



I D E N T I F I C A T I O N

-----

PRODUCT CODE: AC-8676E-MC

PRODUCT NAME: CZDUAEO DU11 OFFLINE LOGIC TESTS

RELEASE DATE: JUN 1978

MAINTAINER : DIAGNOSTICS

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 OF RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1973, 1978 BY DIGITAL EQUIPMENT CORPORATION

GENERAL DESCRIPTION

THIS DIAGNOSTIC CAN CHAIN 16 DU11'S. THIS MEANS THAT 16 DEVICES CAN BE SEQUENTIALLY EXERCISED. THE DIAGNOSTIC MAKES ONE PASS BEFORE PROCEEDING TO THE NEXT DEVICE, AND CONTINUES EXERCISING ALL DEVICES IN THIS FASHION UNTIL HALTED.

1. THE DU11 OFFLINE LOGIC TESTS VERIFY THAT ALL REGISTERS EXIST ,AND ALL RESPECTIVE BITS CAN BE MASTER CLEARED, READ, WRITTEN AND/OR READ/WRITTEN

2. REQUIREMENTS

PDP-11 FAMILY STANDARD COMPUTER WITH OR WITHOUT HARDWARE SWITCH REGISTER (LOC. 177570)

DU11 SYNCHRONOUS/ISOCRONOUS OPTION

ONE CONSOLE TELETYPE OR EQUIVALENT

2.2 STORAGE

THE PROGRAM LOADS AND RUNS IN 8K OF MEMORY.

3. LOADING PROCEDURE

THE STANDARD PROCEDURE FOR LOADING ABSOLUTE BINARY TAPES IS TO BE USED.

	STARTING ADDRESS FOR ABSOLUTE LOADER
4K	017500
8K	037500
12K	057500
16K	077500
20K	117500
24K	137500
28K	157500

4. STARTING PROCEDURE

\*\*\*\*\*

NOTE: BEFORE PROCEEDING IT IS IMPORTANT TO TO REALIZE IF ONE DOESNOT HAVE THE DU11 SET ,P TO THE DEFAULT PARAMETERS ( SEE SECTION

8 OF THIS DOCUMENT) , THEN ONE MUST  
SET SW00 = 1, AND ANSWER THE PARAMETER  
QUESTION ROUTINE.

\*\*\*\*\*

4.1 CONTROL SWITCH SETTINGS

NOTE: SOFTWARE SWITCH REGISTER IS DEFINED AS LOC. 176, WHILE  
THE SOFTWARE DISPLAY REGISTER IS DEFINED AS LOC. 174.

4.1.1 AFTER PROGRAM LOAD (INITIAL PROGRAM START)  
ALL CONSOLE SWITCHES DOWN

4.1.2 TO MODIFY DEVICE VECTOR AND CONTROL REGISTER ADDRESSES  
AFTER PROGRAM RESTART OR TO RUN MULTIPLE DEVICES

SW00-1

4.1.3 TO START PROGRAM AT SELECTED TEST AFTER A PROGRAM RESTART  
(ONLY IN SINGLE DEVICE TESTS)

SW01-1

4.1.4 TO LOCK ON SELECTED TEST AFTER A PROGRAM RESTART  
(ONLY IN SINGLE DEVICE TESTS)

SW02=1

NOTE1: IN GENERAL SW01 WILL BE USED WHEN SW02=1 IS USED

NOTE2: WITHOUT SW01=1 'LOCK ON TEST' WILL DEFAULT TO TEST 1

4.2 STARTING ADDRESS

THE STARTING ADDRESS FOR ALL TESTS IS 000200

THE RETARTING ADDRESS FOR ALL TESTS IS 000200

THE STARTING ADDRESS TO ENTER A SELECTED TEST IS 000200

THE STARTING ADDRESS TO LOCK ON TEST IS 000200

4.3 PROGRAM AND/OR OPERATOR ACTION

4.3.1 INITIAL PROGRAM START

4.3.1.1 LOAD PROGRAM INTO MEMORY WITH ABSOLUTE LOADER

4.3.1.2 LOAD ADDRESS 000200

4.3.1.3 CLEAR CONSOLE SWITCHES

4.3.1.4 PRESS START

4.3.1.5 THE PROGRAM WILL TYPE 'DU11 CZDUA-E TAPE A' (ONCE ONLY)

NOTE: IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING

WILL BE TYPED AFTER THE PROGRAM IDENTIFIES ITSELF:  
SWR=XXXXXX NEW= (REFER TO SECTION 5. FOR OPERATOR'S OPTION)

4.3.1.7 THE PROGRAM WILL TYPE 'R' TO INDICATE THAT IT IS ABOUT  
TO START TESTING ,AND THEN TESTING WILL BEGIN

4.3.2 PROGRAM RESTART WITH ALL SWITCHES DOWN

4.3.2.1 THE PROGRAM WILL TYPE 'R' AND WILL COMMENCE TESTING

4.3.3 PROGRAM RESTART WITH SW00=1

4.3.3.1 LOAD ADDRESS 000200

4.3.3.2 SET SW00-1

4.3.3.3 PRESS START

NOTE: IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING  
WILL BE TYPED AFTER THE PROGRAM IDENTIFIES ITSELF:  
SWR=XXXXXX NEW- (REFER TO SECTION 5. FOR OPERATOR'S OPTION)

4.3.3.4 THE PROGRAM WILL TYPE '' 1ST DEVICE: RECEIVER CONTROL REGISTER  
ADDRESS'' AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.5 TYPE IN THE ADDRESS OF THE FIRST RECEIVER CONTROL  
REGISTER ADDRESS OF THE DU11 TO BE TESTED  
FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ADDRESS IS TYPED ,THE PROGRAM WILL TYPE ''?''  
AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.4

