1) 004074 001407 INSTR ;CALL THE STRING INPUT ROUTINE
(2) 004076 104403 974 ;POINTER TO MESSAGE TO BE PRINTED
(2) 004100 003620 PARAM ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
(2) 004102 104405 1 ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 004104 000001 177777 ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
(2) 004106 177777 DLYCNT ;POINTER TO MAP LOCATION TO BE FILLED
(2) 004110 006476 .BYTE 0 ;MASK OF INVALID BITS FOR THIS PARAMETER
(2) 004112 000

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-21
 CVDZBO.P11 21-AUG-84 08:28 PROGRAM INITIALIZATION AND START UP.

```

(2) 004113 001 .BYTE 1 ;NUMBER OF PARAMETERS TO STORE
(1) 004114 012700 000300 1400: MOV #300,R0 ;PREPARE TO CLEAR THE FLOATING
(1) 004120 012701 000302 MOV #302,R1 ;VECTOR AREA. 300-776
(1) 004124 010120 1450: MOV R1,(R0). ;START PUTTING "PC+2 - HALT"
(1) 004126 005021 CLR (R1). ;IN VECTOR AREA.
(1) 004130 022021 CMP (R0).,(R1). ;POP POINTERS
(1) 004132 005737 017346 TST KXTFLAG ; IF FALCON... ;GPA
(1) 004136 001403 BEQ 10020 ;GPA
(1) 004140 020027 000400 CMP R0,#400 ;...QUIT AT 400. ;GPA
(1) 004144 000402 402 ;SKIP NEXT ;GPA
(1) 004146 10020: CMP #1000,R0 ;ALL DONE?? ;GPA
(1) 004146 022700 001000 BNE 1450 ;BR IF NO.
(1) ;TEST START AND RESTART
(1) ;-----
(1) .BEGIN: MOV #STACK,SP ;SET UP STACK
(1) 004154 012706 001120 MTPS #MASK ;LOCK OUT INTERRUPTS
(1) 004160 106427 000200 TST B#2 ;IS PROGRAM UNDER MONITOR CONTROL
(1) 004164 005737 000042 BNE 20 ;BR IF YES
(1) 004170 001015 BIT #BIT2,BSWR ;CHECK FOR LOCK ON TEST
(1) 004172 032777 000004 175104 BEQ 10 ;BR IF NO LOCK DESIRED.
(1) 004200 001406 TYPE ,MLOCK ;TYPE LOCK SELECTED.
(1) 004202 104402 010136 MOV #NOP,TTST ;ADJUST SCOPE ROUTINE.
(1) 004206 012737 000240 004476 BR 20 ;CONTINUE ALONG.
(1) 004214 000403 10: MOV BRW,TTST ;PREPARE NORMAL SCOPE ROUTINE
(1) 004216 013737 004724 004476 20: MOV #CYCLE,#LPADR ;START AT "CYCLE" FIND WHICH DEVICE TO TEST
(1) 004224 012737 010724 001252 MOVBY SAVNO,SAVNUM ;COPY NO. OF ACTIVE DEVICES
(1) 004232 113737 001416 001415 TYPE ,MR ;TYPE "RUNNING"
(1) 004240 104402 010027 JMP #LPADR ;START TESTING
(1) 004244 000177 175002

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-22
CVDZBD.P11 21-AUG-84 08:28 END OF PASS ROUTINE

0457
(2)
(2)
(2)
(2)
(3)
(3)
(4)
(3)
(3)
(3)
(3)
(3)
(5) 004250
(5) 004250
(5) 004252
(5) 004256
(5) 004262
(5) 004266
(5) 004272
(5) 004276
(5) 004302
(5) 004306
(5) 004312
(5) 004316
(5) 004322
(5) 004326
(5) 004332
(5) 004336
(5) 004342
(5) 004346
(5) 004350
(3) 004356
(3) 004362
(3) 004366
(3) 004374
(3) 004376
(3) 004400
(3) 004402
(3) 004404
(3) 004406
(3) 004410
(3) 004414
(3) 004416
(3) 004420
(3) 004422
(3) 004424
(3) 004426
(3) 004430
(3) 004430
(3) 004432
(2)
(2) 004434
(2) 004436
(2) 004440
(2) 004442
(2) 004444

000004
005037 001262
105037 001247
104402 010003
104402 010165
104412 004434
104402 010173
104412 004442
005237 001126
104402 010201
104412 004450
005337 001126
104402 010212
104412 004456
005237 001130
105337 001415
001030
113737 001416 001415
005037 001354
005237 001126
042737 100000 001126
005327
000001
003013
012737
000001
004376
013700 000042
001405
000005
004710
000240
000240
000240
000137
010724
000001
006 002
002010
000001
003 002

```

;END OF PASS
;TYPE NAME OF TEST
;UPDATE PASS COUNT
;CHECK FOR EXIT TO ACT-11
;RESTART TEST
.SBTTL END OF PASS ROUTINE

;*****
;INCREMENT THE PASS NUMBER (#PASS)
;IF THERES A MONITOR GO TO IT
;IF THERE ISN'T JUMP TO CYCLE

$EOP:
SCOPE
CLR      $ERRPC      ;CLEAR LAST ERROR PC
CLRB     $ERFLG      ;CLEAR ERROR FLAG
TYPE     ,MEPASS     ;TYPE END PASS
TYPE     ,MCSRX      ;TYPE CSR
CNVRT    ,XCSR       ;SHOW IT
TYPE     ,MVECX      ;TYPE VECTOR
CNVRT    ,XVEC       ;SHOW IT
INC      $PASS       ;RAISE PASS COUNT
TYPE     ,MPASSX     ;TYPE PASSES
CNVRT    ,XPASS      ;SHOW IT
DEC      $PASS       ;RESTORE PASS COUNT
TYPE     ,MERRX      ;TYPE ERRORS
CNVRT    ,XERR       ;SHOW IT
INC      $DEVCT      ;INC DEVCNT FOR APT
DECB     SAVNUM      ;ARE ALL DEVICES TESTED?
BNE      $DOAGN      ;BR IF NO.
NOVB     SAVNO,SAVNUM ;RESTORE THE COUNT
CLR      $TIMES      ;;ZERO THE NUMBER OF ITERATIONS
INC      $PASS       ;;INCREMENT THE PASS NUMBER
BIC      @100000,$PASS ;DON'T ALLOW A NEG. NUMBER
DEC      (PC)        ;;LOOP?

$EOPCT: .WORD 1
BGT      $DOAGN      ;;YES
MOV      (PC)+,B(PC) ;RESTORE COUNTER

$ENDCT: .WORD 1

$GET42: MOV      B@42,R0      ;;GET MONITOR ADDRESS
        BEQ      $DOAGN      ;;BRANCH IF NO MONITOR
        RESET
$ENDAD: JSR      PC,(R0)     ;;CLEAR THE WORLD
        NOP
        NOP
        NOP                ;;GO TO MONITOR
                                ;;SAVE ROOM
                                ;;FOR
                                ;;ACT11

$DOAGN: JMP      B(PC)        ;;RETURN

$RTNAD: .WORD  CYCLE

XCSR:   1
        .BYTE  6,2
        DZVCSR

XVEC:   1
        .BYTE  3,2
```


CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-24
 CVDZBD.P11 21-AUG-84 08:28 SCOPE HANDLER ROUTINE

```

(3) 004632 012737 000001 001250 10:  MOV 01, #ICNT ;REINITIALIZE THE ITERATION COUNTER
(3) 004640 013737 004726 001354      MOV #MXCNT, #TIMES ;SET NUMBER OF ITERATIONS TO DO
(3) 004646 105237 001246      #SVLAD: INCB #TSTNM ;COUNT TEST NUMBERS
(3) 004652 113737 001246 001124      MOV #TSTNM, #TESTN ;SET TEST NUMBER IN APT MAILBOX
(3) 004660 011637 001252      MOV (SP), #LPADR ;SAVE SCOPE LOOP ADDRESS
(3) 004664 013777 001246 174414 #OVER: MOV #TSTNM, #DISPLAY ;DISPLAY TEST NUMBER
(3) 004672 013716 001252      MOV #LPADR, (SP) ;FUDGE RETURN ADDRESS
(5) 004676 004737 007322      JSR PC, SERV.G ;FIND OUT IF %G WAS TYPED
(5) 004702 105037 001424      CLRB MNTFLG ;CLEAR THE MAINTENANCE BIT SETTER AFTER EACH TEST
(5) 004706 005737 001372      TST MODE ;HAS THE MODE BEEN CHANGED?
(5) 004712 001003      BNE 40 ;IF NOT INTERNAL, GO DO A TEST
(5) 004714 112737 000010 001424      MOV #MAINT, MNTFLG ;IF INTERNAL MODE NOW, SET THE MAINTENANCE BIT
(5) 004722 000002      RTI ;GO DO THE TEST
(5) 004724 000406      BRW: 406
(3) 004726 000005      #MXCNT: 5 ;MAX. NUMBER OF ITERATIONS

(1) ;CHECK FOR FREEZE ON CURRENT DATA
(1) -----
(1)
(1)
(1) 004730 032777 001000 174346 .SCOP1: BIT #SM09, #SMR ;IS SM09=1(SET)?
(1) 004736 001405      BEQ 10 ;BR IF NOT SET.
(1) 004740 005737 001364      TST LOCK ;IS THERE A TIGHT LOOP SPECIFIED?
(1) 004744 001402      BEQ 10 ;IF NO, RETURN
(1) 004746 013716 001364      MOV LOCK, (SP) ;IF YES, GOTO THE ADDRESS IN LOCK.
(1) 004752 000002      10: RTI ;GO BACK.

(1) 004754 032737 000040 000052 .TYPE: BIT #BITS, #M52 ;ARE WE UNDER UFD ?
(1) 004762 001004      BNE 10 ;YES, THEN SKIP ALL TYPEOUTS
(1) 004764 032777 010000 174312      BIT #SM12, #SMR ;INHIBIT ALL PRINTOUT??
(1) 004772 001403      BEQ #TYPE ;IF NOT, GO TYPE
(1) 004774 062716 007002      10: ADD #2, (SP) ;SKIP OVER MESSAGE POINTER
(1) 005000 000002      RTI ;RETURN TO WHERE PROCEDURE WAS INVOKED

(2) .SBTTL TYPE ROUTINE
(2)
(3) ;*****
(2) ;ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
(2) ;THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
(2) ;NOTE1: #NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
(2) ;NOTE2: #FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
(2) ;NOTE3: #FILLC CONTAINS THE CHARACTER TO FILL AFTER.
(2) ;*
(2) ;CALL:
(2) ;*1) USING A TRAP INSTRUCTION
(2) ;* TYPE ,MESADR ;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
(2) ;OR
(2) ;* TYPE
(2) ;* MESADR
(2) ;*
(2)
(2) 005002 105737 001323      #TYPE: TSTB #TPFLG ;IS THERE A TERMINAL?
(2) 005006 100002      BPL 10 ;BR IF YES
(2) 005010 000000      HALT ;HALT HERE IF NO TERMINAL
(2) 005012 000430      BR 30 ;LEAVE
(2) 005014 010046      10: MOV R0, -(SP) ;SAVE R0
(2) 005016 017600 000002      MOV #2(SP), R0 ;GET ADDRESS OF ASCIZ STRING
(2) 005022 122737 000001 001140      CHPB #APTENV, #ENV ;RUNNING IN APT MODE

```

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-25
 CVDZBO.P11 21-AUG-84 08:28 TYPE ROUTINE

```

(2) 005030 001011          BNE      62#          ;;NO,GO CHECK FOR APT CONSOLE
(2) 005032 132737 000100 001141 BITB    @APTSPOOL,@ENVH ;;SPOOL MESSAGE TO APT
(2) 005040 001405          BEQ      62#          ;;NO,GO CHECK FOR CONSOLE
(2) 005042 010037 005052    MOV      RO,61#      ;;SETUP MESSAGE ADDRESS FOR APT
(2) 005046 004737 005344    JSR     PC,@ATY3    ;;SPOOL MESSAGE TO APT
(2) 005052 000000          .WORD   0            ;;MESSAGE ADDRESS
(2) 005054 132737 000040 001141 61#:   BITB    @APTCSUP,@ENVH ;;APT CONSOLE SUPPRESSED
(2) 005062 001003          BNE     60#          ;;YES,SKIP TYPE OUT
(2) 005064 112046          MOVB   (RO)*,-(SP)  ;;PUSH CHARACTER TO BE TYPED ONTO STACK
(2) 005066 001005          BNE     4#          ;;BR IF IT ISN'T THE TERMINATOR
(2) 005070 005726          TST    (SP)*        ;;IF TERMINATOR POP IT OFF THE STACK
(2) 005072 012600          MOV     (SP)*,RO    ;;RESTORE RO
(2) 005074 062716 000002    ADD     @2,(SP)     ;;ADJUST RETURN PC
(2) 005100 000002          RTI                    ;;RETURN
(2) 005102 122716 000011    4#:   CMPB   @HT,(SP)   ;;BRANCH IF <HT>
(2) 005106 001430          BEQ     8#          ;;BRANCH IF NOT <CRLF>
(2) 005110 122716 000200    CMPB   @CRLF,(SP)
(2) 005114 001006          BNE     5#          ;;POP <CR><LF> EQUIV
(2) 005116 005726          TST    (SP)*        ;;TYPE A CR AND LF
(2) 005120 104402          TYPE
(2) 005122 001357          @CRLF
(2) 005124 105037 005332    CLRB   @CHARCNT    ;;CLEAR CHARACTER COUNT
(2) 005130 000755          BR     2#          ;;GET NEXT CHARACTER
(2) 005132 004737 005214    5#:   JSR     PC,@TYPEC  ;;GO TYPE THIS CHARACTER
(2) 005136 123726 001322    6#:   CMPB   @FILLC,(SP)* ;;IS IT TIME FOR FILLER CHARS.?
(2) 005142 001350          BNE     2#          ;;IF NO GO GET NEXT CHAR.
(2) 005144 013746 001320    MOV     @NULL,-(SP) ;;GET # OF FILLER CHARS. NEEDED
(2)                                ;;AND THE NULL CHAR.
(2) 005150 105366 000001    7#:   DECB   1(SP)      ;;DOES A NULL NEED TO BE TYPED?
(2) 005154 002770          BLT    6#          ;;BR IF NO--GO POP THE NULL OFF OF STACK
(2) 005156 004737 005214    JSR     PC,@TYPEC  ;;GO TYPE A NULL
(2) 005162 105337 005332    DECB   @CHARCNT    ;;DO NOT COUNT AS A COUNT
(2) 005166 000770          BR     7#          ;;LOOP

(2)                                ;HORIZONTAL TAB PROCESSOR
(2) 005170 112716 000040    8#:   MOVB   #' ,(SP)   ;;REPLACE TAB WITH SPACE
(2) 005174 004737 005214    9#:   JSR     PC,@TYPEC  ;;TYPE A SPACE
(2) 005200 132737 000007 005332 BITB    @7,@CHARCNT ;;BRANCH IF NOT AT
(2) 005206 001372          BNE     9#          ;;TAB STOP
(2) 005210 005726          TST    (SP)*        ;;POP SPACE OFF STACK
(2) 005212 000724          BR     2#          ;;GET NEXT CHARACTER
(2) 005214          @TYPEC:
(2) 005214 105777 174070    TSTB   @TKS        ;;CHAR IN KYBD BUFFER?
(2) 005220 100022          BPL    10#         ;;BR IF NOT
(2) 005222 017746 174064    MOV     @TKB,-(SP)  ;;GET CHAR
(2) 005226 042716 177600    BIC    @177600,(SP) ;;STRIP EXTRANEIOUS BITS
(2) 005232 122716 000023    CMPB   @XOFF,(SP)  ;;WAS CHAR XOFF
(2) 005236 001012          BNE     102#       ;;BR IF NOT
(2) 005240          101#:
(2) 005240 105777 174044    TSTB   @TKS        ;;WAIT FOR CHAR
(2) 005244 100375          BPL    101#       ;;BR IF NOT
(2) 005246 117716 174040    MOVB   @TKB,(SP)   ;;GET CHAR
(2) 005252 042716 177600    BIC    @177600,(SP) ;;STRIP IT
(2) 005256 122716 000021    CMPB   @XON,(SP)   ;;WAS IT XON?
(2) 005262 001366          BNE     101#       ;;BR IF NOT
    
```


CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-27
 CVDZBO.P11 21-AUG-84 08:28 APT COMMUNICATIONS ROUTINE

