DQ11

TESTS AND BCC TESTS

MD-11-DZDQE-C

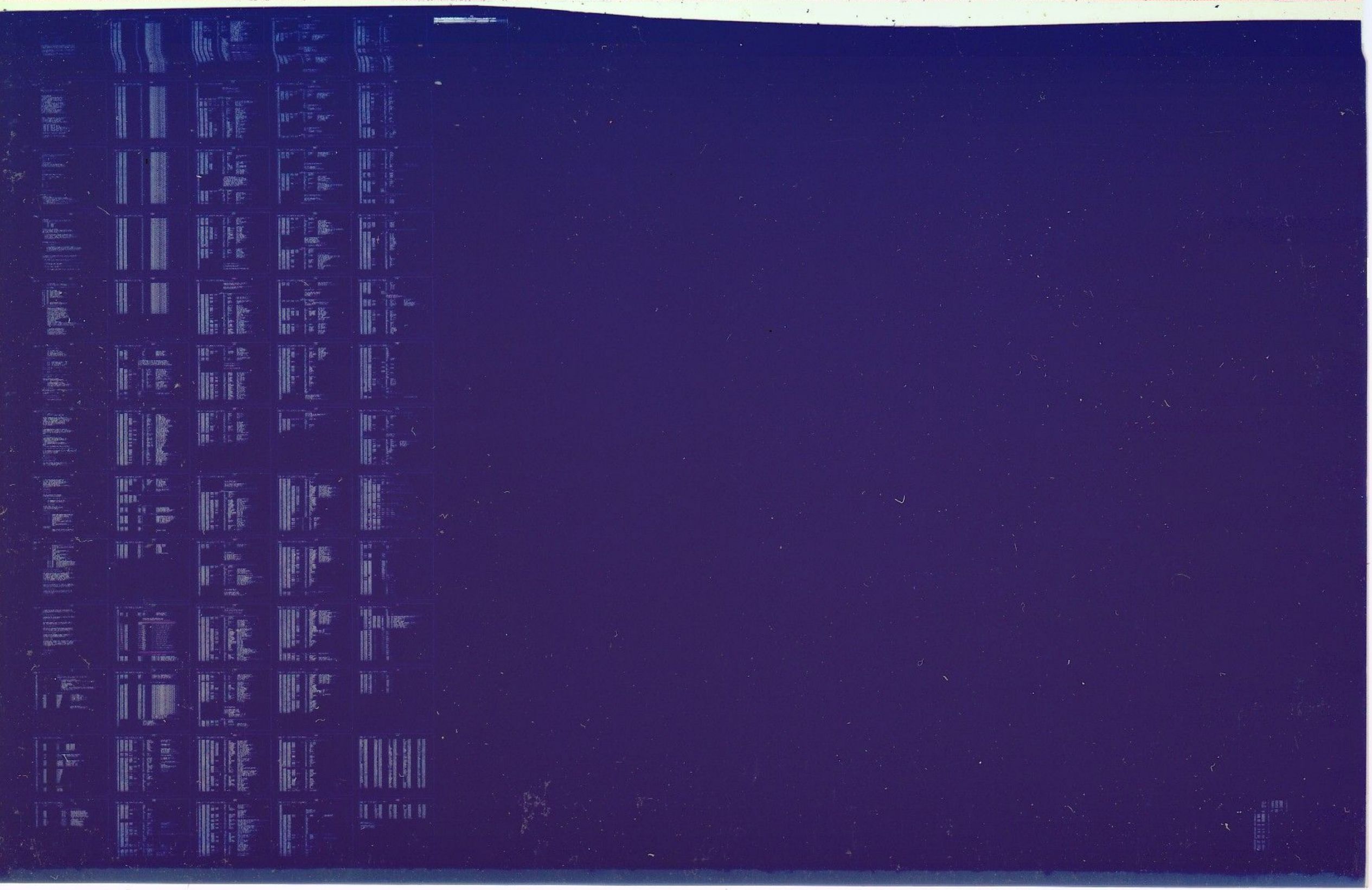
EP-DZDQE-C-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN U.S.A.



IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDQE-C-D

PRODUCT NAME: MISC. RX AND TX TEST PLUS BCC TESTS

DATE: 21 JUNE 1976

MAINTAINER: DIAGNOSTIC GROUP

1. ABSTRACT

THE FUNCTION OF THE DQ11 DIAGNOSTICS ARE TO
VERIFY THAT THE OPTION OPERATES ACCORDING
TO SPECIFICATIONS.