4.3.3.6 THE PROGRAM WILL TYPE 'VECTOR ADDRESS-' AND WAIT FOR AN  
INPUT FROM THE TELETYPE KEYBOARD

4.3.3.7 TYPE IN THE BASE RECEIVER INTERRUPT VECTOR ADDRESS  
FOR THE DU11 TO BE TESTED FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ADDRESS IS TYPED ,THE PROGRAM WILL TYPE ''?''  
AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.6

4.3.3.8 THE PROGRAM WILL TYPE 'ARE YOU RUNNING MULTIPLE DEVICES ?'  
(Y OR N)-' AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.9 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED BY A  
<CARRIAGE RETURN>

IF AN INCORRECT ANSWER IS GIVEN, THE PROGRAM WILL TYPE ''?''  
AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.8

IF A 'NO' ANSWER IS GIVEN: JUMP TO SECTION 4.3.3.12  
IF A 'YES' ANSWER IS GIVEN: THE NEXT QUESTION IS ASKED





4.3.4.2 SET SW01=1

4.3.4.3 PRESS START

NOTE: IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING  
WILL BE TYPED AFTER THE PROGRAM IDENTIFIES ITSELF:  
SWR=XXXXXX NEW= (REFER TO SECTION 5. FOR OPERATOR'S OPTION)

4.3.4.4 THE PROGRAM WILL TYPE 'TEST PC-' AND WAIT FOR AN INPUT FROM  
THE TELETYPE KEYBOARD

4.3.4.5 TYPE IN THE ADDRESS OF THE TEST AT WHICH THE PROGRAM IS TO  
BE STARTED FOLLOWED BY A <CARRIAGE RETURN>

4.3.4.6 THE PROGRAM WILL TYPE 'R' TO INDICATE THAT IT HAS STARTED  
TESTING AT THE SELECTED TEST

NOTE: CARE MUST BE TAKEN WHEN THIS FEATURE IS USED  
SINCE THERE IS NO PROTECTION AGAINST SELECTING AN ADDRESS  
THAT IS IN THE MIDDLE OF A TEST

4.3.5 PROGRAM RESTART WITH SW02 -1  
NOTE: THIS WILL ONLY WORK WHEN A SINGLE DEVICE IS SELECTED  
SEE NOTE IN 4.3.4 FOR MORE DETAILS

4.3.5.1 LOAD ADDRESS 000200

4.3.5.2 SET SW02 -1  
NOTE: IT MAY BE ADVANTAGEOUS TO SET SW01 ' (OPTIONAL)

4.3.5.3 PRESS START

NOTE: IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING  
WILL BE TYPED AFTER THE PROGRAM IDENTIFIES ITSELF:  
SWR-XXXXXX NEW- (REFER TO SECTION 5. FOR OPERATOR'S OPTION)

4.3.5.4 THE PROGRAM WILL TYPE 'LOCK ON SELECTED TEST ? (Y OR N)-'  
AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.5.5 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED BY A  
<CARRIAGE RETURN>

IF A NO ANSWER IS GIVEN: THIS LOCK ON TEST WILL BE IGNORED  
AND THE PROGRAM WILL TYPE 'R' TO INDICATE THAT IT HAS STARTED  
TESTING AT TEST 1

4.3.5.6 IF A YES ANSWER WAS GIVEN: THE PROGRAM WILL ACT AS FOLLOWS...  
THE PROGRAM WILL TYPE 'R' TO INDICATE THAT IT HAS STARTED  
TESTING AT TEST 1 AND WILL REMAIN IN TEST 1 UNTIL HALTED  
OR IF ANY KEY IS STRUCK ON THE TELETYPE, THE PROGRAM  
WILL FREEZE ON THE NEXT TEST UNTIL A KEY IS STRUCK ON  
THE TELETYPE AND SO FORTH THRU THE PROGRAM. IF SW01 =1 IT  
WILL PERFORM AS IN SECTION 4.3.4 ALLOWING ONE TO FREEZE  
ON A SELECTED TEST RATHER THAN DEFAULTING TO TEST 1

#### 4.4 STATUS MAP

---

THE STATUS MAP IS AN AREA OF THE DU11 DIAGNOSTICS, WHICH WILL ALLOW THE TRANSFER OF PARAMETERS BETWEEN DIAGNOSTICS. IF YOU WISH TO TEST A DU11, WHICH IS NOT AT THE DEFAULT VALUES, YOU NEED ONLY GO THROUGH THE TEDIOUS QUESTIONING AND ANSWERING ROUTINE ONCE.

THE FOLLOWING COMBINATIONS OF SWITCH REGISTER SETTINGS WILL ALLOW YOU ACCESS TO THE STATUS MAP.

- 1) SW07=1
- 2) START AT 200
- 3) THE DIAGNOSTIC WILL GO TO THE STATUS MAP AND BYPASS ALL OF THE QUESTIONING ROUTINE.

NOTE: IT IS EXTREMELY IMPORTANT THAT EITHER YOU HAVE JUST ANSWERED THESE QUESTIONS DURING A PRIOR DIAGNOSTIC OR THAT YOU HAVE MANUALLY ENTERED THE CORRECT VALUES FOR VECTOR ADDRESSES ETC., IN THE AREA DESIGNATED FOR THE STATUS MAP. IT IS IMPORTANT THAT THIS BE PERFORMED BEFORE STARTING AT 200.