```

(2) 005536 017637 000004 001122      MOV      04(SP),%FATAL      ;;GET ERROR #
(2) 005544 062766 000002 000004      ADD      02,4(SP)          ;;BUMP RETURN ADDR.
(2) 005552 005237 001120              INC      %MSGTYPE          ;;TELL APT TO TAKE ERROR
(2) 005556 105037 005602      124:    CLR     %FFLG              ;;CLEAR FATAL FLAG
(2) 005562 105037 005601              CLR     %LFLG              ;;CLEAR LOG FLAG
(2) 005566 105037 005600              CLR     %MFLG              ;;CLEAR MESSAGE FLAG
(4) 005572 012601              MOV     (SP)+,R1           ;;POP STACK INTO R1
(4) 005574 012600              MOV     (SP)+,R0           ;;POP STACK INTO R0
(2) 005576 000207              RTS     PC                  ;;RETURN
(2) 005600      000      %MFLG: .BYTE 0           ;;MESSG. FLAG
(2) 005601      000      %LFLG: .BYTE 0           ;;LOG FLAG
(2) 005602      000      %FFLG: .BYTE 0           ;;FATAL FLAG
(2)                                .EVEN
(2)                                APTSIZE=200
(2)                                APTENW=001
(2)                                APTSPOOL=100
(2)                                APTCSUP=040
(1)
(1)                                ;STRING INPUT ROUTINE
(1)                                -----
(1) 005604 010346      .INSTR: MOV     R3,-(SP)      ;SAVE R3 ON STACK
(1) 005606 010446      MOV     R4,-(SP)      ;SAVE R4 ON STACK
(1) 005610 017637 000004 005626      MOV     04(SP),.MSG    ;GET THE ADDRESS OF THE MESSAGE TO BE PRINTED
(1) 005616 062766 000002 000004      ADD     02,4(SP)      ;POINT TO INSTRUCTION AFTER ADDRESS POINTER
(1) 005624 104402      .INST1: TYPE          ;PRINT THE MESSAGE
(1) 005626 000000      .MSG: 0               ;MESSAGE IS POINTED TO FROM HERE
(1) 005630 012704 010620      MOV     0(INBUF,R4)    ;POINT R4 TO THE INPUT BUFFER
(1) 005634 012703 000007      MOV     07,R3          ;SET THE MAXIMUM NUMBER OF CHARACTERS ALLOWED
(1) 005640 105777 173444      14:    TSTB   04TKS          ;HAS A CHARACTER BEEN RECEIVED?
(1) 005644 100375      BPL     14             ;IF NO, KEEP WAITING FOR IT
(1) 005646 117714 173440      MOV     04TKB,(R4)     ;IF YES, SAVE IT IN THE INPUT BUFFER
(1) 005652 142714 000200      BICB   0200,(R4)      ;KEEP ONLY THE 7-BIT ASCII INFORMATION
(1) 005656 122427 000015      CMPB   (R4)+,%15      ;IS THIS CHARACTER A LINE FEED?
(1) 005662 001417      BEQ     INSTR2         ;IF SO, TERMINATE THE INPUT SEQUENCE
(1) 005664 105777 173424      24:    TSTB   04TPS          ;IF NOT, CHECK TO SEE IF THE CHARACTER CAN PRINT
(1) 005670 100375      BPL     24             ;IF WE CAN'T, WAIT UNTIL WE CAN
(1) 005672 017777 173414 173416      MOV     04TKB,04TPB    ;ECHO THE CHARACTER BACK
(1) 005700 005303      DEC     R3             ;REDUCE THE NUMBER OF CHARACTERS RECEIVED
(1) 005702 001356      BNE     14             ;IF WE DON'T HAVE 7, GO GET SOME MORE
(1) 005704 012604      MOV     (SP)+,R4       ;IF WE HAVE 7, RESTORE R4
(1) 005706 012603      MOV     (SP)+,R3       ;RESTORE R3
(1) 005710 010346      .INSTE: MOV     R3,-(SP) ;SAVE R3 ON THE STACK
(1) 005712 010446      MOV     R4,-(SP)      ;SAVE R4 ON THE STACK
(1) 005714 104402 001356      TYPE   ,QUES          ;PRINT A QUESTION MARK... WHAT'S GOING ON?
(1) 005720 000741      BR     .INST1          ;GO PRINT THE MESSAGE AGAIN
(1) 005722 012604      INSTR2: MOV     (SP)+,R4 ;RESTORE R4
(1) 005724 012603      MOV     (SP)+,R3       ;RESTORE R3
(1) 005726 000002      RTI                    ;RETURN TO THE MAIN PROCEDURE
(1)
(1)                                ;CONVERT ASCII STRING TO OCTAL
(1)                                -----
(1) 005730 010546      .PARAM: MOV     R5,-(SP) ;SAVE R5 ON THE STACK
(1) 005732 010446      MOV     R4,-(SP)      ;SAVE R4 ON THE STACK
(1) 005734 016605 000004      MOV     4(SP),R5      ;GET THE SETUP INFORMATION POINTER
    
```

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-28
CVDZBO.P11 21-AUG-84 08:28 APT COMMUNICATIONS ROUTINE

```

(1) 005740 012537 006120      MOV      (R5)+,LOLIM      ;SET THE LOW LIMIT FOR THE INPUT
(1) 005744 012537 006122      MOV      (R5)+,HILIM      ;SET THE HIGH LIMIT FOR THE INPUT
(1) 005750 012537 006124      MOV      (R5)+,DEVADR      ;SAVE THE ADDRESS WHERE THE RESULT WILL BE STORED
(1) 005754 112537 006126      MOV      (R5)+,LOBITS      ;GET THE MASK OF THE INCORRECT BITS
(1) 005760 112537 006127      MOV      (R5)+,ADRCNT      ;GET THE COUNT OF ITEMS TO BE STORED
(1) 005764 010566 000004      MOV      R5,4(SP)        ;POINT TO WHERE MAIN LINE PROGRAM WILL RESUME
(1) 005770 005005      PARAM1: CLR      R5        ;INITIALIZE THE ASCII TO OCTAL RESULT WORD
(1) 005772 012704 010620      MOV      @INBUF,R4        ;POINT TO THE INPUT BUFFER
(1) 005776 122714 000015      CMPB     @15,(R4)         ;IS THIS CHARACTER A CARRIAGE RETURN?
(1) 006002 001420      BEQ      PARERR          ;IF SO, PRINT THE MESSAGE AGAIN
(1) 006004 121427 000060      11:      CMPB     (R4),#60        ;IS THIS CHARACTER BELOW THE NUMERIC RANGE?
(1) 006010 002415      BLT      PARERR          ;IF SO, GO PRINT THE MESSAGE AGAIN
(1) 006012 121427 000067      CMPB     (R4),#67        ;IS THIS CHARACTER ABOVE THE NUMERIC RANGE?
(1) 006016 003012      BGT      PARERR          ;IF SO, GO PRINT THE MESSAGE AGAIN
(1) 006020 142714 000060      BICB     @60,(R4)        ;ISOLATE THE NUMBER THE CHARACTER REPRESENTS
(1) 006024 152405      BISB     (R4)+,R5        ;CONCATENATE THESE BITS TO THE ALREADY EXISTING STRING
(1) 006026 122714 000015      CMPB     @15,(R4)        ;IS THE NEXT CHARACTER A CARRIAGE RETURN?
(1) 006032 001406      BEQ      LIMITS          ;IF SO, GO SEE IF NUMBER IS WITHIN LIMITS
(1) 006034 006305      ASL      R5              ;CLEAR BIT POSITION 0, MOVE EXISTING STRING TO LEFT
(1) 006036 006305      ASL      R5              ;CLEAR POSITION 1, MOVE STRING TO LEFT AGAIN
(1) 006040 006305      ASL      R5              ;MOVE THE STRING ONE MORE TIME TO MAKE ROOM FOR
(1)                                ;NEXT THREE BITS
(1) 006042 000760      BR       11              ;GO GET THE NEXT CHARACTER
(1) 006044 104404      PARERR: INSTER          ;THERE WAS AN ERROR... GO PRINT MESSAGE AGAIN
(1) 006046 000750      BR       PARAM1         ;TRY GETTING THE PARAMETERS AGAIN
(1)                                ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
(1)                                ;-----
(1) 006050 020537 006122      LIMITS: CMP      R5,HILIM      ;DOES RESULT EXCEED ITS MAXIMUM CORRECT VALUE?
(1) 006054 101373      BHI      PARERR          ;IF YES, GO PRINT THE MESSAGE AGAIN
(1) 006056 020537 006120      CMP      R5,LOLIM        ;IS THE RESULT LOWER THAN ALLOWED?
(1) 006062 103770      BLO      PARERR          ;IF YES, GO PRINT THE MESSAGE AGAIN
(1) 006064 133705 006126      BITB     LOBITS,R5        ;ARE ANY INCORRECT BITS SET IN THE RESULT?
(1) 006070 001365      BNE      PARERR          ;IF SO, GO PRINT THE MESSAGE AGAIN
(1)                                ;STORE NUMBER AT SPECIFIED ADDRESS
(1) 006072 013704 006124      11:      MOV      DEVADR,R4        ;POINT TO THE LOCATION WHERE THE RESULT WILL BE STORED
(1) 006076 010524      MOV      R5,(R4)+        ;STORE THE RESULT
(1) 006100 062705 000002      ADD      @2,R5           ;CALCULATE THE NEXT DATUM
(1) 006104 105337 006127      DECB     ADRCNT          ;REDUCE COUNT OF STORED RESULTS. IS IT EXCEEDED?
(1) 006110 001372      BNE      11              ;IF NOT, GO STORE THE NEXT DATUM
(1) 006112 012604      MOV      (SP)+,R4        ;RESTORE R4
(1) 006114 012605      MOV      (SP)+,R5        ;RESTORE R5
(1) 006116 000002      RTI                       ;RETURN TO THE MAIN PROGRAM
(1) 006120 000000      LOLIM:  0                ;LOWEST ACCEPTABLE VALUE
(1) 006122 000000      HILIM:  0                ;HIGHEST ACCEPTABLE
(1) 006124 000000      DEVADR: 0                ;LOCATION WHERE RESULT WILL BE STORED
(1) 006126      000      LOBITS: .BYTE 0      ;INCORRECT BITS MASK
(1) 006127      000      ADRCNT: .BYTE 0        ;COUNT OF ITEMS TO BE STORED
(1)                                ;SAVE PC OF TEST THAT FAILED AND R0-R5
(1)                                ;-----

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-29
 CVDZEO.P11 21-AUG-84 08:28 APT COMMUNICATIONS ROUTINE

```

(1) 006130 016637 000004 001404 .SAV05: MOV 4(SP),SAVPC ;SAVE R7 (PC)
(1)
(1) ;SAVE R0-R5
(1)
(1) 006136 010537 001340 SV05: MOV R5,%REG5 ;SAVE R5
(1) 006142 010437 001336 MOV R4,%REG4 ;SAVE R4
(1) 006146 010337 001334 MOV R3,%REG3 ;SAVE R3
(1) 006152 010237 001332 MOV R2,%REG2 ;SAVE R2
(1) 006156 010137 001330 MOV R1,%REG1 ;SAVE R1
(1) 006162 010037 001326 MOV R0,%REG0 ;SAVE R0
(1) 006166 000002 RTI ;LEAVE.
(1)
(1) ;RESTORE R0-R5
(1)
(1) 006170 013700 001326 .RES05: MOV %REG0,R0 ;RESTORE R0
(1) 006174 013701 001330 MOV %REG1,R1 ;RESTORE R1
(1) 006200 013702 001332 MOV %REG2,R2 ;RESTORE R2
(1) 006204 013703 001334 MOV %REG3,R3 ;RESTORE R3
(1) 006210 013704 001336 MOV %REG4,R4 ;RESTORE R4
(1) 006214 013705 001340 MOV %REG5,R5 ;RESTORE R5
(1) 006220 000002 RTI ;LEAVE
(1)
(1) ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
(1)
(1)
(1)
(1) 006222 104402 001357 .CONVR: TYPE ;CRLF ;PRINT A CARRIAGE RETURN
(1) 006226 010046 .CNVRT: MOV R0,-(SP) ;SAVE R0
(1) 006230 010146 MOV R1,-(SP) ;SAVE R1
(1) 006232 010346 MOV R3,-(SP) ;SAVE R3
(1) 006234 010446 MOV R4,-(SP) ;SAVE R4
(1) 006236 010546 MOV R5,-(SP) ;SAVE R5
(1) 006240 017601 000012 MOV @12(SP),R1 ;PLACE THE ADDRESS OF THE ARGUMENTS IN R1
(1) 006244 062766 000002 000012 ADD @2,12(SP) ;POINT TO WHERE MAIN PROGRAM WILL RESUME
(1) 006252 012137 006376 MOV (R1),%WORDCNT ;GET NUMBER OF WORDS TO BE PRINTED
(1) 006256 112105 11: MOVB (R1),%R5 ;GET THE NUMBER OF CHARACTERS TO BE PRINTED
(1) 006260 112100 MOVB (R1),%R0 ;GET THE NUMBER OF SPACES TO PRINT
(1) 006262 013104 MOV @R1,%R4 ;COPY THE WORD TO BE CONVERTED
(1) 006264 110537 006400 MOVB R5,%CHRCNT ;COPY THE CHARACTER COUNT
(1) 006270 010403 31: MOV R4,R3 ;COPY THE ARGUMENT WORD AGAIN
(1) 006272 042703 177770 BIC @1C<7>,%R3 ;ISOLATE THREE BITS TO BE TREATED AS A CHARACTER
(1) 006276 062703 000060 ADD @060,%R3 ;MAKE AN ASCII CHARACTER OUT OF THEM
(1) 006302 110346 MOVB R3,-(SP) ;SAVE THAT CHARACTER
(1) 006304 006004 ROR R4 ;MOVE THE NEXT THREE BITS INTO PLACE
(1) 006306 006204 ASR R4 ;MOVE THEM AGAIN
(1) 006310 006204 ASR R4 ;AND FINALLY A THIRD TIME
(1) 006312 005305 DEC R5 ;REDUCE CHARACTER COUNT,ARE ALL CHARACTERS
(1) ;BUILT?
(1) 006314 001365 BNE 31 ;IF NO, GO BUILD THE NEXT ONE.
(1) 006316 012703 010662 MOV @0DATA,%R3 ;NOW POINT TO WHERE NUMBER WILL BE PRINTED FROM
(1) 006322 112623 41: MOVB (SP),%R3 ;STORE THE CHARACTER, STARTING WITH THE MOST
(1) 006324 105337 006400 DECB %CHRCNT ;REDUCE COUNT, ARE ALL CHARACTERS TRANSFERRED?
(1) 006330 001374 BNE 41 ;IF NO, GO TRANSFER ANOTHER
(1) 006332 105700 TSTB R0 ;ARE ANY SPACES TO BE PRINTED?
(1) 006334 001404 BEQ 61 ;IF NO, DON'T SET UP ANY
(1) 006336 112723 000040 51: MOVB @040,%R3 ;ADD A SPACE TO THE OUTPUT BUFFER
(1) 006342 105300 DECB R0 ;REDUCE THE COUNT, SHOULD WE PRINT MORE?
    
```



```
(1)
(1)
(1) 006500 013716 001362 .ADVANCE:MOV NEXT.(SP) ;CRUNCH STACK WITH ADDRESS OF SCOPE CALL
(1) 006504 005037 001364 CLR LOCK ;RESET TIGHT LOOP ADDRESS
(1) 006510 000002 RTI ;CHECK TO SEE IF OLD TEST GETS REPEATED
(1)
(1) ;ROUTINE TO SHIFT LINE POINTER
(1) ;AND SWITCH TESTS IF NECESSARY
(1)
(1) .SHIFT: ASLB R2 ;POINT TO THE NEXT LINE
(1) 006512 106302 BIT @BIT4,R2 ;HAVE WE PASSED ALL LINE POINTERS?
(1) 006514 032702 000020 BEQ 14 ;IF NOT, RETURN TO THE TEST
(1) 006520 001402 POP2SP ;REMOVE THE TRAP CALL FROM THE STACK
(1) 006522 022626 ADVANCE ;GO TO THE NEXT TEST
(1) 006524 104400 14: RTI ;RETURN TO THE PRESENT TEST
(1) 006526 000002
(1)
```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-32
 CVDZBO.P11 21-AUG-84 08:28 APT COMMUNICATIONS ROUTINE

```

(1)                                     ;LINE PARAMETER REGISTER SETUP ROUTINE
(1)
(1) 006530 010146 .LPRSET:MOV R1,-(SP) ;SAVE CONTENTS OF R1
(1) 006532 010240 MOV R2,-(SP) ;SAVE CONTENTS OF R2
(1) 006534 013701 001370 MOV PAR,R1 ;MOVE DEFAULT PARAM. INTO R1
(1) 006540 012702 000001 MOV @1,R2 ;INIT. FOR LINE 1
(1) 006544 010177 173250 11: MOV R1,@DZVLP R ;LOAD PARAM. REGISTER
(1) 006550 005201 INC R1 ;SET R1 FOR NEXT LINE
(1) 006552 106302 ASLB R2 ;SET R2 FOR NEXT LINE
(1) 006554 032702 000020 BIT @BIT4,R2 ;ALL LINES DONE?
(1) 006560 001771 BEQ 11 ;IF NO LOAD NEXT LINE
(1) 006562 012602 MOV (SP),R2 ;RELOAD R2
(1) 006564 012601 MOV (SP),R1 ;RELOAD R1
(1) 006566 000002 RTI ;RETURN
(1)
(1)                                     ;ROUTINE TO ZERO DATA BUFFER
(1)
(1) .BUFSET:MOV R0,-(SP) ;SAVE CONTENTS OF R0
(1) 006570 010046 MOV @TDO,R0 ;SET R0 TO TOP OF BUFFER
(1) 006572 012700 001426 11: CLR (R0) ;CLEAR BUFFER LOCATION
(1) 006576 005020 CMP @STOP,R0 ;IS BUFFER ALL CLEARED
(1) 006600 022700 001446 BNE 11 ;IF NOT CLEAR NEXT LOCATION
(1) 006604 001374 MOV (SP),R0 ;RELOAD R0
(1) 006610 000002 RTI ;RETURN
(1)
(2)
(3) ;*****
(2) .SBTTL ABORT ROUTINE FOR LCP/ORION UFD MODE
(3) ;*****
(2)
(2) UFD-BITS
(2) 006612 032737 000040 000052 ABORT: BIT @UFD,52 ;TEST FOR USER FRIENDLY MODE
(2) 006620 001420 BEQ ABORT2 ;IF NOT UFD THEN CONTINUE NORMAL OPERATION
(2)
(2) MOV SAV30,30 ;RESTORE EMT LOCATION (30)
(2) 006630 013737 002154 000032 MOV SAV32,32 ;RESTORE EMT PRIORITY LOCATION (32)
(2) 006636 104042 EMT +42 ;GET DCA LOCATION INTO R0 FROM MONITOR
(2) 006640 012760 177777 000042 MOV @-1,42(R0) ;SET A -1 INTO LOCATION DRSEMR IN MONITOR
(2) 006646 013700 000042 ABORT1: MOV 42,R0 ;AND PUT THE MONITOR RETURN ADDRESS IN R0
(2) 006652 005037 000042 CLR 42 ;CLEAR MONITOR RETURN FLAG
(2) 006656 000137 004420 JMP @ENDAD ;RETURN TO MONITOR-DO NOT PUSH STACK HERE
(2) 006662 000207 ABORT2: RTS PC ;IF NOT UFD RETURN TO MAINLINE
(2)
(3) ;*****
(1)
(1) ;ERROR HANDLER
(1) ;-----
(1)
(1) 006664 004737 006612 ;ERROR: JSR PC,ABORT ;ARE WE UNDER UFD ?
(1) 006670 004737 007322 JSR PC,SERV.G ;FIND OUT IF <+G> WAS HIT
(1) 006674 032777 010000 172402 BIT @SW12,@SHR ;BELL ON ERROR?
(1) 006702 001406 BEQ XBX ;OR IF NO BELL
(1) 006704 105777 172404 TSTB @ITPS ;TTY READY.
(1) 006710 100003 BPL XBX ;DON'T WAIT IF TTY NOT READY.

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-33
CVDZBD.P11 21-AUG-84 08:28 ABORT ROUTINE FOR LCP/ORION UFD MODE

(1)	006712	112777	000207	172376		MOVW	@207,@1TPB	PUSH A BELL AT THE TTY.
(1)	006720	032777	020000	172356	XBX:	BIT	@SM13,@SMR	DELETE ERROR PRINT OUT?
(1)	006726	001113				BNE	HALTS	BR IF NO PRINT OUT WANTED.
(1)	006730	021637	001262			CHP	(SP),@ERRPC	WAS THIS ERROR FOUND LAST TIME?
(2)	006734	001404				BEQ	11	BR IF YES
(1)	006736	011637	001262			MOV	(SP),@ERRPC	RECORD BEING HERE
(1)	006742	103037	001247			CLMB	@ERFLG	PREPARE HEADER
(1)	006746	104407			11:	SAV05		SAVE ALL PROC REGISTERS
(1)	006750	011605				MOV	(SP),R5	GET THE PC OF ERROR
(1)	006752	162705	000002			SUB	@2,R5	GET ADDRESS OF TRAP CALL
(1)	006756	011504				MOV	(R5),R4	GET ERROR INSTRUCTION
(1)	006760	110437	001260			MOVW	R4,@ITEMB	COPY TEST NUMBER FOR APT HANDLING
(1)	006764	006304				ASL	R4	MULT BY TWO
(1)	006766	061504				ADD	(R5),R4	DOUBLE IT
(1)	006770	006304				ASL	R4	MULT AGAIN
(1)	006772	042704	177001			BIC	@177001,R4	CLEAR JUNK
(1)	006776	062704	015422			ADD	@.ERRTAB,R4	GET POINTER
(1)	007002	012437	007126			MOV	(R4),@.ERRMSG	GET ERROR MESSAGE
(1)	007006	012437	007140			MOV	(R4),@.DATAND	GET DATA HEADER
(1)	007012	011437	007152			MOV	(R4),@DATABP	GET DATA TABLE
(1)	007016	105737	001247			TSTB	@ERFLG	TYPE HEADER
(1)	007022	001403				BEQ	TYPMSG	BR IF YES
(1)	007024	005737	007152			TST	DATABP	DOES DATA TABLE EXIST?
(1)	007030	001044				BNE	TYPDAT	BR IF YES.
(1)	007032	104402	001357		TYPMSG:	TYPE	,@CRLF	TYPE A CARRIAGE RETURN
(1)	007036	104402	001357			TYPE	,@CRLF	AND TYPE ANOTHER
(1)	007042	005737	001364			TST	LOCK	
(1)	007046	001402				BEQ	11	
(1)	007050	104402	010235			TYPE	,MASTEK	
(1)	007054	104402	010223		11:	TYPE	,MTSTN	
(1)	007060	104412	007314			CONVRT	,XTSTN	SHOW IT
(1)	007064	104402	010315			TYPE	,@ERRPC	TYPE PC.
(1)	007070	104412	007306			CONVRT	,ERTABO	SHOW IT
(1)	007074	104402	010165			TYPE	,MCSRX	
(1)	007100	104412	004434			CONVRT	,XCSR	
(1)	007104	104402	001357			TYPE	,@CRLF	GIVE A CR/LF
(1)	007110	112737	177777	001247		MOVW	@-1,@ERFLG	NO MORE HEADER UNLESS NO DATA TABLE.
(1)	007116	005737	007126			TST	@ERRMSG	IS THERE AN ERROR MESSAGE?
(1)	007122	001402				BEQ	@WTBS.FM	BR IF NO.
(1)	007124	104402				TYPE		TYPE
(1)	007126	000000			ERRMSG:	0		ERROR MESSAGE
(1)	007130				WTBS.FM:			
(1)	007130	005737	007140			TST	@DATAND	DATA HEADER?
(1)	007134	001402				BEQ	TYPDAT	BR IF NO
(1)	007136	104402				TYPE		TYPE
(1)	007140	000000			DATAND:	0		DATA HEADER
(1)	007142	005737	007152		TYPDAT:	TST	@DATABP	DATA TABLE?
(1)	007146	001402				BEQ	@RESREG	BR IF NO.
(1)	007150	104411				CONVRT		SHOW
(1)	007152	000000			DATABP:	0		DATA TABLE
(1)	007154	104410			RESREG:	RES05		RESTORE PROC REGISTERS
(1)	007156	122737	000001	001140	HALTS:	CHPB	@APTENV,@ENV	IS APT RUNNING?
(1)	007164	001007				BNE	151	SKIP APT CALL IF NOT
(1)	007166	115737	001260	007200		MOVW	@ITEMB,51	COPY ERROR NUMBER
(1)	007174	004737	005354			JSR	@PC,@ATYA	CALL APT SERVICE
(1)	007200	000000			51:	.WORD	0	ERROR NUMBER STUCK HERE

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-34
 CVDZBD.P11 21-AUG-84 08:28 ABORT ROUTINE FOR LCP/ORION UFD MODE

(1)	007202	000777			10:	BR	10:	LOCK UP HERE
(1)	007204	022757	004420	000042	15:	CMP	#1ENDAD, #042	!CHECK TO SEE IF IN ACT-11 MODE
(1)	007212	001403				BEQ	20:	!IF SO, HANDLE ACCORDINGLY
(1)	007214	005777	172064			TST	BSMR	!HALT ON ERROR?
(1)	007220	100006				BPL	EXITER	!BR IF NO HALT ON ERROR
(1)	007222	016677	000002	172056	20:	MOV	2(SP), #DISPLAY	!SHOW ERROR PC IN DATA DISPLAY
(1)	007230	004737	006612			JSR	PC, ABORT	!CHECK IF WE ARE UNDER UFD?
(1)	007234	000000				HALT		!HALT
(1)	007236	005237	001256		EXITER:	INC	#ERTTL	!UPDATE ERROR COUNT
(1)	007242	004737	007322			JSR	PC, SERV.G	!FIND OUT IF 'G WAS TYPED
(1)	007246	032777	000400	172030		BIT	#SW08, BSMR	!GOTO TOP OF TEST?
(1)	007254	001007				BNE	1:	!BR IF YES
(1)	007256	032777	002000	172020		BIT	#SW10, BSMR	!GOTO NEXT TEST?
(1)	007264	001407				BEQ	2:	!BR IF NO
(1)	007266	013737	001362	001252		MOV	NEXT, #LPADR	!SET FOR NEXT TEST
(1)	007274	012706	001120		1:	MOV	#STACK, SP	!RESET SP
(1)	007300	000177	171746			JMP	#LPADR	!GOTO SPECIFIED TEST
(1)	007304	000002			2:	RTI		!RETURN
(1)	007306	000001			ERTABO:	1		
(1)	007310	006	002			.BYTE	6.2	
(1)	007312	001404				SAVPC		
(1)	007314	000001			XTSTN:	1		
(1)	007316	002	002			.BYTE	2.2	
(1)	007320	001246				#TSTN#		
(1)	007322	017746	171764		SERV.G:	MOV	#TKB, -(SP)	!OTHERWISE, GET THE LAST CHARACTER TYPED
(1)	007326	042716	000200			BIC	#BIT7, (SP)	!STRIP PARITY(EIGHTH) BIT
(1)	007332	122726	000007			CHPB	#7, (SP)	!IS IT 'G?
(1)	007336	001076				BNE	6:	!IF NOT, IGNORE INPUT
(1)	007340	032777	004000	171742		BIT	#4000, #TKS	!RX BUSY?
(1)	007346	001365				BNE	SERV.G	!BR IF YES
(1)	007350	007350			GETSMR=	.		!GPA
(1)	007350	017737	171730	007556		MOV	BSMR, #90	!SAVE (SMR).
(1)	007356	104402	007536		1:	TYPE	.#90	!TYPE HEADER FOR OLD SWITCH REGISTER
(1)	007362	104412	007550			CHVRT	.#80	!TYPE THE NUMBER ITSELF
(1)	007366	104402	007560			TYPE	.#90	!AFTER HAVING CONVERTED IT TO ASCII
(1)	007372	105037	007564			CLRB	92:	!CLEAR SMR CHANGE FLAG
(1)	007376	005077	171702			CLR	BSMR	!CLEAR THE SOFTWARE SWITCH REGISTER
(1)	007402	105777	171702		3:	TSTB	#TKS	!WAIT FOR DONE.
(1)	007406	100375				BPL	3:	!CONTINUE WAITING FOR IT
(1)	007410	017746	171676			MOV	#TKB, -(SP)	!PUT THE CHARACTER ON THE STACK
(1)	007414	042716	000200			BIC	#BIT7, (SP)	!STRIP PARITY BIT
(1)	007420	122726	000015			CHPB	#15, (SP)	!IS IT THE CARRIAGE RETURN CHAR?
(1)	007424	001433				BEQ	4:	!IF SO, GO PRINT CRLF
(1)	007426	105777	171662		2:	TSTB	#TPS	!IS THE OUTPUT BUFFER AVAILABLE
(1)	007432	100375				BPL	2:	!IF NOT, WAIT FOR IT TO BE READY
(1)	007434	105237	007564			INCB	92:	!INDICATE THAT THE SMR WAS CHANGED
(1)	007440	014677	171652			MOV	-(SP), #TPB	!PLACE THE CHARACTER THERE(ECHO BACK)
(1)	007444	000241				CLC		!GET READY TO ROTATE
(1)	007446	006177	171632			ROL	BSMR	!MOVE THE EXISTING BITS OVER
(1)	007452	006177	171626			ROL	BSMR	!TO MAKE ROOM FOR THE INCOMING
(1)	007456	006177	171622			ROL	BSMR	!THREE BITS FROM THIS CHARACTER
(1)	007462	103735				BCS	1:	!ERROR
(1)	007464	022627	000060			CMP	(SP), #60	!IS IT LOWER THAN 0?
(1)	007470	002732				BLT	1:	!IF SO, GO ASK AGAIN
(1)	007472	026627	177776	000067		CMP	-2(SP), #67	!IS IT HIGHER THAN 7?
(1)	007500	003326				BGT	1:	!IF SO, GO ASK AGAIN

CVDZB-D MACY11 30A(1052) 21 AUG-84 08:31 PAGE 26-35
 CVDZB0.P11 21-AUG-84 08:28 ABORT ROUTINE FOR LCP/ORION UFD MODE