THIS TEST EXERCISES THE BCC OF THE RECEIVER
AND THE TRANSMITTER.
IT USES EVERY POLYNOMIAL BETWEEN 000000-177777
ON AT LEAST ONE CHARACTER AND USES THE
STANDARD POLYNOMIALS ON BLOCK DATA OF AT
LEAST 400 CHARACTERS. THE METHOD USED TO
"TURN ON " THE BCC IS THROUGH THE USE OF
TOTAL TRANSPARENCY.
IN THE TEST WHERE THE DATA IS TRANSFERED IN
BLOCKS THE ACTUALL CORRECTNESS OF THE BCC
IS NOT CALCULATED BY THE PROGRAM. THE PROGRAM
DOES CHECK THE ERROR CONDITION OF THE DQ11
ERROR REGISTER FOR AN ERROR CONDITION.
IF ONE DOES NOT EXIST THE PROGRAM ASSUMES
THE TRANSFER WAS SUCCESFULL.
WHEN THE TRANSFER IS AT ONE CHARACTER
AND EVERY POLYNOMIAL BETWEEN 000000-177777
IS USED THE PROGRAM CALCULATES WHAT
THE BCC SHOULD BE AND COMPARES IT TO THE ACTUAL
RESULTS IN THE TRANSMITTER AND RECEIVER BCCS

CURRENTLY THERE ARE SEVEN OFF LINE DIAGNOSTICS THAT ARE
TO BE RUN IN SEQUENCE TO INSURE THAT IF AN
ERROR SHOULD OCCUR IT WILL BE DETECTED
AT AN EARLY STAGE AND INSURING THAT
DIAGNOSIS OF ERROR WILL BE IMMEDIATE TO PROBLEM
NOTE: ADDITIONAL DIAGNOSTICS MAY BE ADDED IN THE FUTURE.

THE SEVEN DIAGNOSTICS ARE:
1. DZDQA [REV] BASIS R/W TEST #1
2. DZDQB [REV] BASIC R/W TEST #2
3. DZDQC [REV] BASIC NPR AND INTERUPT TEST
4. DZDQD [REV] RECEIVER TRANSMITTER EXERCISER TEST
5. DZDQE [REV] MISC. RX AND TX TESTS. PLUS BCC TESTS.
6. DZDQF [REV] CHARACTER DETECT TESTS.
7. DZDQH [REV] CHARACTER LENGTH AND INTERRUPT TESTS.

THERE IS ALSO AN ONLINE TEST TO BE DISCUSED LATER.
1. DZDQO [REV] ONLINE TEST. (ITEP OVERLAY)

AND A PARAMETER INPUT PROGRAM IS AVAILABLE
1. DZDQG [REV] DQ11 TRIAL PROGRAM (PARAMETER INPUT)

2. REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 4K MEMORY)-WITH
 OR WITHOUT A HARDWARE SWITCH REGISTER (LOC. 177570
ASR 33 (OR EQUILIVALENT)
DQ11
SYNC MODEM (ONLY REQUIRED FOR ONLINE TEST)

2.2 STORAGE

PROGRAM WILL LOAD AND RUN
IN 4K OF MEMORY.
LOCATION 1400 THRU 1600 ARE ESPECIALLY TO
BE NOTED AND TO BE UNTOUCHED BY OPERATOR
AFTER DQ11 TRIAL PROGRAM HAS BEEN EXECUTED.
OR AFTER THE "AUTO SIZING" HAS BEEN DONE.

3. LOADING PROCEEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND
ARE LOADED USING THE ABSOLUTE LOADER.

ABSOLUTE LOADER STARTING ADDRESS *500

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

3.1.1 LOAD THE ADDRESS OF ABS. LOADER (LOC.XXX500)

3.1.2 THEN START

4. STARTING PROCEEDURE

A. LOAD LOC. 200
B. SET SWR TO ZERO FOR "AUTO SIZING" OR LEAVE
   LEAVE SWR BIT 7=1 TO USE EXISTING PARAMETERS SET UP
   BY DQ11 TRIAL PROGRAM OR A PREVIOUSLY RUN DQ11 DIAGNOSTIC
   THAT USED THE "AUTO SIZING".
    ****REFER TO SECTION 4.1 FOR SOFTWARE SWITCH REGISTER OPERATION
        AND OPTIONS.****
   NOTE:THE SOFTWARE SWITCH REGISTER IS LOCATED AT LOC.176
   SOFTWARE DISPLAY REGISTER IS LOCATED AT LOC.174

C.THEN START
THE PROGRAM WILL TYPE MAINDEC NAME AND PROGRAM NAME
IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO
THE FOLLOWING:

```
        "MAP OF DQ11 STATUS"
        1400    160010
        1402    152300
        1404    160020
        1406    150310
```

THE ABOVE IS ONLY AN EXAMPLE!
THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADD.
1400 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.

****IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING
WILL BE TYPED AFTER THE PROGRAM IDENTIFIES ITSELF:
SWR=XXXXXX NEW= (REFER TO SECTION 4.1 FOR OPERATOR'S OPTION)****
NOTE:IF USING THE SOFTWARE SWITCH REGISTER WHEN A HARDWARE
SWITCH REGISTER IS AVAILABLE THE PROGRAM WILL NOT
TYPE OUT THE TITLE.

THE PROGRAM WILL TYPE "R"
AND PROCEED TO RUN THE DIAGNOSTIC

4.1 CONTROL SWITCH SETTINGS

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.

SW 15	SET: HALT ON ERROR
SW 14	SET: LOOP ON CURRENT TEST
SW 13	SET: INHIBIT ERROR PRINT OUT
SW 12	SET: INHIBIT TYPE OUT/BELL ON ERROR.
SW 11	SET: INHIBIT ITERATIONS
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: USE PREVIOUS STATUS TABLE. CLR-DO AUTO SIZE.
SW 06	SET:
SW 05	SET:
SW 04	SET:
SW 03	SET:
SW 02	SET: LOCK ON SELECTED TEST
SW 01	SET: RESTART PROGRAM AT SELECTED TEST
SW 00	SET: RESELECT DQ11'S DESIRED ACTIVE.

4.1.2 SWITCH REGISTER RESTRICTIONS

SW 00 RESELECT DQ11'S DESIRED ACTIVE.
PLEASE NOTE THAT A MESSAGE IS TYPED
OUT FOR SWITCH REGISTER BEING EQUAL TO DQ11'S
ACTIVE. THIS MEANS IF THE SYSTEM HAS
FOUR DQ11S; BITS 00,01,02,03 WILL
BE SET IN LOC "DQACTV". USING THIS
SWITCH ALTERS THAT LOCATION;THEREFORE
IF FOUR DQ11S ARE IN THE SYSTEM
DO NOT SET SWITCHS GREATER THAN
SW 03 IN THE UP POSITION. THIS WOULD BE
A FATAL ERROR. DO NOT SELECT MORE ACTIVE
DQ11S THAN HAS BEEN GIVEN INFORMATION
ABOUT IN TRIAL PROGRAM.

METHOD:
A: LOAD ADDRESS 200
B: START WITH SW 00=1
C: PROGRAM WILL TYPE MESSAGE
D: CONTINUE THE BINARY NUMBER OF DQ11S DESIRED ACTIVE
EXAMPLE: 1=1 DQ11; 3=2 DQ11; 7=3 DQ11; 17=4 DQ11 37=5 DQ11 ETC.
E: NUMBER (IF VALID) WILL BE IN DATA LIGHTS (EXCLUDING 11/05, 11/04, 11/34)
F: CONTINUE WITH ANY OTHER SWITCH SETTINGS DESIRED.

SW 01 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. ALSO WHEN A TEST IS
SELECTED ALWAYS START AT THE VERY

BEGINNING OF THAT TEST.

SW 09 LOOP ON CURRENT DATA:
THIS SWITCH WILL ONLY WORK IF
CALL "SCOP1" 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 CANN'T BE SINGLED OUT.

4.1.3 SWITCH REGISTER PRIORITYS

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 GOTO BEGINNING OF THE TEST.
5. SW 10 GOTO NEXT TEST ON ERROR.

****HLT (ERROR) ROUTINE SUPPORTS (↑G) OPERATION****

SCOPE SWITCHES

1. SW 09 (IF ENABLED BY "SCOP1")
2. SW 14
3. SW 11

****SCOPE ROUTINE WILL SUPPORT (↑G) OPERATION****

4.2 STARTING ADDRESS

STARTING ADDRESS IS AT 000200
THERE ARE NO OTHER STARTING ADDRESSES
FOR THE DQ11 DIAGNOSTICS PREVIOUSLY MENTIONED

NOTE: IF ADDRESS 000042 IS NON-ZERO
THE PROGRAM ASSUMES IT IS UNDER
ACT11 OR DDP CONTROL AND WILL ACT ACCORDINGLY
AFTER *ALL* AVAILABLE DQ11'S ARE TESTED
THE PROGRAM WILL RETURN TO "DDP2" OR "ACT-11".