THE DIAGNOSTIC HAS NO METHOD TO DETERMINE THAT THE STATUS MAP HAS INDEED BEEN LOADED CORRECTLY. THE DIAGNOSTIC ASSUMES THAT WHEN SW07=1 THE VALUES IN THE STATUS MAP ARE THE VALUES TO BE USED. THESE VALUES CAN BE THE WRONG VALUES, BUT THE DIAGNOSTIC WILL NOT REALIZE THAT A MISTAKE HAS BEEN MADE.

IF BOTH SW07 AND SW00 (SWITCH REGISTER SWITCHES) ARE SET (EQUAL TO 1), THE PROGRAM WILL IGNORE SW00 AND SEEING SW07 SET, THE VALUES FROM THE STATUS MAP WILL BE USED. TO USE THE DEFAULT VALUES FOR THE DU11'S THE OPERATOR MUST SET SW00=0 AND SW07=0. THE USE OF SW00 IS EXPLAINED IN GREATER DETAIL IN SECTION 4.3 OF THIS DOCUMENT.

THE FIRST TIME A PROGRAM IS LOADED OR THE FIRST TIME A PROGRAM IS ALTERED VIA THE PARAMETER RESELECTION QUESTION AND ANSWER ROUTINE, A PARTIAL STATUS MAP WILL BE PRINTED. THIS MAP WILL BE PRINTED ONCE FOR ANY COMBINATION OF SWITCHES EXCEPT SW01. RESTARTING THE PROGRAM WILL NOT PRINT OUT A MAP UNLESS THE PROGRAM PARAMETERS ARE BEING RESELECTED BY PUTTING SW00-1.(ON)

THE MAP WILL LOOK LIKE:

STATUS MAP

1300/ 177777

1302/ 000000

1304/ 177777

THE BYTES ARE DEFINED AS FOLLOWS:

1300 THE NUMBER OF SYNCHRONOUS CHARACTERS REQUIRED FOR  
SYNCHRONIZATION.  
1301 SEC TRANSMIT JUMPER  
1302 SEC RECEIVER JUMPER  
1303 OPTIONAL JUMPER  
1304 MULTIPLE DEVICES (NO=0 , YES- 1)  
1305 EXTERNAL MODEM BYPASS? (NO=0 , YES- 1)

IF THE BYTE IS 0 , THE JUMPER IS NOT CONNECTED  
AND IF THE BYTE IS 377 ETC. THE JUMPER SHOULD BE CONNECTED.

#### 5. OPERATING PROCEDURE

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH  
REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS  
THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER.  
IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES  
AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH  
REGISTER (LOC. 176) IS USED.

#### CONTROL :

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH  
REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY  
DOING THE FOLLOWING:

- 1) TYPE CONTROL G <^G>: THIS WILL ALLOW THE TTY TO ENTER DATA INTO  
LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR-XXXXXXNEW- (XXXXXX IS THE OCTAL CONTENTS  
OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE ''NEW-'' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE  
OF THE FOLLOWING AT THE TTY:
  - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>.  
(ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS  
WILL BE ALLOWED)  
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH  
REGISTER CONTENTS WILL NOT BE CHANGED.
  - B) IF A CONTROL U <^U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU  
BACK TO STEP 2.

#### 5.1 OPERATIONAL SWITCH SETTINGS

SW15	1	HALT ON ERROR
SW14	-1	LOOP ON CURRENT TEST
SW13	1	INHIBIT ERROR TYPEOUT
SW11	1	INHIBIT ITERATIONS

SW10 =1 ESCAPE TO NEXT TEST ON ERROR  
SW08 =1 LOOP ON ERROR  
SW07 =1 USE STATUS MAP PARAMETERS  
SW02 -1 LOCK ON TEST  
SW01 =1 RESTART PROGRAM AT SELECTED TEST  
SW00 -1 RESELECT VECTOR AND CONTROL REGISTER ADDRESSES  
&PARAMETERS AFTER A PROGRAM RESTART  
TO INHIBIT 'END OF PASS' TYPEOUT - TURN TELETYPE OFF

6. ERRORS

6.1 ERROR HALTS  
THERE ARE FOUR DISTINCT ERROR TYPEOUTS

NOTE: IF THE SOFTWARE SWITCH REGISTER IS TO BE CHANGED AFTER A HALT  
THE OPERATOR IS REQUIRED TO TYPE A <^G> BEFORE DEPRESSING CONTINUE.  
THE FOLLOWING WILL BE TYPED:  
SWR-XXXXXX NEW- (REFER TO SECTION 5. FOR OPERATOR OPTION)

6.1.1 PC+2 = ERROR PC  
WHERE PC +2 IS THE ADDRESS OF THE CALL TO THE ERROR HANDLER +2  
REFER TO THE ABOVE 'HLT' IN DIAGNOSTIC FOR ERROR DESCRIPTION

(CHECK ADDRESS @ RXCSR: TO LOCATE THE DEVICE PRESENTLY UNDER  
TEST WHEN RUNNING MULTIPLE DEVICES)