```

(1) 007502 042746 177770          BIC    @+C<7>,-(SP)    ;ISOLATE INFORMATION BITS
(1) 007506 052677 171572          BIS    (SP)+,BSMR     ;ADD THEM TO THE SWITCH REGISTER
(1) 007512 000733                   BR     3#              ;GO CHECK FOR THE NEXT CHARACTER
(1) 007514 105737 007564          4#:   TSTB 92#        ;HAS THE SMR BEEN CHANGED?
(1) 007520 001003                   BNE   5#              ;IF YES GO TYPE CALF
(1) 007522 013777 007556 171554   MOV   90#,BSMR       ;IF NOT RESTORE SMR
(1) 007530 104402 001357          5#:   TYPE ,#CRLF     ;TYPE A CARRIAGE RETURN AND LINE FEED
(1) 007534 000207          6#:   RTS    PC       ;RETURN TO CALLING PROCEDURE
(1)
(1) 007536 020200 051450 051127   89#:  .ASCIZ <200>? (SMR)=/?
(1) 007544 036451 000057
(1)
(1) 007550 0000J1                   .EVEN
(1) 007552    006    000          88#:  1
(1) 007554 007556                   .BYTE 6,0
(1) 007556 000000                   90#:  .WORD 0
(1) 007560 036457 000057          91#:  .ASCIZ ?/=/?
(1) 007564    000          92#:  .BYTE 0
(1) 007566
(2)
(2)
(3)
(2)
;*****
;POWER DOWN ROUTINE
(2) 007566 012737 007732 000024   $PWRDN: MOV    @#ILLUP,@#PWRVEC ;SET FOR FAST UP
(2) 007574 012737 000340 000026   MOV    @340,@#PWRVEC+2 ;PRIO:7
(4) 007602 010046                   MOV    R0,-(SP)       ;PUSH R0 ON STACK
(4) 007604 010146                   MOV    R1,-(SP)       ;PUSH R1 ON STACK
(4) 007606 010246                   MOV    R2,-(SP)       ;PUSH R2 ON STACK
(4) 007610 010346                   MOV    R3,-(SP)       ;PUSH R3 ON STACK
(4) 007612 010446                   MOV    R4,-(SP)       ;PUSH R4 ON STACK
(4) 007614 010546                   MOV    R5,-(SP)       ;PUSH R5 ON STACK
(4) 007616 017746 171462           MOV    BSMR,-(SP)     ;PUSH BSMR ON STACK
(2) 007622 010637 007736           MOV    SP,#SAVR6      ;SAVE SP
(2) 007626 012737 007640 000024   MOV    @#PWRUP,@#PWRVEC ;SET UP VECTOR
(2) 007634 000000                   HALT
(2) 007636 000776                   BR     -2             ;HANG UP
(2)
(3)
;*****
;POWER UP ROUTINE
(2) 007640 012737 007732 000024   $PWRUP: MOV    @#ILLUP,@#PWRVEC ;SET FOR FAST DOWN
(2) 007646 013706 007736           MOV    #SAVR6,SP      ;GET SP
(2) 007652 005037 007736           CLR    #SAVR6         ;WAIT LOOP FOR THE TTY
(2) 007656 005237 007736          1#:   INC    #SAVR6       ;WAIT FOR THE INC
(2) 007662 001375                   BNE   1#              ;OF WORD
(4) 007664 012677 171414           MOV    (SP)+,BSMR     ;POP STACK INTO BSMR
(4) 007670 012605                   MOV    (SP)+,R5       ;POP STACK INTO R5
(4) 007672 012604                   MOV    (SP)+,R4       ;POP STACK INTO R4
(4) 007674 012603                   MOV    (SP)+,R3       ;POP STACK INTO R3
(4) 007676 012602                   MOV    (SP)+,R2       ;POP STACK INTO R2
(4) 007700 012601                   MOV    (SP)+,R1       ;POP STACK INTO R1
(4) 007702 012600                   MOV    (SP)+,R0       ;POP STACK INTO R0
(2) 007704 012737 007566 000024   MOV    @#PWRDN,@#PWRVEC ;SET UP THE POWER DOWN VECTOR
(2) 007712 012737 000340 000026   MOV    @340,@#PWRVEC+2 ;PRIO:7
(2) 007720 104402                   TYPE                                     ;REPORT THE POWER FAILURE
(2) 007722 007740          $PWRMG: .WORD  MPFAIL ;POWER FAIL MESSAGE POINTER
(2) 007724 012716                   MOV    (PC)+,(SP)    ;RESTART AT RESTART

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-36
 CVDZB0.P11 21-AUG-84 08:28 POWER DOWN AND UP ROUTINES

```

(2) 007726 011270          #PWRAD: .WORD RESTART          ;;RESTART ADDRESS
(2) 007730 000002          RTI
(2) 007732 000000          #ILLUP: HALT                ;;THE POWER UP SEQUENCE WAS STARTED
(2) 007734 000776          BR -.2                      ;; BEFORE THE POWER DOWN WAS COMPLETE
(2) 007736 000000          #SAVR6: 0                   ;;PUT THE SP HERE
(2) 007740 050200 051127 043040 #PFAIL: .ASCIZ <200>/PWR FAILED. RESTART AT LAST TEST /
(2) 010003 200 047105 020104 #EPASS: .ASCIZ <200>/END PASS CVDZB-D /
(2) 010027 200 052522 047116 #R: .ASCIZ <200>/RUNNING /
(2) 010043 200 051120 043517 #ERR2: .ASCIZ <200>/PROGRAM INDICATES NO DEVICES PRESENT./
(2) 010112 044600 051516 043125 #ERR3: .ASCIZ <200>/INSUFFICIENT DATA!/
(2) 010136 046200 041517 020113 #LOCK: .ASCIZ <200>/LOCK ON SELECTED TEST/
(2) 010165 103 051123 020072 #CSR: .ASCIZ /CSR: /
(2) 010173 126 041505 020072 #VEC: .ASCIZ /VEC: /
(2) 010201 120 051501 042523 #PASSX: .ASCIZ /PASSES: /
(2) 010212 051105 047522 051522 #ERRX: .ASCIZ /ERRORS: /
(2) 010223 124 051505 020124 #TSTN: .ASCIZ /TEST NO: /
(2) 010235 052 000040 #ASTEX: .ASCIZ /* /
(2) 010240 052200 050131 020105 #MEM: .ASCIZ <200>/TYPE A BIT MAP OF DZV11'S DESIRED ACTIVE: /
(2) 010315 120 035103 000040 #ERRPC: .ASCIZ /PC: /
(2) 010322 046600 050101 047440 #HEAD: .ASCIZ <200>/MAP OF DZV11 STATUS/<200>
(2) 010350 044600 046114 043505 #BADLN: .ASCIZ <200>/ILLEGAL ENTRY IN STAGGERED MODE/<200>
(2) .EVEN
(2) 010412 000002          XSTATQ: 2
(2) 010414 006 003 .BYTE 6.3
(2) 010416 001344 #TMP1
(2) 010420 006 002 .BYTE 6.2
(2) 010422 001346 #TMP2
(1) .EVEN
(2) ; THIS ROUTINE ESTABLISHES WHICH MAINTENANCE MODE THE DEVICE IS IN
(2) ;-----
(2) ;E=EXTERNAL LOOP BACK
(2) ;I=INTERNAL LOOP BACK
(2) ;S=STAGGERED LOOP BACK
(2) 010424 017605 000000 .SETFLG: MOV @SP,R5 ;PICK UP ADDRESS OF TAG
(2) 010430 042737 000040 010620 BIC #40,INBUF ;STRIP LOWER CASE
(2) 010436 122737 000105 010620 CMPB #'E,INBUF ;IS IT EXTERNAL LOOP BACK ?
(2) 010444 001005 BNE 4; ;NO
(2) 010446 013715 010536 MOV 1#,(R5) ;YES STORE INFO
(2) 010452 105037 001424 CLRB MNTFLG ;SET MAINT BIT =0
(2) 010456 000422 BR 7; ;GET OUT
(2) 010460 122737 000111 010620 4;: CMPB #'I,INBUF ;IS IT INTERNAL LOOP BACK ?
(2) 010466 001006 BNE 5; ;NO
(2) 010470 013715 010540 MOV 2#,(R5) ;YES STORE INFO
(2) 010474 112737 000010 001424 MOVB #MAINT,MNTFLG ;SET UP THE MAINTENANCE FLAG LOADER
(2) 010502 000410 BR 7; ;GET OUT
(2) 010504 122737 000123 010620 5;: CMPB #'S,INBUF ;IS IT STAGGERED LOOP BACK ?
(2) 010512 001007 BNE 6; ;WHAT ?
(2) 010514 013715 010542 MOV 3#,(R5) ;YES STORE INFO
(2) 010520 105037 001424 CLRB MNTFLG ;ZERO BITS
(2) 010524 062716 000002 7;: ADD #2,(SP) ;POP AROUND
(2) 010530 000002 RTI
(2) 010532 104404 6;: INSTER ;RETRY
(2) 010534 000733 BR .SETFLG ;DITTO
(2) 010536 000200 1;: .WORD 200 ;EXTERNAL = E
(2) 010540 000000 2;: .WORD 0 ;INTERNAL = I
(2) 010542 100000 3;: .WORD 100000 ;STAGGERED = S
    
```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-37
CVDZBO.P11 21-AUG-84 08:28 POWER DOWN AND UP ROUTINES

(2)

CVDZB-D MACY11 30A(1052) 21 AUG-84 08:31 PAGE 26-38
CVDZB0.P11 21-AUG-84 08:28 POWER DOWN AND UP ROUTINES

(2) ;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
(2) ;BUFFER TO THE CHARACTERS "E" AND "C".
(2) ;IF THE CHARACTER IS "E" CLEAR THE FLAG
(2) ;IF THE CHARACTER IS "C" SET THE FLAG

(2) 010544 017605 000000 .PAWCH:MOV B(SP),R5
(2) 010550 142737 000040 010620 BICL #40,INBUF ;SET FOR LOWER CASE INPUT
(2) 010556 122737 000105 010620 CMPB #'E,INBUF ;IS IT "E" ?
(2) 010564 001002 SNE 1#
(2) 010566 105015 CLRB (R5) ;000
(2) 010570 000406 BR 2#
(2) 010572 122737 000103 010620 1#: CMPB #'C,INBUF ;IS IT "C" ?
(2) 010600 001005 SNE 3#
(2) 010602 112715 177777 MOVB #-1,(R5) ;3177
(2) 010606 062716 000002 2#: ADD #2,(SP)
(2) 010612 000002 RTI
(2) 010614 104404 3#: INSTER ;RETRY
(2) 010616 000752 BR .PAWCH

(2) ;BUFFERS FOR INPUT-OUTPUT

(2) 010620 000000 INBUF: 0
(2) 010662 .+.40
(2) ; TEMP: 0 ; TEMP AREA UNUSED. ;:GPA
(2) ; .+.40 ; DELETED TO CONSERVE SPACE. ;:GPA
(2) 010662 000000 MDATA: 0
(2) 010724 .+.40
(2)

```

(2)
(2)
(2)
(2)
(2)
(2)
(2)
(2)
(2)
(2)
(2) 010724 005737 001406          CYCLE: TST      DZVACTV      ;ARE ANY DZV11'S TO BE TESTED?
(2) 010730 001006                    BNE      14      ;BR IF OK.
(2) 010732 104402 010043          TYPE     ,MERR2   ;NO DZV11'S SELECTED!!
(2) 010736 004737 006612          JSR      PC,ABORT ;CHECK IF WE ARE UNDER UFD?
(2) 010742 000000                    HALT                                ;STOP THE SHOW.
(2) 010744 000776                    BR       .-2      ;DISQUALIFY CONT. SW.
(2) 010746 013737 004726 001354 14:  MOV      @MXCNT,@TIMES ;RESTORE THE NUMBER OF ITERATIONS TO MAKE
(2) 010754 033737 001412 001406    BIT      RUN,DZVACTV ;IS THIS ONE "ACTIVE"
(2) 010762 001017                    BNE      24      ;BR IF GOOD ONE FOUND.
(2) 010764 006137 001412          ROL      RUN      ;UPDATE POINTER
(2) 010770 005337 001412          ADC      RUN      ;CATCH CARRY FROM RUN
(2) 010774 062737 000012 001420    ADD     @12,ACTIVE ;UPDATE ADDRESS POINTER.
(2) 011002 022737 001740 001420    CMP     @DZV.END,ACTIVE ;HAVE WE PASSED THE END OF THE MAP?
(2) 011010 001356                    BNE      14      ;IF NO, KEEP GOING; NOT ALL TESTED FOR.
(2) 011012 012737 001500 001420    MOV     @DZV.MAP,ACTIVE ;RESET ADDRESS POINTER.
(2) 011020 000752                    BR       14      ;KEEP LOOKING FOR ACTIVE DZV11
(2) 011022 006137 001412          ROL      RUN      ;UPDATE POINTER.
(2) 011026 005337 001412          ADC      RUN      ;CATCH CARRY.
(2) 011032 013700 001420          MOV     ACTIVE,R0  ;GET ADDRESS POINTER.
(2) 011036 062737 000012 001420    ADD     @12,ACTIVE ;UPDATE.
(2) 011044 022737 001740 001420    CMP     @DZV.END,ACTIVE ;ALL DONE?
(2) 011052 001003                    BNE      34      ;BR IF NO.
(2) 011054 012737 001500 001420    MOV     @DZV.MAP,ACTIVE ;RESTORE POINTER.
(2) 011062 012037 001174          MOV     (R0),@BASE ;LOAD SYSTEM CTRL. REG
(2) 011066 012037 002040          MOV     (R0),@DZVRIV ;LOAD VECTOR
(2) 011072 012037 001366          MOV     (R0),@LINE  ;SET UP DZV LINES ACTIVE
(2) 011076 012037 001370          MOV     (R0),@PAR   ;SET UP PARAMETERIZATION
(2) 011102 012037 001372          MOV     (R0),@MODE  ;SET UP MAINTENANCE MODE
(2) 011106 105037 001424          CLRB    MNTFLG ;RESET MAINT. FLAG IF
(2) 011112 005737 001372          TST     MODE      ;RUNNING TESTS
(2) 011116 001003                    BNE      94      ;IN
(2) 011120 112737 000010 001424    MOVB   @MAINT,MNTFLG ;INTERNAL MAINT. MODE
(2) 011126 004737 011274          JSR     PC,DZVLEV  ;SET UP
(2) 011132 005737 000042          TST     @M42      ;ARE WE UNDER MONITOR CONTROL?
(2) 011136 001051                    BNE      74      ;IF YES, SKIP THIS SETUP
(2) 011140 032777 000002 170136    BIT     @SW01,BSWR ;IF SW01=1, GET STARTING TEST #
(2) 011146 001445                    BEQ     74      ;BR IF NO TEST IS TO BE INPUTTED
(2) 011150 104402 001357          TYPE     ,@CRLF
(3) 011154 104403                    INSTR
(3) 011156 010223                    HTSTN
(3) 011160 104405                    PARAM
(3) 011162 000001                    1
(3) 011164 001000                    1000
(3) 011166 001246                    @TSTNM
(3) 011170 000                                .BYTE 0
(3) 011171 001                                .BYTE 1
    ;CALL THE STRING INPUT ROUTINE
    ;POINTER TO MESSAGE TO BE PRINTED
    ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
    ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
    ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
    ;POINTER TO MAP LOCATION TO BE FILLED
    ;MASK OF INVALID BITS FOR THIS PARAMETER
    ;NUMBER OF PARAMETERS TO STORE
    
```

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-40
 CVDZB.D.P11 21-AUG-84 08:28 POWER DOWN AND UP ROUTINES

```

(2) 011172 0127C0 012274      MOV    @TST1,RO
(2) 011176 022710 000004      50:   CMP    #4,(RO)
(2) 011202 001020              BNE    60
(2) 011204 022760 012737 000002  CMP    @12737,2(RO)
(2) 011212 001014              BNE    60
(2) 011214 023760 001246 000004  CMP    @TSTM,4(RO)      ;IS THIS THE TEST ?
(2) 011222 001010              BNE    60      ;IF NOT, DCN'T PROCESS NUMBER
(2) 011224 010037 001252      MOV    RO,#LPADR      ;SAVE PC
(2) 011230 062737 000002 001252  ADD    @2,#LPADR      ;POP OVER PREVIOUS SCOPE
(2) 011236 104402 001357      TYPE   ,#CRLF
(2) 011242 000412              BR     80
(2) 011244 005720      60:   TST    (RO)+
(2) 011246 020027 014426      CMP    RO,@TLAST+10
(2) 011252 001351              BNE    50
(2) 011254 104402 001356      TYPE   ,#QUES
(2) 011260 000733              BR     40
(2) 011262 012737 012274 001252  70:   MOV    @TST1,#LPADR    ;PREPARE TEST ADDRESS
(2) 011270              80:
(2) 011270 000177 167756      RESTART:JMP    #0LPADR    ;GO START TESTING...WARNING!****
(2)                                ;THIS .UMP IS USED BY POWER UP ROUTINE!!!!

(2)                                ;THIS UTILITY SETS UP CSR'S,SETS UP VECTORS.
(2) DZVLEV: MOV    DZVRIV,RO    ;PLACE THE BASE VECTOR ADDRESS IN RO
(2)      ADD    @2,RO        ;CALCULATE THE RECEIVER INTERRUPT STATUS ADDR.
(2)      MOV    RO,DZVRIS    ;STORE IT HERE
(2)      ADD    @2,RO        ;CALCULATE THE TRANSMITTER INTERRUPT VECTOR
(2)      MOV    RO,DZVTIV    ;STORE IT HERE
(2)      ADD    @2,RO        ;CALCULATE THE TRANSMITTER VECTOR STATUS ADDRESS
(2)      MOV    RO,DZVTIS    ;STORE IT HERE

(2)                                ;THIS SEGMENT SETS UP POINTERS FOR THE GIVEN DZV11. #BASE IS THE BASE ADDRESS
(2) ;OF THE DEVICE
(2)      MOV    #BASE,RO    ;COPY THE ADDRESS BEING LOADED
(2)      MOV    RO,DZVCSR    ;XXX0
(2)      INC    RO
(2)      MOV    RO,HDZVCSR    ;XXX1
(2)      INC    RO
(2)      MOV    RO,DZVRBUF    ;XXX2
(2)      MOV    RO,DZVLPR    ;XXX2
(2)      INC    RO
(2)      MOV    RO,HDZVRBUF    ;XXX3
(2)      MOV    RO,HDZVLPR    ;XXX3
(2)      INC    RO
(2)      MOV    RO,DZVTCR    ;XXX4
(2)      INC    RO
(2)      MOV    RO,HDZVTCR    ;XXX5
(2)      INC    RO
(2)      MOV    RO,DZVMSR    ;XXX6
(2)      MOV    RO,DZVTDR    ;XXX6
(2)      INC    RO
(2)      MOV    RO,HDZVMSR    ;XXX7
(2)      MOV    RO,HDZVTDR    ;XXX7
(2) 011432 000207      RTS    PC

```


CVDZB D MACY11 30A(1052) 21 AUG-84 08:31 PAGE 26-42
CVDZBD P11 21-AUG 84 08:28 POWER DOWN AND UP ROUTINES

(2)

; END OF .PARAM DELETE RANGE

E

;;GPA


```

(2) ;*ROUTINE USED TO SET UP THE DIAGNOSTIC VIA APT.
(2) ;*IF BIT7 IN THE ENVIRONMENT MODE (ENVM) BYTE IS SET.
(2) ;*THE PROGRAM WILL LOAD ITS PARAMETERS FROM THE ETABLE.
(2)
(2) 011436 012700 001500 SETAPT: MOV #DZV.MAP,R0 ;POINT TO THE DEVICE MAP TABLE
(2) 011442 013701 001174 MOV #BASE,R1 ;BUILD DEVICE ADDRESSES IN R1
(2) 011446 013702 001170 MOV #VECT1,R2 ;BUILD DEVICE VECTORS IN R2
(2) 011452 042702 177007 BIC #C<770>,R2 ;STRIP AWAY OTHER INFORMATION
(2) 011456 012704 001204 MOV #DDMO,R4 ;POINT TO THE BEGINNING OF DEVICE PARAMETERS
(2) 011462 013705 001176 MOV #DEVM,R5 ;GET THE MAP OF ACTIVE DEVICES
(2) 011466 105037 001414 CLRB DZVNUM ;INITIALIZE THE NO. OF ACTIVE DEVICES
(2) 011472 005037 001410 CLR SAVACTV ;CLEAR THE ACTIVE BIT MAP
(2) 011476 006005 14: ROR R5 ;GET A DEVICE SELECTION BIT
(2) 011500 103407 BCS 34 ;IF IT IS SELECTED, GO SFT UP A MAP
(2) 011502 001422 BEQ 54 ;IF NO MORE ARE SELECTED, GET OUT OF SETUP
(2) 011504 005724 TST (R4). ;POINT TO NEXT DEVICE DESCRIPTOR
(2) 011506 062701 000010 24: ADD #10,R1 ;SET UP THE NEXT ADDRESS
(2) 011512 062702 000010 ADD #10,R2 ;SET UP THE NEXT VECTOR GROUP
(2) 011516 000767 BR 14 ;GO SEE IF MORE DEVICES REMAIN
(2) 011520 006137 001410 34: ROL SAVACTV ;SET BIT IN ACTIVE DEVICE MAP
(2) 011524 105237 001414 INCB DZVNUM ;INCREMENT NO. OF ACTIVE DEVICES
(2) 011530 010120 MOV R1,(R0). ;LOAD DEVICE ADDRESS
(2) 011532 010220 MOV R2,(R0). ;LOAD THE VECTOR ADDRESS
(2) 011534 013720 001200 MOV #CDM1,(R0). ;GET THE NUMBER OF LINES IN OPERATION
(2) 011540 012420 MOV (R4).,(R0). ;LOAD DEVICE PARAMETERS
(2) 011542 013720 001202 MOV #CDM2,(R0). ;LOAD DEFAULT TESTING MODE
(2) 011546 000757 BR 24 ;GO BUILD THE NEXT ADDRESS
(2) 011550 012710 177777 54: MOV #0-1,(R0) ;TERMINATE THE DEVICE MAP
(2) 011554 012737 001142 001304 MOV #SMREG,SMR ;SET TO SOFTWARE APT SWITCH REGISTER
(2) 011562 000207 RTS PC ;RETURN TO PRINT STATUS TABLE
    
```

```

(2) ;*ROUTINE USED TO "AUTO SIZE" THE DZV11
(2) ;*CSR AND VECTOR.
(2) ;*NOTE: THE CSR MAY BE ANY WHERE IN THE FLOATING
(2) ;* ADDRESS RANGE (160000:167770)
(2) ;* AND THE VECTOR MAY BE ANY WHERE IN THE
(2) ;* FLOATING VECTOR RANGE (300:770)
(2) ;*
    
```

```

(2) 011564 AUTO.SIZE:
(2) 011564 000005 RESET ;INSURE A BUS INIT.
(2) 011566 105337 001422 DECB INIFLG ;SHOW THAT I WAS HERE
(2) 011572 012702 001500 CSRMAP: MOV #DZV.MAP,R2 ;LOAD MAP POINTER.
(2) 011576 012703 001204 MOV #DDMO,R3 ;POINT TO ETABLE DEVICE DESCRIPTOR WORDS
(2) 011602 005022 14: CLR (R2). ;ZERO ENTIRE MAP
(2) 011604 022702 001740 CMP #DZV.END,R2 ;ALL DONE?
(2) 011610 001374 BNE 14 ;BR IF NO
(2) 011612 105037 001414 CLRB DZVNUM ;SET OCTAL NUMBER OF DZV11'S TO 0
(2) 011616 012702 001500 MOV #DZV.MAP,R2
(2) 011622 012701 160000 MOV #160000,R1 ;SET FOR FIRST ADDRESS TO BE TESTED
(2) 011626 012737 012076 000004 MOV #61,804 ;SET FOR NON-EXISTENT DEVICE TIME OUT
(2) 011634 052711 000040 24: BIS #BITS,(R1) ;TRY TO SET MASTER SCAN ENABLE
(2) 011640 052761 000017 000004 BIS #17,4(R1) ;TRY TO TRANSMIT ON ANY LINE
(2) 011646 005000 CLR R0 ;USE R0 AS A COUNTER
(2) 011650 005711 74: TST (R1) ;HAS TRANSMITTER READY COME UP?
    
```


8461
8462
8463
8464
8465
8466
8467
8468
8469
8470
8471
8472
8473
8475
(5)
(4)
(2)
(2)
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