6.1.2 PC +2 - REGISTER ERROR PC  
REGISTER EXPECTED ACTUAL  
16XXXX YYYYYY ZZZZZZ

WHERE 16XXXX IS THE ADDRESS OF THE FAILING DEVICE REGISTER

WHERE YYYYYY IS THE EXPECTED CONTENTS OF THAT REGISTER

WHERE ZZZZZZ IS THE ACTUAL CONTENTS OF THAT REGISTER

6.1.3 PC +2 RECEIVER ERROR PC  
REGISTER EXPECTED ACTUAL  
16XXXX YYYYYY ZZZZZZ

WHERE 16XXXX IS THE ADDRESS OF THE FAILING RECEIVER (RXDBUF) REGISTER

WHERE YYYYYY IS THE EXPECTED DATA CONTENTS OF THAT REGISTER

WHERE ZZZZZZ IS THE ACTUAL DATA CONTENTS OF THAT REGISTER

6.1.4 PC +2 = TRANSMITTER ERROR PC  
REGISTER EXPECTED ACTUAL  
16XXXX YYYYYY ZZZZZZ

WHERE 16XXXX IS THE ADDRESS OF THE FAILING TRANSMITTER (TXCSR) REGISTER

WHERE YYYYYY IS THE EXPECTED CONTENTS OF THAT REGISTER

WHERE ZZZZZZ IS THE ACTUAL CONTENTS OF THAT REGISTER

6.1.5 ERROR DESCRIPTIONS  
SEE LISTINGS FOR DETAILS OF ERRORS

6.2 ERROR RECOVERY

6.2.1 SW15 =0  
IF THE PROGRAM IS RUN WITH SW15 =0 ,NO OPERATOR ACTION IS  
REQUIRED TO CONTINUE TESTING

6.2.2 SW15 =1  
IF THE PROGRAM IS RUN WITH SW15 =1 ,TO CONTINUE TESTING  
AFTER THE PROGRAM HAS HALTED ,PRESS THE PROCESSOR  
CONSOLE "CONTINUE SWITCH"

NOTE: THE PC + 2 OF THE 'HLT' WILL BE DISPLAYED IN THE DATA LIGHTS

6.2.3 ILLEGAL INTERRUPTS  
IF AN INTERRUPT OCCURS TO A VECTOR ADDRESS NOT SELECTED  
DURING PROGRAM INITIALIZATION, THE PROGRAM WILL HALT IN  
THE TRAPCATCHER. THE ADDRESS AT WHICH THE PROGRAM  
HALTS IS 2 GREATER THAN THE ADDRESS TO WHICH THE INTERRUPT  
OCCURED. THE PROGRAM MUST BE RESTARTED AT 000200 TO  
RECOVER FROM THIS ERROR.

6.2.4 ADDITIONAL TROUBLESHOOTING AIDS ERRCNT: & PASCNT:  
CHECK THESE TWO TAG LOCATIONS FOR TOTAL # OF ERRORS AND PASSES RESPECTIVELY.  
LOADING 000200 AND RESTARTING WILL CLEAR THESE LOCATIONS.

6.3 END OF PASS ROUTINE  
THIS TYPEOUT IS MENTIONED HERE FOR CONVENIENCE  
IT IS IN THE FORM:

END OF PASS TAPE Y  
16XXXX - DEVICE

WHERE Y IS THE TAPE LOADED

WHERE 16XXXX IS THE DEVICE'S BASE REGISTER ADDRESS

TO INHIBIT THIS TYPEOUT - TURN TELETYPE OFF

7. RESTRICTIONS

7.1 MULTIPLE DEVICES  
UP TO 16(10) DEVICES MAY BE TESTED. HOWEVER, THEY  
MUST HAVE CONTIGUOUS ADDRESSES AND VECTORS

NOTE: IF ALL DEVICES UNDER TEST HAVE THE SAME INTERRUPT VECTOR  
YOU CAN CHANGE 'ZERO: ADD #10,BASEIV ;NEXT BLOCK  
(VECTORS)' TO 'ZERO: ADD #0,BASEIV';  
THEREBY THE VECTOR ADDRESSES WILL NOT BE  
UPDATED AFTER EACH PASS.

7.2 DISQUALIFYING DEVICES WHEN RUNNING MULTIPLE DEVICES



DO YOU HAVE THE EXTERNAL MODEM BYPASS JUMPER  
CONNECTOR ON (H315)- YES

JMRBY: 377

9. PROGRAM DESCRIPTION
- 9.1 THIS PROGRAM PERFORMS THE OFFLINE LOGIC BIT BANGING  
OF THE DEVICE  
SEE LISTING FOR DETAILS
10. FLOW CHARTS: RECEIVER FLOW, TRANSMITTER FLOW, TRANSMITTER & RECEIVER FLOW
11. LISTINGS