012274 000004
012276 012737 000001 001246
012304 012737 012736 001362
012312 012737 012652 001364
012320 104417
012322 104421
012324 005037 001374
012330 104422
012332 012702 000001
012336 032777 010040 167444
012344 030237 001366
012350 001533
012352 013700 001374
012356 006300
012360 010277 167440
012364 105777 167420
012370 100001
012372 104020
012374 005003
012376 005004
012400 005777 167404
012404 100404
012406 104414
012410 005204
012412 001372
012414 104003
012416 116077 001426 167410
012424 005260 001426
012430 020327 000017
012434 103006
012436 032777 020000 167344
012444 001413
012446 104013
012450 000411
012452 005004
012454 032777 020000 167326
012462 001004
012464 104414
012466 005204
012470 001371

```
***** TEST 1 *****  
; THIS TEST VERIFIES OVERRUN AND SILO ALARM  
; ONE LINE AT A TIME - BASED UPON VALID LINES  
; AS EACH OF THE FIRST 16 CHARS ARE SENT, SILO ALARM IS  
; TESTED TO BE CLEARED. ON THE 16TH CHAR THE PROGRAM THEN  
; EXPECTS SILO ALARM TO SET. THEN THE ENTIRE  
; SILO IS FILLED AND AN OVERRUN IS EXPECTED ON THE 65TH  
; CHAR PULLED OUT OF THE SILO.  
; ERROR PRINTOUTS WILL REPORT TRANSMITTING LINE NO.  
; USING SWITCH NINE FOR THIS TEST SENDS 20. CHARACTERS  
; ON DZV LINE PREVIOUSLY SELECTED CONTINUOUSLY WHILE SW09=1.  
; USED TO SCOPE SILO ALARM PULSES, ETC.  
  
:: TEST 1  
;*****  
TST1: SCOPE  
MOV 01,ITSTNM ;LOAD THE NUMBER OF THIS TEST  
MOV 0TST2,NEXT ;POINT TO THE START OF THE NEXT TEST  
MOV 0180,LOCK ;SET FOR LOOP  
DCLASH ;SET DCLR IN CSR AND SET INTFLG  
LPRSET ;LOAD LINE PARAMETERS  
CLR SAVLIN ;INIT LINE INDICATOR  
BUFSET ;ZERO DATA BUFFER  
MOV 01,R2 ;LINE POINTER  
BIS 0MSENAB,SILOEN,0DZVCSR ;START SCANNER & SET SILO ENABLE  
34: BIT R2,LINE ;VALID LINE?  
BEQ 211 ;IF NOT GO TO NEXT LINE  
MOV SAVLIN,R0 ;MAKE OFFSET  
ASL R0 ;MAKE POWER OF TWO  
MOV R2,0DZVTCR ;SET TCR BIT  
44: TSTB 0DZVCSR ;REC DONE = 1 ?  
BPL .+4  
ERROR. 20 ;REC DONE SHOULD NOT = 1  
CLR R3 ;SET CHARACTER COUNT  
54: CLR R4  
64: TST 0DZVCSR ;IS TRDY SET?  
BPI 74 ;IF YES, LOAD CHAR.  
DELAY ;WAIT FOR TRDY TO SET  
INC R4 ;INC DELAY COUNTER  
BNE 64  
ERROR. 3 ;TRDY FAILED TO SET  
74: MOVE TDC(R0),0DZVTCR ;LOAD A CHARACTER  
INC TDC(R0) ;SET UP NEXT CHARACTER  
CMP R3,015. ;16 CHARACTERS ?  
BMS 84  
BIT 0SILOAL,0DZVCSR ;SILO ALARM = 0 ?  
BEQ 104 ;YES  
ERROR. 13 ;SILO ALARM SHOULD NOT = 1  
UNTIL 16. DATA CHARACTERS  
  
BR 104  
84: CLR R4  
94: BIT 0SILOAL,0DZVCSR  
BNE 104  
TNC R4  
BNE 94
```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-47
CVDZBD.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP CORP

```

8514 012472 104014          ERROR. 14          ;SILO ALARM FAILED TO SET!
8515                                ;SILO ALARM SHOULD =1 AFTER 16.
8516                                ;DATA CHARACTERS
8517 012474 005203          100:  INC      R3          ;INC CHAR COUNT
8518 012476 022703 000102    CMP      @66.,R3      ;FINISHED SENDING CHARACTERS ?
8519 012502 001335          BNE     S1          ;NO
8520 012504 005004          CLR     R4
8521 012506 104414          DELAY
8522 012510 105204          INCB   R4
8523 012512 001375          BNE     -4
8524                                ;NOW LETS READ THE SILO
8525 012514 013705 001374    MOV     SAVLIN,R5     ;MAKE EXPECTED LINE #
8526 012520 005737 001372    TST    MODE          ;IS THIS TEST IN STAGGERED MODE?
(1) 012524 100006          BPL     130          ;IF NOT, SKIP STAGGERED SETUP
(1)
(1)
(1)
(1) 012526 006205          ASR     R5          ;GET THE LAST BIT INTO THE CARRY BIT
(1) 012530 103402          BCS    110          ;IF IT IS SET, GO CLEAR IT
(1) 012532 000261          SEC
(1) 012534 000401          BR     120          ;IF IT IS CLEAR SET IT HERE
(1) 012536 000241          110:  CLC
(1) 012540 006105          120:  ROL     R5          ;SKIP THE CLEARING
8527 012542 000305          130:  SWAB   R5          ;CLEAR THE CARRY BIT (INVERSION OF LINE PARITY)
8528 012544 052705 100000    BIS    @INVALID,R5   ;GET THE NEW BIT BACK INTO R5
8529 012550 017704 167240    140:  MOV     @DZVRBUF,R4 ;PUT IN UPPER BYTE
6 30 012554 020405          CMP     R4,R5        ;ADD DATA VALID
8531 012556 001401          BEQ    150          ;ACTUAL
8532 012560 104006          ERROR. 6          ;ACTUAL VS. EXPECTED
8533 012562 032777 020000 167220 150:  BIT     @SILOAL,@DZVCSR ;DATA/CONTENTS DID NOT COMPARE
8534 012570 001401          BEQ    160          ;SILO ALARM= 0 ?
8535 012572 104016          ERROR. 16         ;YES
8536 012574 005203          160:  INC     R5          ;READING DZVRBUF DID NOT CLEAR SILO ALARM
8537 012576 120527 000077    CMPB   R5,@63.       ;UP CHARACTER
8538 012602 101762          BLOS   140          ;LAST SILO CHAR ?....64TH CHAR
8539 012604 005203          INC     R5          ;ADD 1 MORE FOR THE CLOBBERED CHAR
8540 012606 052705 040000    BIS    @OVRUN,R5     ;ADD OVERRUN TO EXPECTED
8541 012612 120527 000101    CMPB   R5,@65.       ;LAST CHARACTER ?
8542 012616 001754          BEQ    140
8543 012620 017704 167170    MOV     @DZVRBUF,R4 ;FOR GOOD MEASURE
8544 012624 005704          TST    R4          ;DATA VALID SHOULD = 0
8545 012626 100001          BPL    170          ;YES
8546 012630 104017          ERROR. 17         ;DATA VALID SHOULD = 0
8547 012632 040277 167166    170:  BIC    R2,@DZVTCR   ;CLR TCR BIT
8548 012636 104401          SCOP1 ;LOOP?
8549 012640 005237 001374    210:  INC     SAVLIN      ;INC EXPECTED LINE
8550 012644 104420          SHIFT ;NEXT LINE
8551 012646 000137 012344    JMP     30          ;YES
8552
8553                                ;TIGHT SCOPE LOOP FOR THIS TEST. SENDS 20. CHARACTERS
8554                                ;ON DZV LINE PREVIOUSLY SELECTED CONTINUOUSLY WHILE SW09=1.
8555                                ;USED TO SCOPE SILO ALARM PULSES, ETC.
8556
8557 012652 052777 010040 167130 180:  BIS    @SENAB!SILOEN,@DZVCSR ;SETUP DEVICE
8558 012660 012777 012726 167156    MOV     @20,@DZVTIV  ;SETUP TRANSMITTER VECTOR
8559 012666 012701 000024          MOV     @20.,R1     ;TEMPORARY COUNT OF CHARACTER BURST

```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-48
 CVDZBD.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP CORP

8560	012672	050277	167126			BIS	R2, BDZVTCR	ENABLE LINE
8561	012676	052777	040000	167104		BIS	#TIE, BDZVCSR	ENABLE INTERRUPTS
8562	012704	106427	000000			MTPS	#0	LOWER PRIORITY
8563	012710	000001			190:	WAIT		ALLOW INTERRUPTS
8564	012712	077102				SQB	R1, 190	REDUCE COUNT. ALL CHARACTERS SENT?
8565	012714	042777	050040	167066		BIC	#SILOEN!#SENAB!TIE, BDZVCSR	RESET SILO COUNTER, CLEAR STROBE
8566	012722	104401				SCOPI		LOOP AGAIN?
8567	012724	000742				BR	170	IF NOT, RETURN TO WHERE YOU LEFT OFF
8568	012726	112777	000252	167100	200:	MOVB	#252, BDZVTCR	SEND A CHARACTER
8569	012734	000002				RTI		ALLOW MORE CHARACTERS TO COME
8570								***** TEST 2 *****
8571								*THIS TEST THAT "SILO ENABLE" WILL INHIBIT
8572								*RECEIVER INTERRUPTS AND THAT ON THE
8573								*16TH CHAR THAT "SILO ALARM" WILL CAUSE AN
8574								*INTERRUPT WITH "RIE" SET.
8575								*THIS WILL DO ALL SELECTED LINES ONE AT A TIME.
8576								*ERROR PRINTOUTS WILL REPORT TRANSMITTING LINE NO.
8578								*** TEST 2
(5)								*****
(4)	012736	000004				TST2:	SCOPE	
(2)	012740	012737	000002	001246		MOV	#2, #T:STN	LOAD THE NUMBER OF THIS TEST
(2)	012746	012737	013236	001362		MOV	#TST3, NEXT	POINT TO THE START OF THE NEXT TEST
8579	012754	012737	013000	001364		MOV	#30, LOCK	SET FOR LOOP
8580	012762	104417				DCLASH		SET DCLR IN CSR AND SET INTFLG
8581	012764	104421				LPRSET		LOAD LINE PARAMETERS
8582	012766	005037	001374			CLR	SAVLIN	INIT LINE INDICATOR
8583	012772	104422				BIFSET		ZERO DATA BUFFER
8584	012774	012702	000001			MOV	#1, R2	LINE POINTER
8585	013000	012777	013210	167032	30:	MOV	#110, BDZVRIV	SET FOR UNEXPECTED INTER.
8586	013006	012777	000200	167026		MOV	#MASK, BDZVRIS	SET PRIO.
8587	013014	052777	010140	166766		BIS	#SENAB!SILOEN!RIE, BDZVCSR	START SCANNER & SET SILO ENABLE
8588								VALID LINE?
8589	013022	030237	001566			BIT	R2, LINE	
8590	013026	001477				BEG	180	IF NOT GO TO NEXT LINE
8591	013030	005777	166760			TST	BDZVRBUF	EMPTY THE SILO
8592	013034	100775				BMI	.-4	OR IF DATA VALID IS SET!
8593	013036	106427	000000			MTPS	#0	SET PROCESSOR PRIORITY TO 0
8594	013042	013700	001374			MOV	SAVLIN, R0	MAKE OFFSET
8595	013046	006300				ASL	R0	MAKE POWER OF TWO
8596	013050	010277	166750			MOV	R2, BDZVTCR	SET TCR BIT
8597	013054	005004			50:	CLR	R4	
8598	013056	005777	166726		60:	TST	BDZVCSR	
8599	013062	100404				BMI	70	
8600	013064	104414				DELAY		
8601	013066	005204				INC	R4	
8602	013070	001372				BNE	60	
8603	013072	104003				ERROR.	3	*TRY FAILED TO SET
8604	013074	116077	001426	166732	70:	MOVB	TDO(R0), BDZVTCR	LOAD A CHARACTER
8605	013102	005260	001426			INC	TDO(R0)	SET UP NEXT CHARACTER
8606	013106	022760	000017	001426		CMP	#15, TDO(R0)	15 CHARS YET?
8607	013114	001406				BEG	80	
8608	013116	032777	020000	166664		BIT	#SILOAL, BDZVCSR	SILO ALARM = 0 ?
8609	013124	001401				BEG	.-4	YES
8610	013126	104013				ERROR.	13	*SILO ALARM SHOULD NOT = 1
8611								UNTIL 16. DATA CHARACTERS
8612	013130	000752				BR	60	

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 26-49
 CVDZBD.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS.

COPYRIGHT 1977 DIGITAL EQUIP CORP

8613	013132	012777	013216	166700	81:	MOV	#128, BDZVRIV	!SET NEW VECTOR
8614	013140	005777	166644			TST	BDZVCSR	!READY FOR 16TH CHAR
8615	013144	100375				BPL	.-4	
8616	013146	016077	001426	166660		MOV	TDO(R0), BDZVTDR	!LOAD THE 16TH CHAR.
8617	013154	005004				CLR	R4	
8618	013156	032777	020000	166624	91:	BIT	#SILOAL, BDZVCSR	
8619	013164	001005				BNE	101	
8620	013166	104414				DELAY		
8621	013170	005204				INC	R4	
8622	013172	001371				BNE	91	
8623	013174	104014				ERROR.	14	!SILO ALARM FAILED TO SET!
8624	013176	000410				BR	171	!SILO ALARM SHOULD =1 AFTER 16.
8625								!DATA CHARACTERS
8626	013200	000240			101:	NOP		!STALL
8627	013202	000240				NOP		
8628	013204	104027				ERROR.	27	!SILO ALARM NOT INTERRUPTING.
8629	013206	000404				BR	171	!CONTINUE TEST.
8630	013210	022626			111:	POP2SP		!FAKE RTI
8631	013212	104012				ERROR.	12	!RX SHOULD NOT INTERRUPT
8632	013214	000401				BR	171	!CONTINUE
8633	013216	022626			121:	POP2SP		!GOOD INTERRUPT TO HERE.
8634	013220	040277	166600		171:	BIC	R2, BDZVTCR	!CLR TCR BIT
8635	013224	104401				SCOPI		!LOOP?
8636	013226	005237	001374		181:	INC	SAVLIN	!INC EXPECTED LINE
8637	013232	104420				SHIFT		!NEXT LINE
8638	013234	000661				BR	31	!YES

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27
 CVDZBO P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP.

```

0640
0641
0642
0643
0644
0645
0647
(5)
(4) 013236 000004
(2) 013240 012737 000003 001246
(2) 013246 012737 014000 001362
0648 013254 104417
0649 013256 013737 001366 013776
0650 013264 013737 001366 013516
0651 013272 104421
0652 013274 104422
0653 013276 012777 013520 166534
0654 013304 012777 000200 166530
0655 013312 012777 013404 166524
0656 013320 012777 000200 166520
0657 013326 052777 040140 166454
0658 013334 113777 001366 166462
0659 013342 106427 000000
0660
0661
0662 013346 005037 013402
0663 013352 104414
0664 013354 105737 013776
0665 013360 001002
0666 013362 000137 013724
0667 013366 005237 013402
0668 013372 001367
0669 013374 104007
0670 013376 104011
0671 013400 104400
0672 013402 000000
0673
0674
0675 013404 117703 166402
0676 013410 042703 177774
0677 013414 010304
0678 013416 010337 001374
0679 013422 005777 166362
0680 013426 100401
0681 013430 104003
0682 013432 012702 000001
0683 013436 105303
0684 013440 100402
0685 013442 006302
0686 013444 000774
0687 013446 030237 001366
0688 013452 001001
0689 013454 104015
0690 013456 030237 013516
0691 013462 001003
0692 013464 040277 166334

;***** TEST 3 *****
;THIS TEST RUNS ALL LINES FULL BORE
;BASED UPON QUALIFIED LINES
;.. THIS IS AN INTERRUPT TEST ON THE RECEIVER AND
;TRANSMITTER
;: TEST 3
;*****
TST3: SCOPE
MOV #3, #TSTNM ;LOAD THE NUMBER OF THIS TEST
MOV #TST4, NEXT ;POINT TO THE START OF THE NEXT TEST
DCLASH ;SET DCLR IN CSR AND SET MNTFLG
MOV LINE, RXTCR ;SET IMAGE OF TCR BITS
MOV LINE, TXTCR ;SET IMAGE OF TCR BITS
LPRSET ;LOAD LINE PARAMETERS
BUFSET ;ZERO DATA BUFFER
MOV #RXSVC, #DZVRIV ;SET UP REC INTR VECTOR
MOV #MASK, #DZVRIS ;STATUS
MOV #TXSVC, #DZVTIV ;SET UP TRANS INTR VECTOR
MOV #MASK, #DZVTIS ;STATUS
BIS #SENAB!RIE!TIE, #DZVCSR ;SET MASTER SCAN ENABLE
MOVB LINE, #DZVTCR ;SET TCR BITS
HTPS #CLEAR ;ALLOW INTERRUPTS

SNAP: CLR 4# ;CLEAR DELAY COUNTER
2#: DELAY ;WAIT FOR RECEIVERS TO FINISH
TSTB RXTCR ;WAIT FOR ALL RECIEVERS TO FINISH
BNE 3#
JMP OUT
3#: INC 4# ;INCREMENT DELAY COUNTER
BNE 2# ;DELAY FINISHED?
ERROR# 7 ;*TRANSMITTER FAILED TO INTERRUPT
ERROR# 11 ;*RECEIVER FAILED TO INTERRUPT
ADVANCE ;LEAVE THIS TEST
4#: 0

;TRANS INTR SVC ROUTINE
TXSVC: MOVB #DZVCSR, R3 ;FIND LINE NO.
BIC #C<3>, R3 ;ISOLATE LINE NO.
MOV R3, R4 ;SAVE LINE NO.
MOV R3, SAVLIN ;SAVE LINE NO.
TST #DZVCSR ;TRANS READY SET ?
BMI .+4
ERROR# 3 ;*TRANSMITTER FAILED
MOV #1, R2 ;SET UP POSITION POINTER
3#: DECB R3 ;IS IT THIS LINE ?
BMI 4# ;YES
ASL R2 ;UP THE LINE #
BR 3# ;GO 'ROUND AGAIN
4#: BIT R2, LINE ;INVALID LINE ?
BNE .+4 ;YES
ERROR# 15 ;NO, INVALID LINE!!!!
BIT R2, TXTCR ;DATA FINISHED?
BNE 6# ;IF NOT SEND CHAR.
BIC R2, #DZVTCR ;CLEAR TCR BIT
    
```


CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-1
 CVDZBD.P11 21-AUG-84 08:28 DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP.

8693	013470	000411			BR	51		;RETURN
8694	013472	006304			ASL	R4		;MAKE POWER OF 2
8695	013474	116477	001426	166332	MOV	TDO(R4),BDZVTDR		;LOAD CHARACTER
8696	013502	105264	001426		INCB	TDO(R4)		;SET UP NEXT CHARACTER
8697	013506	001002			BNE	51		;LAST CHARACTER ?
8698	013510	040237	013516		BIC	R2,TXTCR		;INDICAT LINE FINISHED
8699	013514	000002			RTI			
8700								
8701	013516	000000			TXTCR:	0		
8702								
8703								
8704	013520	105777	166264		;REC INTR SVC ROUTINE			
8705	013524	100401			RXSVC: TSTB	BDZVCSR		;REC DONE ?
8706	013526	104004			BMI	.+4		;YES
8707	013530	032777	020000	166252	ERROR.	4		;FALSE INTERRUPT
8708	013536	001401			BIT	#SILOAL,BDZVCSR		;SILO ALARM?
8709	013540	104013			BEQ	.+4		;NO
8710	013542	017704	166246		ERROR.	13		;SILO ALARM SHOULD NOT =1
8711	013546	010403			MOV	BDZVRBUF,R4		;SAVE IT
8712	013550	000303			MOV	R4,R3		
8713	013552	042703	177774		SWAB	R3		
8714	013556	010337	001374		BIC	#C<3>,R3		;STRIP JUNK
8715	013562	005704			MOV	R3,SAVLIN		;SAVE LINE NUMBER
8716	013564	100401			TST	R4		;DATA VALID?
8717	013566	104023			BMI	41		;IF YES SKIP ERROR PRINTOUT
8718	013570	032704	040000		ERROR.	23		;YOU LOSE ...DATA VALID WAS'NT SET
8719	013574	001401			BIT	#OVRUN,R4		;TEST FOR OVERRUN
8720	013576	104024			BEQ	11		;IF NO OVERRUN SKIP ERROR
8721	013600	032704	020000		ERROR.	24		;DATA OVERRUN
8722	013604	001401			BIT	#FRMERR,R4		;DATA FRAMING ERROR
8723	013606	104025			BEQ	21		;IF NO FRAMING ERROR CONTINUE
8724	013610	032704	010000		ERROR.	25		;FRAMING ERROR
8725	013614	001401			BIT	#PARER,R4		;TEST FOR PARITY ERROR
8726	013616	104026			BEQ	31		;BRANCH IF NO ERROR
8727	013620	012702	000001		ERROR.	26		;TYPE OUT PARITY ERROR
8728	013624	105303			MOV	#1,R2		;SET UP POSITION POINTER
8729	013626	100402			DECB	R3		
8730	013630	006302			BMI	61		;RE POSITION POINTER
8731	013632	000774			ASL	R2		;GO 'ROUND AGAIN
8732	013634	030237	001366		BR	51		;LINE VALID ?
8733	013640	001001			BIT	R2,LINE		
8734	013642	104015			BNE	.+4		;YES
8735	013644	013703	001374		ERROR.	15		;INVALID LINE #
8736	013650	006303			MOV	SAVLIN,R3		;GET THE LINE NUMBER AGAIN
8737	013652	126304	001436		ASL	R3		;USE R3 AS A POINTER IN THE DATA TABLE
8738	013656	001410			CMPE	TRO(R3),R4		;DOES THE DATA CHARACTER COMPARE ?
8739	013660	013705	001374		BEQ	71		;YES
8740	013664	000305			MOV	SAVLIN,R5		;MOVE LINE NO INTO EXPECTED
8741	013666	052705	100000		SWAB	R5		;ADJUST TO HIGH BYTE
8742	013672	056305	001436		BIS	#DVALID,R5		;SET DVALID IN EXPECTED
8743	013676	104005			BIS	TRO(R3),R5		;SET DATA IN EXPECTED
8744	013700	005263	001436		ERROR.	5		;NO, DATA DOES NOT COMPARE
8745	013704	105763	001436		71:	INC	TRO(R3)	;SET UP FOR NEXT CHARACTER
8746	013710	001002			TSTB	TRO(R3)		;ALL CHARS DONE?
8747	013712	040237	013776		BNE	.+6		
8748	013716	012716	013346		BIC	R2,RXTCR		;ZERO LINE DONE INDICATOR
					MOV	#SNAP,(SP)		;RESET THE BACKGROUND TIMING LOOP

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-2
CVDZBO.P:1 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP

```

8749 013722 000002          RTI
8750
8751
8752                                ;FINISH UP ROUTINE
8753 013724 106427 000200    OUT:  MTPS  @MASK          ;STOP ALL INTERRUPTS
8754 013730 104413          DEVICE.CLR ;CLEAR ALL INTERRUPTS AWAY
8755 013732 005003          CLR    R3
8756 013734 005037 001374    CLR    SAVLIN
8757 013740 012702 000001    MOV    @1,R2
8758 013744 030237 001366    16:   BIT    R2,LINE      ;VALID LINE ?
8759 013750 001405          BEQ    26                ;NO
8760 013752 022763 000400 001436  CMP    @400,TRO(R3)    ;RECEIVED A BINARY COUNT PATTERN ?
8761 013760 001401          BEQ    .+4              ;YES
8762 013762 104030          ERROR+ 30             ;THE LINE FAILED TO RECEIVE A FULL
8763                                ;BINARY COUNT PATTERN
8764 013764 005237 001374    24:   INC    SAVLIN      ;SET UP FOR NEXT LINE
8765 013770 005723          TST    (R3)+          ;ADD 2
8766 013772 104420          SHIFT          ;SET UP NEXT LINE POINTER
8767 013774 000763          BR     16            ;FINISHED ?
8768 013776 000000    RXTCR: 0             ;RX IMAGE OF TCR BITS
8769
8770
8771                                ;***** TEST 4 *****
8772                                ;DZV11 RELATIVE TIMING TEST.
8773                                ;EACH SELECTED LINE WILL IN TURN RUN 16. CHARS
8774                                ;AT ALL BAUD RATES AND THEN THE HIGHEST BAUD
8775                                ;WITH ALL CHAR LENGTHS. EACH NEW PARAMETER SHOULD
8776                                ;DECREASE IN TIME FROM THE PREVIOUS PARAMETERS SELECTED.
8777                                ;THE TIME IS CHECKED AGAINST THE LAST PARAMETER USED
8778                                ;AND A LOWER TIME IS EXPECTED ON THE CURRENT PARAMETER.
8779                                ;PARAMETERS ARE:
8780                                ;EIGHT BITS/PER/CHAR - TWO STOP BITS AT
8781                                ;50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000
8782                                ;2400, 3600, 4800, 7200, 9600 BAUD.
8783                                ;19.2 K BAUD - TWO STOP BITS AT
8784                                ;SEVEN, SIX, FIVE BITS/PER/CHAR.
8785                                ;AFTER EACH LINE HAS FINISHED ALL THE ABOVE PARAMETERS
8786                                ;THE NEXT SELECTED LINE IS THEN TESTED.
8787                                ;WHEN RUNNING UNDER THE APT MANUFACTURING SYSTEM
8788                                ;THIS TEST IS ONLY RUN THE FIRST PASS
8789
8790                                ;* TEST 4
8791                                ;*****
8792                                ;TST4: SCOPE
8793                                ;MOV    @4,@TSTNM      ;LOAD THE NUMBER OF THIS TEST
8794                                ;MOV    @TST5,NEXT    ;POINT TO THE START OF THE NEXT TEST
8795                                ;MOV    @30,LOCK      ;USE THIS ADDRESS IF A TIGHT SCOPE LOOP IS SELECTED
8796                                ;BITB   @1,@ENV        ;RUNNING UNDER APT?
8797                                ;BEQ    106          ;IF NOT CONTINUE WITH TEST
8798                                ;TST    @PASS        ;IF YES IS THIS FIRST PASS
8799                                ;BEQ    106          ;IF NOT 1ST PASS SKIP TEST
8800                                ;JMP    @NEXT
8801                                ;MOV    @2,@TIMES      ;SET UP FOR 2 ITERATIONS
8802                                ;CLR    OFFSET        ;RESET THIS VARIABLE
8803                                ;CLR    SAVLIN        ;RESET LINE NUMBER INDICATOR
8804                                ;CLR    XMTLIN       ;USE THIS WORD TO TELL WHAT LINE TRANSMITTED
8805                                ;MOV    @1,R2          ;USE R2 AS A BIT POINTER

```

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-3
 CVDZB0.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP

8801	C14074	012703	010070		MOV	@RCVON!SSO!EIGHT!TWOSTOP,R3	;BUILD TEMPORARY PARAMETERS	
8802	014100	030237	001366	14:	BIT	R2.LINE	;IS THIS LINE ACTIVE?	
8803	014104	001014			BNE	34	;IF SO, GO GET STARTED	
8804	014106	012703	010070	24:	MOV	@RCVON!SSO!EIGHT!TWOSTOP,R3	;LOAD PARAMETERS TEMPORARILY	
8805	014112	005237	001376		INC	XMTLIN	;POINT TO THE NEXT LINE TO TRANSMIT	
8806	014116	042703	000007		BIC	#7,R3	;MAKE SURE TEMPORARY PARAMETERS POINT TO 0	
8807	014122	053703	001376		BIS	XMTLIN,R3	;ADD DESIRED LINE NUMBER	
8808	014126	005037	015420		CLR	OFFSET		
8809	014132	104420			SHIFT		;POINT TO THE NEXT LINE	
8810	014134	000761			BR	14	;PROCESS THE NEXT LINE	
8811	014136			34:				
(1)	014136	104417			DCLASH		;CLEAR DEVICE AND SET MAINT BIT IF 1 MODE	
8812	014140	042703	010000		BIC	@RCVON,R3	;ZERO PARAMTERS FOR TX LINE	
8813	014144	010377	165650		MOV	R3,@DZVLPR	;LOAD PARAMTERS FOR TX	
8814	014150	005737	001372		TST	MODE	;STAGGERED?	
8815	014154	100007			BPL	1004	;BR IF NO	
8816	014156	000241			CLC		;SET UP LINE	
8817	014160	006003			ROR	R3		
8818	014162	103002			BCC	984	;BR IF LINE WAS EVEN	
8819	014164	000241			CLC		;PREPARE TO MKE LINE EVEN	
8820	014166	000401			BR	994	;CONTINUE	
8821	014170	000261		984:	SEC		;PREPARE TO MAKE LINE ODD	
8822	014172	006103		994:	ROL	R3	;SET ALTERED LINE	
8823	014174	052703	010000	1004:	BIS	@RCVON,R3	;SET RX ON	
8824	014200	010377	165614		MOV	R3,@DZVLPR	;LOAD RX PARAMETERS	
8825	014204	010337	001374		MOV	R3,SAVLIN	;SET FOR RECEIV. LINE	
8826	014210	042737	177774	001374	BIC	#C<3>,SAVLIN	;ISOLATE LINE NO.	
8827	014216	042703	000003		BIC	#3,R3	;CLEAR OLD LINE #	
8828	014222	053703	001376		BIS	XMTLIN,R3	;SET LINE UP AGAIN	
8829	014226	010337	001402		MOV	R3,REGIST	;SAVE PARAMETERS FOR PRINTOUT	
8830	014232	104422			BUFSET		;ZERO DATA BUFFER	
8831	014234	005037	001342		CLR	#TMP0	;USE #TMP0 TO COUNT TOTAL NUMBER OF TRANSMISSIONS	
8832	014240	005037	001344		CLR	#TMP1	;INITIALIZE THE TIMER	
8833	014244	005037	001350		CLR	#TMP3	;INITIALIZE THESE BITS ALSO	
8834	014250	012737	000020	001400	MOV	#20,XMTCNT	;SET HOW MANY CHARACTERS TO TRANSMIT	
8835	014256	012777	015046	165560	MOV	@XMTSRV,@DZVTIV		
8836	014264	012777	015216	165546	MOV	@RXISR1,@DZVRIV		
8837	014272	012777	000200	165542	MOV	@MASK,@DZVRIS		
8838	014300	012777	000200	165540	MOV	@MASK,@DZVTIS		
8839	014306	110277	165512		MOVB	R2,@DZVTCR	;START THE VALID LINE	
8840	014312	052777	040140	165470	BIS	#TIE!RIE!MSENA@,@DZVCSR		
8841	014320	106427	000000		MTPS	#0	;LOWER THE PRIORITY TO ALLOW INTERRUPTS	
8842	014324	032777	000100	165456	44:	BIT	@RIE,@DZVCSR	;IS ROUTINE DONE?
8843	014332	001407			BEQ	54	;WHEN ALL IS DONE RX IE IS CLEARED IN ISR.	
8844	014334	005237	001344		INC	#TMP1	;INCREMENT TIMER	
8845	014340	001371			BNE	44	;WHEN IT OVERFLOWS	
8846	014342	005237	001350		INC	#TMP3	;CATCH CARRY	
8847	014346	001366			BNE	44	;CONTINUE TEST	
8848	014350	104011			ERROR.	11	;INTERRUPTS NOT FINISHED	
8849	014352	004737	007322	54:	JSR	PC,SERV.G	;<?G>?	
8850	014356	104401			SCOP1		;LOOP?	
8851	014360	062737	000002	015420	ADD	#2,OFFSET		
8852	014366	022703	017400		CMF	#17400,R3		
8853	014372	003006			BGT	64		
8854	014374	032703	000030		BIT	@BIT4+BIT3,R3	;IS CHARACTER SIZE DONE?	
8855	014400	001642			BEQ	24		

CVDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-4
CVDZBO.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP.

8856 014402 162703 000010
8857 014406 000653
8858 014410 062703 000400
8859 014414 000650

SUB #BITS,R3
BR 3#
6# : ADD #400,R3
BR 3#

***** TEST 5 *****
;THE MAIN FUNCTION OF THIS TEST IS TO VERIFY
;THAT "PE" (PARITY ERROR) CAN BE FLAGGED BY
;THE UARTS. THIS TEST WILL NOT BE DONE UNLESS
;YOU ARE IN "STAGGERED" MODE.
;40(8) CHARS ARE USED FOR THIS TEST.
;ALL SELECTED LINES WILL BE ENABLED AT THE SAME TIME.
;THIS TEST FIRST CHECKS EVEN PARITY FOR ODD LINES AND
;ODD PARITY FOR EVEN LINES, THEN IT CHECKS THE REVERSE.

::* TEST 5

8860
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(3)
(6)
(5) 014416 000004
(3) 014420 012737 000005 001246
(2) 014426 012737 004250 001362
(1) 014434 005737 001372
(1) 014440 100131
(1) 014442 105037 001425
(1) 014446 104413
(1) 014450 013701 001370
(1) 014454 042701 000200
(1) 014460 052701 000100
(1) 014464 012702 000001
(1) 014470 030237 001366
(1) 014474 001420
(1) 014476 105737 061425
(1) 014502 001004
(1) 014504 032701 000001
(1) 014510 001006
(1) 014512 000403
(1) 014514 032701 000001
(1) 014520 001402
(1) 014522 052701 000200
(1) 014526 010177 165266
(1) 014532 042701 000200
(1) 014536 005201
(1) 014540 006302
(1) 014542 032702 000020
(1) 014546 001750
(1) 014550 005037 001374
(1) 014554 005037 001342
(1) 014560 005003
(1) 014562 012737 000040 001400
(1) 014570 104422
(2) 014572 012777 015046 165244
(2) 014600 012777 014726 165232
(2) 014606 012777 000200 165226
(2) 014614 012777 000200 165224
(2) 014622 052777 040140 165160
(1) 014630 113777 001366 165166
(1) 014636 106427 000000
(1) 014642 005037 014720
(1) 014646 005037 014722

TST5: SCOPE
MOV #5,#TSTNM ;LOAD THE NUMBER OF THIS TEST
MOV #1EOP,NEXT ;POINT TO THE END-OF-PASS HANDLER
TST MODE ;IS THIS STAGGERED MODE?
BPL 17# ;IF NOT, DON'T DO THIS TEST
CLRB DONFLG ;SET UP FOR FIRST TEST PASS
14# : DEVICE.CLR ;SET DCLR IN CSR
MOV PAR,R1 ;USE R1 TO BUILD PARAMETERS TO BE LOADED
BIC #ODDPAR,R1 ;MAKE SURE ODD PARITY ISN'T SET
BIS #PARITY,R1 ;MAKE SURE PARITY IS TURNED ON
MOV #1,R2 ;USE R2 AS A LINE POINTER
1# : BIT R2,LINE ;IS THIS A VALID LINE?
BEQ 3# ;IF NOT, SKIP TO THE NEXT LINE
TSTB DONFLG ;FIRST PASS THROUGH TEST?
BNE 15# ;IF NO BRANCH
BIT #BIT0,R1 ;IS THIS LINE AN ODD LINE?
BNE 2# ;IF IT'S ODD, USE EVEN PARITY
BR 16# ;IF EVEN SET FOR ODD PARITY
15# : BIT #BIT0,R1 ;IF THE LINE IS EVEN SET FOR EVEN PAR.
BEQ 2# ;GO LOAD PARAMETER
16# : BIS #ODDPAR,R1 ;IF IT'S ODD, USE ODD PARITY
2# : MOV R1,#DZVLPB ;LOAD THE LINE PARAMETER REGISTER
BIC #ODDPAR,R1 ;SET UP THE NEXT PARITY TO EVEN
3# : INC R1 ;POINT TO THE NEXT LINE
ASL R2 ;
BIT #BIT4,R2 ;ALL LINES DONE?
BEQ 1# ;IF NOT, GO CHECK THE NEXT LINE
CLR SAVLIN ;CLEAR THE LINE NUMBER INDICATOR
CLR #THPO ;USE #THPO TO COUNT TOTAL NUMBER OF TRANSMISSIONS
CLR R3 ;USE R3 TO COUNT TOTAL NUMBER OF RECEPTIONS
MOV #40,XMTCNT ;TRANSMIT A BINARY COUNT PATTERN(00-40)
BUFSET ;ZERO BUFF'A AREA
MOV #XMTSRV,#DZVTIV ;SET UP THE TRANSMITTER INTERRUPT VECTOR
MOV #9#,#DZVRIV ;SET UP THE RECEIVER INTERRUPT VECTOR
MOV #MASK,#DZVRIS ;SET THE INTERRUPT VECTOR STATUS
MOV #MASK,#DZVTIS ;SET TRANSMITTER INTERRUPT PRIORITY
BIS #RIE!TIE!MSENAB,#DZVCSR ;ENABLE THE DEVICE
MOVB LINE,#DZVTCR ;ENABLE ALL SELECTED LINES
MTPS #0 ;ALLOW INTERRUPTS
4# : CLR 7#
CLR 8#

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-5
 CVDZBO.P11 21-AUG-84 08:28 DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP CORP

```

(1) 014652 032777 000100 165130 54: BIT @RIE,@DZVCSR ;WHEN RX DONE, RIE WILL =0
(1) 014660 001407 BEQ 64 ;BR IF ALL DONE
(1) 014662 005237 014720 INC 74
(1) 014666 001371 BNE 54
(1) 014670 105237 014772 INCB 84
(1) 014674 100366 BPL 54
(1) 014676 104011 ERROR. 11 ;RX FAILED TO FINISH (INTERRUPT)
(1) 014700 106427 000200 64: MTPS @MASK ;SHUT OFF INTERRUPTS
(1) 014704 105737 001425 TSTB DONFLG ;IS THIS SECOND TEST PASS
(1) 014710 001005 BNE 174 ;IF SO GET OUT
(1) 014712 105237 001425 INCB DONFLG ;INDICATE FIRST TEST PASS DONE
(1) 014716 000653 BR 144 ;START OVER
(1) 014720 000000 74: 0
(1) 014722 000000 84: 0
(1) 014724 104400 174: ADVANCE

(1)
(1)
(1) ;RECEIVER SERVICE ROUTINE
(1)
(1) 014726 01. 74 165062 94: MOV @DZVRBUF,R4 ;GET THE CHARACTER
(1) 014732 010401 MOV R4,R1 ;COPY THE RECEIVED INFORMATION
(1) 014734 000301 SWAB R1 ;GET THE LINE NUMBER IN THE LOWER BYTE
(1) 014736 042701 177774 BIC @+C<3>,R1 ;ISOLATE THE LINE NUMBER
(1) 014742 010137 001374 MOV R1,SAYLIN ;SET LINE INDIC. TO RECEIVING LINE
(1) 014746 005704 TST R4 ;IS DATA VALID SET?
(1) 014750 100401 BMI 104 ;IF YES DON'T PRINT ERROR
(1) 014752 104023 ERROR. 23 ;DATA VALID NOT SET
(1) 014754 010105 104: MOV R1,R5 ;BUILD LINE NO. FOR
(1) 014756 000305 SWAB R5 ;EXPECTED DATA IN RECEIVER BUFFER
(1) 014760 006301 ASL R1 ;ADJUST R1 FOR OFFSET
(1) 014762 156105 001436 BISB TRO(R1),R5 ;LOAD CHARACTER IN EXPECTED
(1) 014766 052705 110000 BIS @VALID!PARER,R5 ;BUILD WHAT WAS EXPECTED
(1) 014772 020405 CMP R4,R5 ;DOES RECEIVED=EXPECTED
(1) 014774 001401 BEQ 124 ;IF YES DON'T PRINT ERROR
(1) 014776 104006 ERROR. 6 ;ERROR- DID NOT GET CORRECT INFORMATION
(1) 015000 005261 001436 124: INC TRO(R1) ;SET UP THE NEXT CHARACTER
(1) 015004 005203 INC R3 ;ADD TO THE TOTAL RECEIVED COUNT
(1) 015006 032777 040000 164774 BIT @TIE,@DZVCSR ;ARE TRANSMISSIONS DONE?
(1) 015014 001011 BNE 134 ;IF NO, GO RECEIVE SOME MORE
(1) 015016 023703 001342 CMP @TMP0,R3 ;ARE ALL CHARACTERS RECEIVED?
(1) 015022 001006 BNE 134 ;IF NO, GO RECEIVE SOME MORE
(1) 015024 042777 000100 164756 BIC @RIE,@DZVCSR ;DISABLE RECEIVER INTERRUPTS
(1) 015032 012716 014700 MOV @64,(SP) ;CRUNCH THE STACK
(1) 015036 000002 RTI ;RETURN AND FINISH
(1) 015040 012716 014642 134: MOV @44,(SP) ;CRUNCH THE STACK
(1) 015044 000002 RTI ;GO BACK TO RECEIVER WAIT LOOP