674  
675 000000\* 000000G D

676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709

```
.ENABLE ABS

;DU11 CZDUA-E TAPE A
;COPYRIGHT 1973, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

;STARTING PROCEDURE
;LOAD PROGRAM
;PRESS START
;PROGRAM WILL TYPE 'DU11 CZDUA-E TAPE A '
;PROGRAM WILL TYPE 'R' TO INDICATE THAT TESTING HAS STARTED
;AT THE END OF A PASS, PROGRAM WILL TYPE 'END OF PASS TAPE A'
;AND THEN RESUME TESTING

;SWITCH REGISTER OPTIONS
SW15=100000          ;=1,HALT ON ERROR
SW14=40000           ;=1,LOOP ON CURRENT TEST
SW13=20000           ;=1,INHIBIT ERROR TYPEOUT
SW12=10000
SW11=4000            ;-1,INHIBIT ITERATIONS
SW10=2000            ;=1,ESCAPE TO NEXT TEST ON ERROR
SW09=1000            ;=1,LOOP WITH CURRENT DATA
SW08=400             ;=1,LOOP ON ERROR
SW07=200             ;+ =1, USE STATUS MAP
SW06=100
SW05=40
SW04=20
SW03=10
SW02=4               ;LOCK ON TEST SELECT
SW01=2               ;RESTART PROGRAM AT SELECTED TEST
SW00=1               ;RESELECT VECTOR AND CONTROL REGISTER
                      ;ADDRESS AFTER PROGRAM RESTART
```

```

710
711
712
713      000000      R0=%0           :GENERAL REGISTER
714      000001      R1=%1           :GENERAL REGISTER
715      000002      R2=%2           :GENERAL REGISTER
716      000003      R3=%3           :GENERAL REGISTER
717      000004      R4=%4           :GENERAL REGISTER
718      000005      R5=%5           :GENERAL REGISTER
719      000006      SP=%6           :PROCESSOR STACK POINTER
720      000007      PC=%7           :PROGRAM COUNTER
721
722      :LOCATION EQUIVALENCIES
723
724      177570      DSWR=177570      :HARDWARE SWITCH REGISTER LOC.
725      177570      DLIGHTS=177570  :HARDWARE DISPLAY REGISTER LOC.
726      177776      PS=177776      :PROCESSOR STATUS WORD
727      001100      STACK=1100     :START OF PROCESSOR STACK
728
729      :INSTRUCTION DEFINITIONS
730
731      005746      PUSH1SP=5746     :DECREMENT PROCESSOR STACK 1 WORD =TST -(SP)
732      005726      POP1SP=5726     :INCREMENT PROCESSOR STACK 1 WORD -TST (SP)+
733      010046      PUSHRO=10046     :SAVE R0 ON STACK =MOV R0,-(SP)
734      012600      POPRO=12600     :RESTORE R0 FROM STACK =MOV (SP)+,R0
735      024646      PUSH2SP=24646   :DECREMENT STACK TWICE =CMP -(SP),-(SP)
736      022626      POP2SP=22626    :INCREMENT STACK TWICE =CMP (SP)+,(SP)+
737      .EQUIV EMT,HLT :BASIC DEFINITION OF ERROR CALL
738
739
740      100000      BIT15=100000
741      040000      BIT14=40000
742      020000      BIT13=20000
743      010000      BIT12=10000
744      004000      BIT11=4000
745      002000      BIT10=2000
746      001000      BIT9=1000
747      000400     BIT8=400
748      000200     BIT7=200
749      000100     BIT6=100
750      000040     BIT5=40
751      000020     BIT4=20
752      000010     BIT3=10
753      000004     BIT2=4
754      000002     BIT1=2
755      000001     BIT0=1
756
757      :PROCESSER LEVELS
758      000340     LEVEL7=340
759      000300     LEVEL6=300
760      000240     LEVEL5=240
761      000200     LEVEL4=200
762      000140     LEVEL3=140
763      000100     LEVEL2=100
764      000040     LEVEL1=040
765      000000     LEVEL0=000

```

```

766 ;REGISTER DEFINITIONS
767 ;RXCSR BIT DEFINITIONS
768 100000 DSC=BIT15 ;DATA SET CHANGE
769 040000 RING=BIT14 ;RING
770 020000 CTS=BIT13 ;CLR TO SEND
771 010000 CARDET=BIT12 ;CARRIER DETECT
772 004000 RECACT=BIT11 ;REC ACTIVE
773 002000 SRD=BIT10 ;SEC REC DATA
774 001000 DSR=BIT9 ;DATA SET RDY
775 000400 STPSYN=BIT8 ;STRIP SYNC
776 000200 RXDONE=BIT7 ;REC DONE
777 000100 RINTEN=BIT6 ;REC INTR ENABLE
778 000040 DSINTE=BIT5 ;DSC INTR ENABLE
779 000020 SYN SCH=BIT4 ;SYNC SEARCH
780 000010 STD=BIT3 ;SEC XMIT DATA
781 000004 RTS=BIT2 ;REQ TO SEND
782 000002 DTR=BIT1 ;DATA TERM RDY
783 000001 VOID=BIT0
784 ;RXDBUF BIT DEFINITIONS
785 100000 RXERR=BIT15 ;REC ERROR
786 040000 OVRRUN=BIT14 ;OVERRUN
787 020000 FRMERR=BIT13 ;FRAME ERROR
788 010000 PARER=BIT12 ;PARITY ERROR
789 ;PARCSR BIT DEFINITIONS
790 001000 PAREN=BIT9 ;PARITY ENABLE
791 000400 EVPAR=BIT8 ;EVEN PARITY SENSE
792 ;PARCSR WRD DEFINITIONS
793 030000 SYNINT=30000 ;SYNC EXTERNAL MODE
794 020000 SYNEXT=20000 ;SYNC INTERNAL MODE
795 000000 ISYMOD=0 ;ISOC MODE
796 000000 FIVE=0 ;WORD LENGTH 5 BITS
797 002000 SIX=2000 ;WORD LENGTH 6 BITS
798 004000 SEVEN=4000 ;WORD LENGTH 7 BITS
799 006000 EIGHT=6000 ;WORD LENGTH 8 BITS
800 000000 NOPAR=0 ;NO PARITY
801 001000 ODDPAR=1000 ;ODD PARITY
802 001400 EVEPAR=1400 ;EVEN PARITY
803 ;TXCSR BIT DEFINITIONS
804 100000 DNA=BIT15 ;DATA NOT AVAILABLE
805 040000 MTDATA=BIT14 ;MAINT DATA
806 020000 CLK=BIT13 ;CLK
807 002000 BITW=BIT10 ;BIT WINDOW
808 000400 MRESET=BIT8 ;MASTER RESET
809 000200 TXDONE=BIT7 ;XMIT DONE
810 000100 TXINTE=BIT6 ;XMIT INTR ENABLE
811 000040 DNAINTE=BIT5 ;DNA INTR ENAB
812 000020 SEND=BIT4 ;SEND
813 000010 HDXEN=BIT3 ;HDX/FDX
814 000001 BREAK=BIT0 ;BREAK
815 ;TXCSR WRD DEFINITIONS
816 000000 USER=0 ;USER MODE
817 004000 MINT=4000 ;MAINT INT MODE
818 010000 MEXT=10000 ;MAINT EXT MODE
819 014000 SYSTST=14000 ;SYSTEM TEST MODE
820 ;TRAPCATCER FOR ILLEGAL INTERRUPTS
  
```

```

821                                ;STANDARD INTERRUPT VECTORS
822
823
824                                .-24
825 000024 016250                .PFAIL                    ;POWER FAIL HANDLER
826 000026 000340                340                    ;SERVICE AT LEVEL 7
827 000030 016000                .HLT                    ;ERROR HANDLER
828 000032 000340                340                    ;SERVICE AT LEVEL 7
829 000034 015746                .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
830 000036 000340                340                    ;SERVICE AT LEVEL 7
831
832                                ;SOFTWARE SWITCH REGISTER
833
834                                .-174
835 000174 000000                DISPRG: .WORD 0        ;SOFTWARE DISPLAY REG.
836 000176 000000                SWREG:  .WORD 0        ;SOFTWARE SWITCH REGISTER
837 000200 000167 001214        JMP      .START        ;GO TO START OF PROGRAM
838
839
840
841                                .-1100
842
843                                ;INDIRECT POINTERS
844
845 001100 177570                SWR:      177570        ;SWITCH REGISTER POINTER
846 001102 177570                LIGHTS: 177570        ;DISPLAY REGISTER POINTER
847 001104 177560                TKCSR:   177560        ;TELETYPE KEYBOARD CONTROL REGISTER
848 001106 177562                TKDBR:   177562        ;TELETYPE KEYBOARD DATA BUFFER
849 001110 177564                TPCSR:   177564        ;TELEPRINTER CONTROL REGISTER
850 001112 177566                TPDBR:   177566        ;TELEPRINTER DATA BUFFER
851
852                                ;PROGRAM CONTROL PARAMETERS
853
854 001114 000000                RTRN:    0              ;SCOPE ADDRESS FOR LOOP ON TEST
855 001116 000000                NEXT:    0              ;ADDRESS OF NEXT TEST TO BE EXECUTED
856 001120 000000                LOCK:    0              ;ADDRESS FOR LOCK ON CURRENT DATA
857 001122 000000                ICOUNT:  0              ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
858 001124 000000                LPCNT:   0              ;NUMBER OF ITERATIONS COMPLETED
859 001126 000000                TSTNO:   0              ;NUMBER OF TEST IN PROGRESS
860 001130 000000                PASCNT:  0              ;NUMBER OF PASSES COMPLETED
861 001132 000000                ERRCNT:  0              ;TOTAL NUMBER OF ERRORS
862 001134 000000                LSTERR:  0              ;PC OF LAST ERROR CALL
863
864                                ;PROGRAM VARIABLES
865
866 001136 000020                HOLD:    20            ;TEMPORARY STORAGE=DELAY TIME FOR CABLES
867 001140 000000                SHIFT:   0            ;TEMPORARY STORAGE= # OF SHIFTS PER CHAR
868 001142 000000                COUNT:   0            ;TEMPORARY STORAGE= # OF TIMES A CHAR WILL BE SENT
869 001144 000000                TEMP1:   0            ;TEMPORARY STORAGE
870 001146 000000                TEMP2:   0            ;TEMPORARY STORAGE
871 001150 000000                TEMP3:   0            ;TEMPORARY STORAGE
872 001152 000000                TEMP4:   0            ;TEMPORARY STORAGE
873 001154 000000                TEMP5:   0            ;TEMPORARY STORAGE
874 001156 000000                SAVR0:   0            ;R0 STORAGE
875 001160 000000                SAVR1:   0            ;R1 STORAGE
876 001162 000000                SAVR2:   0            ;R2 STORAGE

```

877 001164 000000  
878 001166 000000  
879 001170 000000  
880 001172 000000  
881 001174 000000