```

(VDZB D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 27-6
 CVDZB0.P11 21-AUG-84 08:28 0ZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP.

```

0862
0863
0864
0865
0866 015046 117701 164740
0867 015052 042701 177774
0868 015056 013705 001374
0869 015062 010137 001374
0870 015066 006301
0871 015070 023761 001400 001426
0872 015076 003414
0873 015100 005777 164704
0874 015104 100401
0875 015106 104003
0876 015110 116177 001426 164716 24:
0877 015116 005261 001426
0878 015122 005237 001342
0879 015126 000415
0880 015130 012700 000001 64:
0881 015134 006201
0882 015136 005301 124:
0883 015140 100402
0884 015142 006300
0885 015144 000774
0886 015146 140077 164652 34:
0887 015152 001403
0888 015154 042777 040000 164626
0889 015162 010537 001374 74:
0890 015166 000002
0891
0892
0893
0894
0895 015170 012737 000004 001346 BUILD: MOV 04, $TMP2
0896 015176 006037 001350 14: ROR $TMP3
0897 015202 006037 001344 ROR $TMP1
0898 015206 005337 001346 DEC $TMP2
0899
0900 015212 001371 BNE 14
0901 015214 000207 RTS PC
0902

```

TRANSMITTER INTERRUPT SERVICE

 ; GET THE LINE NUMBER.
 ; CLEAR JUNK
 ; SAVE REC. LINE NO.
 ; LOAD TRANS LINE NO FOR ERROR PRINTOUT
 ; ADJUST R1 FOR OFFSET
 ; HAVE ALL CHAR. BEEN SENT
 ; IF YES GO CLEAR TCR
 ; TRY SET?
 ; IF YES GO LOAD CHAR.
 ; *TRANSMITTER NOT READY- FALSE INTERRUPT
 ; LOAD THE CURRENT CHARACTER FOR THIS LINE
 ; SET UP NEXT CHARACTER FOR THIS LINE
 ; UP THE NUMBER OF TRANSMISSIONS
 ; GO RETURN
 ; SET UP A DESELECTION POINTER
 ; GET LINE NO. AGAIN
 ; REDUCE THE COUNT. WAS THIS THE LINE?
 ; IF SO, GO DISABLE THE ENABLE BIT FOR IT
 ; MOVE THE POINTER TO THE NEXT LINE
 ; GO CHECK THE NEXT LINE
 ; DISABLE THE LINE POINTED TO BY R0
 ; IF MORE LINES ARE ACTIVE, GO CONTINUE TRANSMIT
 ; IF NOT, DISABLE TRANSMITTER INTERRUPTS
 ; RESTORE RECEIV. LINE
 ; RETURN TO THE TIMING LOOP

RELATIVE TIME BUILDING ROUTINE

 ; ROTATE 4 BITS BACK INTO \$TMP1
 ; GET THE BITS FROM \$TMP3, THE HIGH BYTE
 ; OF THE RELATIVE TIME COUNTER. PUT THEM BACK
 ; INTO \$TMP1 USING THE CARRY BIT WITH
 ; ROTATE INSTRUCTIONS
 ; REDUCE COUNT. ALL BITS BACK? IF NOT, GET MORE
 ; RETURN TO CALLING TEST

RECEIVER SERVICE ROUTINE

```

8904
8905
8906 015216 105777 164566      RXISR1: TSTB   BDZVCSR      ;IS THE RECEIVER REALLY READY?
8907 015222 100401              BMI          1#      ;IF SO, GO SERVICE IT
8908 015224 104004              ERROR*     4        ;ERROR- RECEIVER DONE FLAG ISN'T SET
8909 015226 017704 164562      1#:  MOV     BDZVRBUF,R4 ;SAVE THE RECEIVER INFORMATION
8910 015232 100401              BMI          2#      ;IF IT WAS VALID, GO PROCESS IT
8911 015234 104023              ERROR*     23       ;ERROR- DATA VALID WASH'T SET
8912 015236 032704 040000      2#:  BIT     @OVRUN,R4 ;OVERRUN ERROR FLAG SET?
8913 015242 001401              BEQ         6#      ;IF NOT DON'T TYPE ERROR
8914 015244 104024              ERROR*     24       ;OVERRUN ERROR
8915 015246 032704 020000      6#:  BIT     @FRMERR,R4 ;FRAMING ERROR FLAG SET?
8916 015252 001401              BEQ         9#      ;IF NOT DON'T TYPE ERROR
8917 015254 104025              ERROR*     25       ;FRAMING ERROR
8918 015256 032704 010000      9#:  BIT     @PARER,R4 ;PARITY ERROR FLAG SET?
8919 015262 001401              BEQ         3#      ;IF NOT, GO CONTINUE PROCESSING
8920 015264 104026              ERROR*     26       ;ERROR- RECEIVER ERROR FLAG SET
8921 015266 013701 001374      3#:  MOV     SAVLIN,R1  ;CALCULATE THE DATA OFFSET
8922 015272 006301              ASL         R1      ;ALIGN IT ON A WORD BOUNDARY
8923 015274 120461 001436      CMPB      R4,TRO(R1) ;IS THE CHARACTER WHAT IT SHOULD BE?
8924 015300 001407              BEQ         4#      ;IF SO,GO CONTINUE PROCESSING
8925 015302 116105 001436      MOVB      TRO(R1),R5 ;GET WHAT WAS EXPECTED FOR ERROR REPORTING
8926 015306 042705 177400      BIC       @+C<37>,R5 ;ELIMINATE PROPAGATED SIGN
8927 015312 042704 177400      BIC       @+C<37>,R4 ;ISOLATE THE ACTUAL CHARACTER
8928 015316 104005              ERROR*     5        ;DATA ERROR
8929 015320 005261 001436      4#:  INC     TRO(R1)   ;SET UP THE NEXT EXPECTED CHARACTER
8930 015324 122761 000020 001436  CMPB      @20,TRO(R1) ;HAVE ALL CHARACTERS BEEN RECEIVED?
8931 015332 001031              BNE        8#      ;IF NOT RETURN
8932 015334 126137 001436 001342  CMPB      TRO(R1),@TMP0 ;ALL CHARAC. RECEIVED?
8933 015342 001025              BNE        8#      ;IF SO,GO DETERMINE THE TIMING
8934 015344 004737 015170      JSR       PC,BUILD  ;GET THE RELATIVE TIME (SIGNIFICANT BITS)
8935 015350 013700 015420      MOV       OFFSET,R0 ;GET POINTER
8936 015354 013760 001344 002050  MOV       @TMP1,TMTBL(R0) ;SAVE THIS TEST'S TIME
8937 015362 005737 015420      TST      OFFSET    ;FIRST TEST?
8938 015366 001410              BEQ         7#      ;IF NOT, GO CHECK THE TIME
8939 015370 005740              TST      -(R0)     ;POINT TO THE PREVIOUS TIME TAKEN
8940 015372 026037 002050 001344  CMP       TMTBL(R0),@TMP1 ;IS THIS TIME WHAT IT SHOULD BE?
8941 015400 101003              BMI          7#      ;IF SO, GO TO THE NEXT TEST
8942 015402 016005 002050      MOV       TMTBL(R0),R5 ;PLACE WHAT WAS EXPECTED IN R5
8943 015406 104021              ERROR*     21       ;TIMING ERROR
8944 015410 042777 000140 164372  7#:  BIC     @RIE!HSENAB,BDZVCSR ;DISABLE THE DEVICE
8945 015416 000002      8#:  RTI          ;RETURN TO THE PROGRAM
8946 015420 000000      OFFSET: 0

```

			,ERROR TABLE	
			.ERRTAB:	
8948				
8949				
8950	015422	000000	0	,ERROR 0
8951	015424	000000	0	
8952	015426	000000	0	
8953				
8954	015430	015650	EM1	,ERROR
8955	015432	016774	DM1	
8956	015434	017174	DT1	
8957				
8958	015436	015723	EM2	,ERROR 2
8959	015440	017020	DM2	
8960	015442	017206	DT2	
8961				
8962	015444	015751	EM3	,ERROR 3
8963	015446	017053	DM3	
8964	015450	017224	DT3	
8965				
8966	015452	016010	EM4	,ERROR 4
8967	015454	017053	DM3	
8968	015456	017224	DT3	
8969				
8970	015460	016037	EM5	,ERROR 5
8971	015462	017065	DM4	
8972	015464	017232	DT4	
8973				
8974	015466	016066	EM6	,ERROR 6
8975	015470	017065	DM4	
8976	015472	017232	DT4	
8977				
8978	015474	000000	0	
8979	015476	000000	0	
8980	015500	000000	0	
8981				
8982	015502	000000	0	
8983	015504	000000	0	
8984	015506	000000	0	
8985				
8986	015510	016125	EM11	,ERROR 11
8987	015512	017053	DM3	
8988	015514	017224	DT3	
8989				
8990	015516	000000	0	
8991	015520	000000	0	
8992	015522	000000	0	
8993				
8994	015524	016163	EM13	,ERROR 13
8995	015526	017053	DM3	
8996	015530	017224	DT3	
8997				
8998	015532	016214	EM14	,ERROR 14
8999	015534	017053	DM3	
9000	015536	017224	DT3	
9001				
9002	015540	016246	EM15	,ERROR 15
9003	015542	000000	0	

9004	015544	000000	0	
9005				
9006	015546	016310	EM16	
9007	015550	017053	DMS	
9008	015552	017224	DT3	
9009				
9010	015554	016362	EM17	ERROR 17
9011	015556	017053	DMS	
9012	015560	017224	DT3	
9013				
9014	015562	016420	EM20	
9015	015564	017053	DMS	
9016	015566	017224	DT3	
9017				
9018	015570	016461	EM21	ERROR 21
9019	015572	017114	DMS	
9020	015574	017250	DT3	
9021				
9022	015576	000000	0	
9023	015600	000000	0	
9024	015602	000000	0	
9025				
9026	015604	016511	EM23	ERROR 23
9027	015606	017053	DMS	
9028	015610	017224	DT3	
9029				
9030	015612	016541	EM24	
9031	015614	017053	DMS	
9032	015616	017224	DT3	
9033				
9034	015620	016567	EM25	
9035	015622	017053	DMS	
9036	015624	017224	DT3	
9037				
9038	015626	016617	EM26	
9039	015630	017053	DMS	
9040	015632	017224	DT3	
9041				
9042	015634	016646	EM27	
9043	015636	017053	DMS	
9044	015640	017224	DT3	
9045				
9046	015642	016714	EM30	
9047	015644	017053	DMS	
9048	015646	017224	DT3	

```

9050
9051
9053 015650 047200 020117 052502 EM1: .ASCIZ <200>/NO BUS REPLY RESPONSE FROM DZV11 REGISTER/
9056 015723 200 042522 044507 EM2: .ASCIZ <200>/REGISTER R/W FAILURE?
9057 015751 200 051124 047101 EM3: .ASCIZ <200>/TRANSMIT READY (TRDY) NOT SET/
9058 016010 051200 041505 044505 EM4: .ASCIZ <200>/RECEIVER DONE NOT SET/
9059 016037 200 040504 040524 EM5: .ASCIZ <200>/DATA COMPARISON ERROR/
9060 016066 042200 053132 030461 EM6: .ASCIZ <200>/DZV11 «RECEIVER BUFFER» ERROR/
9061 016125 200 042522 042503 EM11: .ASCIZ <200>/RECEIVER FAILED TO INTERRUPT/
9062 016163 200 044523 047514 EM13: .ASCIZ <200>/SILO ALARM SET TOO SOON/
9063 016214 051600 046111 020117 EM14: .ASCIZ <200>/SILO ALARM FAILED TO SET/
9064 016246 040600 052103 047511 EM15: .ASCIZ <200>/ACTION DETECTED ON INVALID LINE./
9065 016310 051200 040505 044504 EM16: .ASCIZ <200>/READING DZVRBUF DID NOT CLEAR SILO ALARM/
9066 016362 042200 052101 020101 EM17: .ASCIZ <200>/DATA VALID SHOULD NOT BE SET/
9067 016420 051200 041505 044505 EM20: .ASCIZ <200>/RECEIVER DONE SHOULD NOT BE SET/
9068 016461 200 042522 040514 EM21: .ASCIZ <200>/RELATIVE TIMING ERROR./
9069 016511 200 040504 040524 EM23: .ASCIZ <200>/DATA VALID IS NOT SET!/
9070 016541 200 040504 040524 EM24: .ASCIZ <200>/DATA OVERRUN IS SET!/
9071 016567 200 051106 046501 EM25: .ASCIZ <200>/FRAMING ERROR OCCURRED/
9072 016617 200 040520 044522 EM26: .ASCIZ <200>/PARITY ERROR OCCURRED/
9073 016646 051600 046111 020117 EM27: .ASCIZ <200>/SILO ALARM FAILED TO CAUSE INTERRUPT/
9074 016714 046200 047111 020105 EM30: .ASCIZ <200>/LINE DID NOT RECEIVE FULL BINARY COUNT PATTERN/
9075
9076 016774 052200 040522 020120 DM1: .ASCIZ <200>/TRAP PC DZV11 REG/
9077 017020 042600 050130 041505 DM2: .ASCIZ <200>/EXPECTED FOUND REGISTER/
9078 017053 200 044514 042516 DM3: .ASCIZ <200>/LINE NO./
9079 017065 200 054105 042520 DM4: .ASCIZ <200>/EXPECTED FOUND LINE/
9080 017114 052200 020130 044514 DM5: .ASCIZ <200>/TX LINE PREVIOUS TIME ACTUAL TIME PARAMETER/
9081
9082 017174 .EVEN
9086
9087 017174 000002 DT1: .DATA TABLES FOR ERROR MESSAGES
9088 017176 006 003 .BYTE 6,3
9089 017200 001330 $REG1
9090 017202 006 001 .BYTE 6,1
9091 017204 001326 $REG0
9092
9093 017206 000003 DT2: 3
9094 017210 006 004 .BYTE 6,4
9095 017212 001340 $REG5
9096 017214 006 001 .BYTE 6,1
9097 017216 001336 $REG4
9098 017220 006 001 .BYTE 6,1
9099 017222 001326 $REG0
9100
9101 017224 000001 DT3: 1
9102 017226 003 001 .BYTE 3,1
9103 017230 001374 SAVLIN
9104
9105 017232 000003 DT4: 3
9106 017234 006 004 .BYTE 6,4
9107 017236 001340 $REG5
9108 017240 006 001 .BYTE 6,1
9109 017242 001336 $REG4
9110 017244 003 001 .BYTE 3,1
9111 017246 001374 SAVLIN
  
```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 29-1
CVDZBD.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP CORP

9112			
9113	017250	000004	
9114	017252	003	005
9115	017254	001374	
9116	017256	006	011
9117	017260	001340	
9118	017262	006	007
9119	017264	001344	
9120	017266	006	001
9121	017270	001402	

DT5: 4
 .BYTE 3.5
 SAVLIN
 .BYTE 6.9.
 #REGS
 .BYTE 6.7
 #TMP1
 .BYTE 6.1
 REGIST

TABLE OF DELAY TIMES FOR INDIVIDUAL BAUD RATES

9128			
9129			
9130			
9131	017272	002450	
9132	017274	001560	
9133	017276	001120	
9134	017300	000750	
9135	017302	000660	
9136	017304	000330	
9137	017306	000150	
9138	017310	000060	
9139	017312	000040	
9140	017314	000030	
9141	017316	000020	
9142	017320	000010	
9143	017322	000001	
9144	017324	000001	
9145	017326	000001	
9146	017330	000001	
9147			
9148			
9149			

DLTYBL:	2450	TIME FOR	50 BAUD
	1560	TIME FOR	75 BAUD
	1120	TIME FOR	110 BAUD
	750	TIME FOR	134 BAUD
	.60	TIME FOR	150 BAUD
	330	TIME FOR	300 BAUD
	150	TIME FOR	600 BAUD
	60	TIME FOR	1200 BAUD
	40	TIME FOR	1800 BAUD
	30	TIME FOR	2000 BAUD
	20	TIME FOR	2400 BAUD
	10	TIME FOR	3600 BAUD
	1	TIME FOR	4800 BAUD
	1	TIME FOR	7200 BAUD
	1	TIME FOR	9600 BAUD
	1	TIME OF DELAY FOR	19200 BAUD

DELAYS WERE COMPUTED TO ALLOW MAXIMUM TIME AT EACH BAUD RATE
FOR ALL TESTS TO FUNCTION CORRECTLY ON A LSI11.

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 30
 CVDZB0.P11 21-AUG-84 08:28

DZV11 DEVICE DIAGNOSTICS. COPYRIGHT 1977 DIGITAL EQUIP. CORP.

```

9151 .SBTTL FALCON (KXT-11) UPGRADE ROUTINES. ;GPA
9152 ;
9153 ; THE FOLLOWING ROUTINES HAVE BEEN ADDED TO ALLOW DIAGNOSTIC(S)
9154 ; TO RUN ON A FALCON (KXT-11) BASED SYSTEM.
9155 ; TO DETERMINE WHETHER WE'RE A FALCON OR NOT, WE'LL SIZE THE 1ST 3/4 OF
9156 ; THE I/O PAGE (28K TO 31K). FALCON HAS 2KW LOCAL RAM AT 28K(+4) TO 30K
9157 ; AND A MACRO-ODT AT 30K TO 31K. CONSEQUENTLY, ALL I/O DEVICES MUST
9158 ; BE PLACED BETWEEN 174000 AND 177776. ADDITIONALLY, WE'LL STRAP THE
9159 ; EMT AND TRAP SERVICE LEVEL TO PRI6, AND SET THE HALT VECTOR SO THAT
9160 ; WE CAN STOP THE SUCKER !!
9161 ;
9162 ; TO MINIMIZE THE IMPACT OF THESE CHANGES ON FINAL PROGRAM SIZE, THE
9163 ; BULK OF THIS CODE IS PLACED IN THE FLOATING VECTOR SPACE (400-776).
9164 ; IF THE CPU AT HAND IS A FALCON (KXT11), IT STAYS THERE (NO HARM DONE).
9165 ; OTHERWISE, THE AREA IS RESTORED TO ITS ORIGINAL "TRAP-CATCHER" STATE.
9166 ;
9167 ;
9168 017332 005227 177777 FALCON: INC 0-1 ; ONCE-ONLY !!! ;GPA
9169 017336 001002 BNE 1# ; ;GPA
9170 017340 004737 000400 CALL KXTCHK ; EXECUTE FALCON CHECK ;GPA
9171 017344 005727 1# TST (PC)+ ; TEST FALCON FLAG... ;GPA
9172 017346 000000 KXTFLAG: 0 ; ...NZ = FALCON... ;GPA
9173 017350 000207 RETURN ; ...AND RETURN TO CALLER... ;GPA
9174 ;
9175 ;SVPC= ;GPA
9176 000400 ; = 400 ;GPA
9177 000400 005037 017346 KXTCHK: CLR KXTFLAG ; RESTORE FROM 374:376 AT END ;GPA
9178 000404 013746 000004 MOV B#4, -(SP) ; ASSUME NOT FALCON. ;GPA
9179 000410 012737 000504 000004 MOV #2#, B#4 ; SAVE ERROR VECTOR. ;GPA
9180 000416 012700 160010 MOV #160010, R0 ; SET A TRAP CATCHER. ;GPA
9181 000422 005720 1# TST (R0)+ ; FALCON RAM STARTS AT 28K+4. ;GPA
9182 000424 000240 ; ;GPA
9183 000426 020027 174000 CMP R0, #174000 ; SIZE TO 31K. ;GPA
9184 000432 103773 BLO 1# ;GPA
9185 000434 010037 017346 MOV R0, KXTFLAG ; MUST BE FALCON, SET THE FLAG ;GPA
9186 000440 012700 000040 MOV #40, R0 ; GET PRI1 BIT... ;GPA
9187 000444 040037 000006 BIC R0, B#6 ; ...AND LOWER BUS-ERROR... ;GPA
9188 000450 040037 000016 BIC R0, B#16 ; ...BPT... ;GPA
9189 000454 040037 000022 BIC R0, B#22 ; ...IOT... ;GPA
9190 000460 040037 000032 BIC R0, B#32 ; ...EMT... ;GPA
9191 000464 040037 000036 BIC R0, B#36 ; ...AND TRAP SERVICE TO PRI6 ;GPA
9192 000470 012737 170000 000140 MOV #170000, B#140 ; ENABLE "BREAK" HALT. ;GPA
9193 000476 012637 000004 MOV (SP)+, B#4 ; RESTORE ERROR VECTOR... ;GPA
9194 000502 000207 RETURN ; ...AND RETURN. ;GPA
9195 ;
9196 000504 012716 000512 2# MOV #3#, (SP) ; TRAP -- NOT A FALCON... ;GPA
9197 000510 000002 RTI ; ...CONTINUE. ;GPA
9198 000512 012637 000004 3# MOV (SP)+, B#4 ; RESET ERROR VECTOR ;GPA
9199 000516 012700 000402 MOV #402, R0 ; SET-UP TO RESTORE FLOATING... ;GPA
9200 000522 013701 000376 MOV B#376, R1 ; ...VECTORS (400 - 776). ;GPA
9201 000526 010602 MOV SP, R2 ; SAVE STACK POINTER IN R2 ;GPA
9202 000530 012704 000570 MOV #6#, R4 ;GPA
9203 000534 014446 4# MOV -(R4), -(SP) ; PUSH THE RESTORE CODE... ;GPA
9204 000536 020427 000546 CMP R4, #5# ; ...ONTO THE STACK. ;GPA
9205 000542 101374 BHI 4# ;GPA
9206 000544 010607 MOV SP, PC ; AND EXECUTE IT. ;GPA
    
```

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 30-1
 CVDZB0.P11 21-AUG-84 08:28 FALCON (KXT-11) UPGRADE ROUTINES.

;GPA

```

9208
9209
9210
9211 000546 010060 177776
9212 000552 010110
9213 000554 022020
9214 000556 020027 000776
9215 000562 101771
9216 000564 010206
9217 000566 000207
9218 000570
9219
9220
9221
9222
9223
9224
9225
9226
9227 000570 023727 001174 160010
9228 000576 001003
9229 000600 012737 174040 001174
9230 000606 023727 001170 000300
9231 000614 001003
9232 000616 012737 000370 001170
9233 000624 012737 000670 002464
9234 000632 012737 174000 002470
9235 000640 012737 177770 002472
9236 000646 012737 000732 002510
9237 000654 005037 002514
9238 000660 012737 000370 002516
9239 000666 000207
9240
9241 000670 030600 052123 041440
      000676 051123 040440 042104
      000704 042522 051523 024040
      000712 033461 030064 030060
      000720 030472 033467 033467
      000726 024460 000040
9242 000732 030600 052123 053040
      000740 041505 047524 020122
      000746 042101 051104 051505
      000754 020123 030050 030060
      000762 031472 030057 020051
      000770 020040 000040
9243
9244
9245
9249
9250
9251 017352
9255

; THIS CODE IS RELOCATED TO AND EXECUTED IN THE STACK AREA.
50:  MOV    R0,-2(R0)      ; RESTORE .+2...
      MOV    R1,(R0)      ; ...HALT (OR IOT).
      CMP    (R0), (R0)
      CMP    R0,#776
      BLOS  51            ; LOOP 'TIL DONE
      MOV    R2,SP        ; THEN RESTORE SP...
      RETURN             ; ...AND RETURN TO CALLER
60:
; IF FALCON, THIS AREA IS FREE FOR ANY PROGRAM UNIQUE
; CHANGES OR DATA STRUCTURES.
; BE SURE IT DOESN'T GET SCREWED UP !!
; INIT #BASE AND #VECT1 AND TWEAK THE "#GETPAR" CALLING
; SEQUENCE TO ACCEPT THE VALID FALCON RANGE.
FALCINI: CMP    #BASE,#BASE      ; IS #BASE VIRGIN ??
          BNE  10              ; SKIP NEXT IF NOT
          MOV  #174040,#BASE     ; YES, SET ENGINEERING DEFAULT
10:      CMP    #VECT1,#AVECT1   ; IS #VECT1 VIRGIN ??
          BNE  20              ; SKIP NEXT IF NOT
          MOV  #370,#VECT1      ; YES, SET ENGINEERING DEFAULT
20:      MOV  #30,GETCSR+2      ; SUBSTITUTE CSR TEXT...
          MOV  #174000,GETCSR+6
          MOV  #177770,GETCSR+10 ; ...AND VALID RANGE.
          MOV  #40,GETVEC+2     ; SUBSTITUTE VECTOR TEXT...
          CLR  GETVEC+6
          MOV  #370,GETVEC+10   ; ...AND VALID RANGE.
          RETURN              ; RETURN TO CALLER.
30:      .ASCIZ <200>'1ST CSR ADDRESS (174000:177770) '
40:      .ASCIZ <200>'1ST VECTOR ADDRESS (000:370) '
          .EVEN
#FREE= <1000-.>/2 ; FREE WORDS LEFT.
CORMAX: .=#SVPC
.END

```

CVDZB-D MACY11 30A(1032) 21-AUG-84 08:31 PAGE 31
 CVDZBD.P11 21-AUG-84 08:28 CROSS REFERENCE TABLE -- USER SYMBOLS

ABASE - 160010	83470	8456	9227	
ABORT 006612	84570			
ABORT1 006646	84570			
ABORT2 006662	84570			
ACDM1 - 000017	83520	8456		
ACDM2 - 000000	8456			
ACPUOP- 000000	8456			
ACTIVE 001420	84560*	8457*		
ADDW0 - 017470	83510	8456		
ADDW1 - 017470	83510	8456		
ADDW10- 017470	83510	8456		
ADDW11- 017470	83510	8456		
ADDW12- 017470	83510	8456		
ADDW13- 017470	83510	8456		
ADDW14- 017470	83510	8456		
ADDW15- 017470	83510	8456		
ADDW2 - 017470	83510	8456		
ADDW3 - 017470	83510	8456		
ADDW4 - 017470	83510	8456		
ADDW5 - 017470	83510	8456		
ADDW6 - 017470	83510	8456		
ADDW7 - 017470	83510	8456		
ADDW8 - 017470	83510	8456		
ADDW9 - 017470	83510	8456		
ADEVCT- 000000	8456			
ADEVN - 000001	83530	8456		
ADRCNT 006127	84570*			
ADVANC- 104400	84560	8457	8671	8860
AENV - 000000	8456			
AENV1 - 000000	8456			
AFATAL- 000000	8456			
AMADR1- 000000	8456			
AMADR2- 000000	8456			
AMADR3- 000000	8456			
AMADR4- 000000	8456			
AMMS1- 000000	8456			
AMMS2- 000000	8456			
AMMS3- 000000	8456			
AMMS4- 000000	8456			
AMSGAD- 000000	8456			
AMSGLG- 000000	8456			
AMSGTY- 000000	8456			
AMTYP1- 000000	8456			
AMTYP2- 000000	8456			
AMTYP3- 000000	8456			
AMTYP4- 000000	8456			
APASS - 000000	8456			
APRIOR- 000000	8456			
APTCSU- 000040	84570			
APTENV- 000001	84570			
APTSIZ- 000200	84570			
APTSPO- 000100	84570			
ASWREG- 000000	8456			
ATESTN- 000000	8456			
AUNIT - 000000	8456			
AUSMR - 000000	8456			

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 31-1
 CVDZBD.P11 21-AUG-84 08:28 CROSS REFERENCE TABLE -- USER SYMBOLS

AUTO.S	011564	8456	8457*				
AVECT1-	000300	8348*	8456	9230			
AVECT2-	000000	8456					
BINARD	006402	8457*					
BIT0	000001	8456*	8860				
BIT00	000001	8456*					
BIT01	000002	8456*					
BIT02	000004	8456*					
BIT03	000010	8456*					
BIT04	000020	8456*					
BIT05	000040	8456*					
BIT06	000100	8456*					
BIT07	000200	8456*					
BIT08	000400	8456*					
BIT09	001000	8456*					
SIT1	000002	8456*					
BIT10	002000	8456*					
BIT11	004000	8456*	8457				
BIT12	010000	8456*					
BIT13	020000	8456*					
BIT14	040000	8456*	8457				
BIT15	100000	8456*					
BIT2	000004	8456*					
BIT3	000010	8456*	8854	8856			
BIT4	000020	8456*	8457	8854	8860		
BIT5	000040	8456*	8457				
BIT6	000100	8456*					
BIT7	000200	8456*	8457				
BIT8	000400	8456*					
BIT9	001000	8456*					
BPTVEC-	000014	8456*					
BRK0	000400	8456*					
BRK1	001000	8456*					
BRK2	002000	8456*					
BRK3	004000	8456*					
BRW	004724	8456	8457*				
BUFSET-	104422	8456*	8480	8583	8652	8830	8860
BUILD	015170	8895*	8934				
CYRCNT	006400	8457*					
CLEAR	000000	8456*	8659*				
CHVRT	104412	8456*	8457				
CONVRT-	104411	8456*	8457				
CONMAX	017352	9251*	9252				
CO0	000400	8456*					
CO1	001000	8456*					
CO2	002000	8456*					
CO3	004000	8456*					
CR	000015	8456*	8457				
CRLF	000200	8456*	8457				
CSRMAP	011572	8457*					
CYCLE	010724	8456	8457*				
DATABP	007152	8457*					
DATAHD	007140	8457*					
DCLASH-	104417	8456*	8477	8580	8648	8811	
DCLR	000020	8456*	8457				
DDISP	177570	8456*					

CVDZB-D MACY11 30A(1052) 21-AUG-84 08:31 PAGE 31 7
CVDZBD.P11 21-AUG-84 08:28 CROSS REFERENCE TABLE

USER SYMBOLS

SETAPT	011436	8456	8457					
SETFLG	104406	8456						
SEVEN	000020	8456						
SEVENS	000060	8456						
SHIFT	104420	8456	8550	8637	8766	8809		
SILOM	020000	8456	8503	8509	8533	8608	8618	8707
SILOEN	010000	8456	8482	8557	8565	8587		
SIX	000010	8456						
SIXS	000050	8456						
SNRP	013346	8662	8748					
SPACHT	006401	8457						
STACK	001120	8456	8457					
STK.MT	177774	8456						
STOP	001446	8456	8457					
SV05	006136	8457						
SMR	001304	8456	8457					
SMREG	000176	8456						
SMD	000001	8456						
SM00	000001	8456						
SM01	000002	8456	8457					
SM02	000004	8456						
SM03	000010	8456						
SM04	000027	8456						
SM05	000040	8456						
SM06	000100	8456						
SM07	000200	8456						
SM08	000400	8456	8457					
SM09	001000	8456	8457					
SM1	000002	8456						
SM10	002000	8456	8457					
SM11	004000	8456						
SM12	010000	8456	8457					
SM13	020000	8456	8457					
SM14	040000	8456						
SM15	100000	8456						
SM2	000004	8456						
SM3	000010	8456						
SM4	000020	8456						
SM5	000040	8456						
SM6	000100	8456						
SM7	000200	8456						
SM8	000400	8456						
SM9	001000	8456						
S110	001000	8456						
S1200	003400	8456						
S134	001400	8456						
S150	002000	8456						
S1800	004000	8456						
S19200	007400	8456						
S2000	004400	8456						
S2400	005000	8456						
S300	002400	8456						
S3600	005400	8456						
S4800	006000	8456						
S50	000000	8456	8801	8804				
S600	003000	8456						

T600	002064	84560				
T7200	002102	84560				
T75	002052	84560				
T9600	G32104	84560				
UFD	- 000040	84570				
UFDSET	- 000001	84560				
VECMAP	012104	84570				
VPROR	002156	84560				
WRDCNT	006376	84570*				
WTBS.F	007130	84570				
XBX	006720	84570				
XCSR	004434	84570				
XERR	004456	84570				
XHEAD	010322	8456	84570			
XMYCNT	001400	84560	8834*	8860*	8871	
XMYLIN	001376	84560	8799*	8805*	8807	8828
XMYSRV	015046	8835	8860	88660		
XPASS	004450	84570				
XSTATQ	010412	8456	84570			
XTSTM	007314	84570				
XVEC	004442	84570				
XX	- 160210	84560				
YY	- 000500	84560				
ZZ	- 000020	84560				
%APTHD	001446	84560				
%ASTAT	- ***** U	8457				
%ATYC	005362	84570				
%ATY1	005336	84570				
%ATY3	005344	84570				
%ATY4	005354	84570				
%AUTOB	001300	84560				
%BASE	001174	84560*	8457*	9227	9229*	
%BNADR	001266	84560				
%BDDAT	001272	84560				
%CDM1	001200	84560	8457			
%CDM2	001202	84560	8457*			
%CHARC	005332	84570*				
%CHTAG	001244	84560				
%CM1	- 000006	84560				
%CM2	- 000014	84560				
%CM3	- 000006	84560				
%CM4	- 000005	84560				
%CPUOP	001146	84560				
%CRLF	001357	84560	8457			
%DDM0	001204	84560	8457			
%DDM1	001206	84560				
%DDM10	001230	84560				
%DDM11	001232	84560				
%DDM12	001234	84560				
%DDM13	001236	84560				
%DDM14	001240	84560				
%DDM15	001242	84560				
%DDM2	001210	84560				
%DDM3	001212	84560				
%DDM4	001214	84560				
%DDM5	001216	84560				

COMMEN	15580	84560								
ENDCOM	15700	84560								
ESCAPE	16840	84560								
GETPRI	13150	84560								
GETSMR	17530	84560								
MULT	45310	84560								
NEWST	16160	84560	8475	8578	8647	8790	8860			
PASEND	68550	8457								
POP	21270	84560	8457							
PRGEND	79200	8457								
PRGFRT	70050	8456								
PUSH	21190	84560	8457							
REPORT	54770	56770	84560							
SAVENT	11670	56750	8456							
SC	67940	8457								
SC1	68030	8457								
SETPRI	12840	84560								
SETUP	13380	84560								
SKIP	17170	84560								
SLASH	15110	84560								
STARS	14810	84560	8457	8475	8578	8647	8790	8860		
SMRSU	14510	84560								
TYPBIN	20640	84560								
TYPDEC	20340	84560								
TYPNAM	18060	84560								
TYPNUM	20010	84560								
TYPPCS	19540	84560								
TYPDCT	19170	84560								
TYPXT	18720	84560								
#ABORT	28380	56760	8457							
#BLFFE	69340	8457								
#CYCLE	75980	8457								
#EOP	68750	8457								
#GETFL	67380	8456								
#GETPA	67280	8456	8457							
#HEADE	64870	8456								
#INTSE	69810	8860								
#JUNK	69670	8456								
#PRESE	67180	8811								
#MSG	69000	8457								
#PARTS	83550	8860								
#SCOPE	68120	8457								
#SETFL	67440	8457								
#STAG	69880	8526								
#STAGF	69750									
#TRPDE	65120	8456								
#TSTN	68260	8475	8578	8647	8790	8860				
#VARIA	65250	8456								
#XZ	79010	8462	8474	8570	8577	8641	8646	8771	8789	8860
##CHRE	84560									
##CHTH	84560									
##ESCA	16970	84560								
##NEWT	16520	84560	8475	8578	8647	8790	8860			
##SKIP	17300	84560								
.EQUAT	1910	56760	8456							
.HEADE	670	56750	8456							

KT11	3330		
SETUP	12130	56750	
SWRMI	1080		
IAC1	50900	56770	8456
IAPT8	51330	56770	84560
IAPTH	53880	56770	8456
IAPTY	55600	56770	8457
IASTA	54330		
ICATC	9170	56750	
ICHTA	10260	84560	
IDB2D	47260		
IDB2O	48470		
IDIV	46300		
IEOP	21850	56750	8457
IERR0	26640	56760	
IERRT	29190		
IMULT	45680		
IPOWE	42440	56760	8457
IRAND	43180		
IRIDE	39180		
IRDOC	38280		
IREAD	34270		
IRAZ	49880		
ISAVE	39920		
ISB2D	48090		
ISB2O	49080		
ISCOF	24190	56760	8457
ISIZE	43700		
ISUPR	49450		
ITRAP	40930	56760	
ITYPB	33210		
ITYPD	32450		
ITYPE	30050	56750	8457
ITYPO	31500		
I4OCA	9550		
.1170	5110		

ABS. 017352 000

ERRORS DETECTED: 0

CVDZBO,CVDZBO/CRF=CVDZBO.MLB,CVDZBO.P11
 RUN TIME: 14 17 1 SECONDS
 RUN-TIME RATIO: 229/33=6.7
 CORE USED: 49K (97 PAGES)