SAVR3: 0  
SAVR4: 0  
SAVR5: 0  
SAVSP: 0  
SAVPC: 0

:R3 STORAGE  
:R4 STORAGE  
:R5 STORAGE  
:STACK POINTER STORAGE  
:PROGRAM COUNTER STORAGE

```
882 ;PROGRAM CONVERSATIONAL PARAMETERS
883 001176 377 SYNCNO: .BYTE 377 ;# OF SYNC CHARS REQ'D FOR SYNC'ZATION
884 001177 377 SEXMIT: .BYTE 377 ;SEC XMIT JUMPER 'IN'
885 001200 377 SEREC: .BYTE 377 ;SEC REC JUMPER 'IN'
886 001201 377 OPTCLR: .BYTE 377 ;OPTIONAL JUMPER CLR 'IN'
887 001202 000 MULTD: .BYTE 0 ;NO MULTIPLE DEVICE FLAG
888 001203 377 JMRBY: .BYTE 377 ;EXTERNAL MODEM BYPASS JUMPER 'IN'
889 .EVEN
890
891 ;PROGRAM MULTIPLE DEVICE PARAMETERS
892 001204 000000 BASEADD: 0 ;PROG CONTROLLED 1ST DEVICE ADDR
893 001206 000000 KEEPADD: 0 ;SAVED 1ST DEVICE ADDR
894 001210 000000 LASTADD: 0 ;LAST DEVICE RXCSR ADDR
895 001212 000000 BASEIV: 0 ;PROG CONTROLLED IV
896 001214 000000 KEEPIV: 0 ;SAVED INTR VECTOR
897 001216 000000 ACTREG: 0 ;ACTIVE REGISTER,,,MODIFY THIS
898 ;LOCATION TO DISQUALIFY OR QUALIFY
899 ;DEVICES (1= RUN,,,0= DON'T RUN)
900 001220 000000 ROTADD: 0 ;ROTATING POINTER FOR ACTREG..POINTS
901 ;TO DEVICE PRESENTLY UNDER TEST WHEN RUNNING MULTIPLE DE
902 ;*****
903
904 ; THESE ARE STORAGE FOR THE STATUS MAP PRINT OUT
905 001222 000000 FLAG:0 ; FLAGS FOR STATUS MAP PRINT OUT (SSP)
906 001224 000000 HOLD0:0 ; HOLDS R0 IN STATUS MAP PRINT
907 001226 000000 HOLD1:0 ; R1 ETC.
908 001230 000000 COUNT1:0 ; FOR COUNTING 3 WORDS
909 001232 000002 TABLE : 2 ; FOR CONVRT ROUTINE
910 001234 003006 0 3006
911 001236 000000 0 3006
912 001240 003006 0
913 001242 000000 0
914 ;*****
915 ;PROGRAM CONTROL FLAGS
916
917
918 001244 000 INIFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG
919 001245 000 STFLG: .BYTE 0 ;TEST START FLAG
920 001246 000 ERRFLG: .BYTE 0 ;ERROR OCCURED FLAG
921 001247 000 LOKFLG: .BYTE 0 ;LOCK ON CURRENT TEST FLAG
922
923 .EVEN
924
925 ;***** STATUS MAP *****
926
927 -1300
928 001300 000001 STATUS: NOSYNC: .BLKB 1 ;SYNC CHARS
929 001301 000001 MITSEX: .BLKB 1 ;XMIT JUMPER
930 001302 000001 RESEC: .BLKB 1 ;REC SEC JUMPER
931 001303 000001 CLROPT: .BLKB 1 ;OPTIONAL JUMPER
932 001304 000001 DMULT: .BLKB 1 ;MULTIPLE DEVICE FLAG
933 001305 000001 BYJMR: .BLKB 1 ;EXTERNAL MODEM
934
935 ; MULTIPLE DEVICE PARAMETERS
936
937 001306 000001 ADDRBASE: .BLKW 1 ;PROG CONTROLLED 1ST DEVICE ADDR
```



















```
1442 004266 032777 000004 014414 BIT #RTS,@RXCSR ;TEST THIS BIT
1443 004274 001001 BNE 64$ ;BR IF '1'
1444 004276 104000 HLT ;THIS BIT SHOULD BE SET
1445 004300 64$:
1446 004300 042777 000004 014402 BIC #RTS,@RXCSR ;CLR THIS BIT
1447 004306 032777 000004 014374 BIT #RTS,@RXCSR ;TEST THIS BIT
1448 004314 001401 BEQ 65$ ;BR IF '0'
1449 004316 104000 HLT ;THIS BIT SHOULD BE CLR
1450 004320 65$:
1451 ;NOW SET THIS BIT
1452 004320 052777 000004 014362 BIS #RTS,@RXCSR
1453 004326 052777 000400 014370 BIS #MRESET,@TXCSR ;MASTER RESET
1454 ;;CHECK EXISTANCE OF OPTIONAL CLEAR JUMPER
1455 ;;
1456 004334 105767 174641 TSTB OPTCLR ;TEST FLAG
1457 004340 100006 BPL 1$ ;OPTIONAL CLR JUMPER IS NOT IN
1458 004342 032777 000004 014340 BIT #RTS,@RXCSR ;TEST THIS BIT
1459 004350 001401 BEQ 66$ ;BR IF '0'
1460 004352 104000 HLT ;CHECK OUT MASTER RESET LOGIC
1461 004354 66$:
1462 004354 000405 BR 2$ ;JMP AROUND
1463 004356 032777 000004 014324 1$: BIT #RTS,@RXCSR ;TEST THIS BIT
1464 004364 001001 BNE 67$ ;BR IF '1'
1465 004366 104000 HLT ;CHECK OUT OPTIONAL CLR JUMPER
1466 004370 67$:
1467 004370 000240 2$: NOP
1468 004372 104400 SCOPE
1469 ;;THIS TEST PERFORMS MASTER RESET TESTING &
1470 ;;TESTING OF READ/WRITE BIT STD
1471 ;;
1472 004374 012767 000011 174524 TST9: MOV #9,TSTNO ;SAVE THIS
1473 004402 012767 004524 174506 MOV #TST10,NEXT ;GO TO THIS TEST WHEN THRU
1474 004410 052777 000010 014272 BIS #STD,@RXCSR ;SET THIS BIT
1475 004416 032777 000010 014264 BIT #STD,@RXCSR ;TEST THIS BIT
1476 004424 001001 BNE 64$ ;BR IF '1'
1477 004426 104000 HLT ;THIS BIT SHOULD BE SET
1478 004430 64$:
1479 004430 042777 000010 014252 BIC #STD,@RXCSR ;CLR THIS BIT
1480 004436 032777 000010 014244 BIT #STD,@RXCSR ;TEST THIS BIT
1481 004444 001401 BEQ 65$ ;BR IF '0'
1482 004446 104000 HLT ;THIS BIT SHOULD BE CLR
1483 004450 65$:
1484 ;NOW SET THIS BIT
1485 004450 052777 000010 014232 BIS #STD,@RXCSR
1486 004456 052777 000400 014240 BIS #MRESET,@TXCSR ;MASTER RESET
1487 ;;CHECK EXISTANCE OF OPTIONAL CLEAR JUMPER
1488 ;;
1489 004464 105767 174511 TSTB OPTCLR ;TEST FLAG
1490 004470 100006 BPL 1$ ;OPTIONAL CLR JUMPER IS NOT IN
1491 004472 032777 000010 014210 BIT #STD,@RXCSR ;TEST THIS BIT
1492 004500 001401 BEQ 66$ ;BR IF '0'
1493 004502 104000 HLT ;CHECK OUT MASTER RESET LOGIC
1494 004504 66$:
1495 004504 000405 BR 2$ ;JMP AROUND
1496 004506 032777 000010 014174 1$: BIT #STD,@RXCSR ;TEST THIS BIT
1497 004514 001001 BNE 67$ ;BR IF '1'
```





















































```

2954 014034 042777 020000 004662 BIC #CLK,@TXCSR :POKE CLK DOWN
2955 014042 052777 020000 004654 BIS #CLK,@TXCSR :POKE CLK UP
2956 014050 016703 004640 MOV RXDBUF,R3 :SET UP FOR ERROR MESSAGE
2957 014054 012700 000000 MOV #0,R0 :EXPECTED
2958 014060 012767 000007 165052 MOV #7,SHIFT :# OF SHIFTS
2959 014066 012767 000100 165050 MOV #100,TEMP1 :DATA CHAR
2960 014074 004767 004314 JSR PC,RPOKE :SHIFT IN THIS CHAR
2961 014100 105777 004604 TSTB @RXCSR ;RXDONE ?
2962 014104 100401 BMI 64$
2963 014106 104000 HLT ;RXDONE SHOULD BE SET
2964 014110
2965 014110 017701 004600 64$: MOV @RXDBUF,R1 ;ACTUAL
2966 014114 020001 CMP R0,R1 ;COMPARE EXPECTED VS. ACTUAL
2967 014116 001401 BEQ 65$
2968 014120 104002 HLT 2 ;RECEIVED DATA DID NOT MATCH
2969 ;EXPECTED DATA - CHECK MAINT DATA
2970 ;OR RECEIVER LOGIC
2971 014122 65$:
2972 014122 012767 000007 165010 MOV #7,SHIFT :# OF SHIFTS
2973 014130 012767 000100 165006 MOV #100,TEMP1 :DATA CHAR
2974 014136 004767 004252 JSR PC,RPOKE :SHIFT IN THIS CHAR
2975 ;NOW SHIFT IN A SECOND CHARACTER WITHOUT READING RXDBUF
2976 014142 012767 000007 164770 MOV #7,SHIFT :# OF SHIFTS
2977 014150 012767 000100 164766 MOV #100,TEMP1 :DATA CHAR
2978 014156 004767 004232 JSR PC,RPOKE :SHIFT IN THIS CHAR
2979 014162 012700 140000 MOV #140000.0,R0 :EXPECTED DATA PLUS
2980 ;RXERR & OVRRUN
2981 014166 017701 004522 MOV @RXDBUF,R1 ;ACTUAL
2982 014172 020001 CMP R0,R1 ;COMPARE EXP VS. ACT
2983 014174 001401 BEQ 66$
2984 014176 104002 HLT 2 ;SPECIFICALLY LOOK AT RXERR &
2985 ;OVRRUN BITS...THEY BOTH SHOULD BE SET
2986 014200 66$:
2987 014200 104400 SCOPE
2988

```



























```

3717 020744 036440 042440 051122 EM0: .ASCIIZ / ERROR PC/
3718 020752 051117 050040 000103
3719 020760 036440 051040 043505 EM1: .ASCIIZ / = REGISTER ERROR PC/<15><12><1>/REGISTER ,/
3720 020766 051511 042524 020122
3721 020774 051105 047522 020122
3722 021002 041520 005015 051001
3723 021010 043505 051511 042524
3724 021016 020122 000040
3725 021022 036440 051040 041505 EM2: .ASCIIZ / - RECEIVER ERROR PC/<15><12><1>/REGISTER /
3726 021030 044505 042526 020122
3727 021036 051105 047522 020122
3728 021044 041520 005015 051001
3729 021052 043505 051511 042524
3730 021060 020122 000040
3731 021064 036440 052040 040522 EM3: .ASCIIZ / TRANSMITTER ERROR PC/<15><12><1>/REGISTER /
3732 021072 051516 044515 052124
3733 021100 051105 042440 051122
3734 021106 051117 050040 006503
3735 021114 000412 042522 044507
3736 021122 052123 051105 020040
3737 021130 000
3738 ;DATA HEADERS FOR ERROR MESSAGES
3739 021131 105 050130 041505 DH1: .ASCIIZ /EXPECTED ACTUAL,
3740 021136 042524 020104 040440
3741 021144 052103 040525 000114
3742 .EVEN
3743 ;DATA TABLES FOR ERROR MESSAGES
3744 021152 000003 D1: 3
3745 021154 006 004 .BYTE 6,4
3746 021156 001164 SAVR3 :REGISTER
3747 021160 006 004 .BYTE 6,4
3748 021162 001156 SAVR0 :EXPECTED DATA
3749 021164 006 002 .BYTE 6,2
3750 021166 001160 SAVR1 :ACTUAL DATA
3751 000001 .END
    
```





