5. OPERATING PROCEDURE

WHEN PROGRAM IS INITIALLY STARTED MESSAGES AS DESCRIBED IN SECTION
FOUR WILL BE PRINTED.

AND PROGRAM WILL BEGIN RUNNING THE
DIAGNOSTIC

5.2 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

1. HALT ON ERROR (VIA SW 15=1)
WHEN EVER AN ERROR OCCURS

2. CLEAR SW 15
3. SET SW 14: (LOOP ON THIS TEST)
4. SET SW 13: (INHIBIT ERROR PRINT OUT)

THE TEST NUMBER AND PC WILL BE TYPED OUT AND
POSSIBILY AN ERROR MESSAGE (THIS DEPENDS ON THE TEST)
TO GIVE THE OPERATOR AN IDEA AS TO THE SOURCE OF THE
PROBLEM. IF IT IS NECESSARY TO KNOW MORE INFORMATION
CONCERNING THE ERROR REPORT; LOOK IN THE LISTING
FOR THAT TEST NUMBER WHICH WAS TYPED OUT
AND THEN NOTE THE PC OF THE ERROR REPORT
THIS WAY THE EXACT FUNCTIONING OF THE TEST
CAN BE INTERPEDITED

6. ERRORS

AS DESCRIBED PREVIOUSLY THERE WILL ALWAY 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 THE THE ERROR MESSAGE
WHICH IS TO GIVE THE OPERATOR AN INDICATION OF THE
ERROR.

6.2 ERROR RECOVERY

IF FOR SOME REASON THE DQ11 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
"TSTNO" (ADDRESS 1222)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 DQ11 WAS DOING AT THE TIME OF THE ERROR.

6.3 ****HALT RECOVERY WHEN USING SOFTWARE SWITCH REGISTER****

IF THE SOFTWARE SWITCH REGISTER IS TO BE CHANGED AFTER A HALT
THE THE OPERATOR IS REQUIRED TO TYPE A <↑G> BEFORE DEPRESSING CONTINUE.
THE FOLLOWING WILL BE TYPED:
SWR=XXXXXX NEW= (REFER TO SECTION 4.1 FOR OPERATOR OPTION)

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

SEE SECTION 4. (PLEASE)

7.2 OPERATING RESTRICTIONS

DQ11 TRIAL PROGRAM MUST BE RUN PRIOR TO THE
FIRST AND ONLY THE FIRST RUNNING OF ANY DQ11 DIAGNOSTIC
NOTE: IF NO PROGRAM OTHER THAN A
DQ11 DIAGNOSTIC WAS LOADED AFTER DQ11 TRIAL OR

IF CORE MEMORY HAS NOT BEEN CHANGED;OR IF THERE
IS NO DQ11 CONFIGURATION CHANGES; THE
DQ11 TRIAL PROGRAM NEED NEVER BE RUN AGAIN.
HOWEVER IF ANY OF THE ABOVE HAVE BEEN VIOLATED
THE DQ11 TRIAL PROGRAM MUST BE RUN AGAIN
BEFORE RUNNING THE DIAGNOSTICS
NOTE: AN ALTERNATIVE TO THE ABOVE IS ATTEMPTING
THE "AUTO SIZING" WHEN PROGRAM IS INITIALLY STARTED
WITH SW07=0.

8. MISCELLANEOUS

8.1 EXECUTION TIME

8.2 PASS COMPLETE

WHEN THE DIAGNOSTIC HAS COMPLETED
A PASS THE FOLLOWING IS AN EXAMPLE
OF THE PRINT OUT TO BE EXPECTED.

END PASS DZDQE-C CSR: 160000 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.

8.3 TST1 (MINI MONITOR)

THE VERY FIRST "TEST" (TST1)
IS *NOT* A TEST OF THE DQ11 HARDWARE
IT IS A MINI-MONITOR USED TO CYCLE DQ11 IN THE
SYSTEM THROUGH THE DIAGNOSTIC.

REMEMBER: TST1 IS NOT A TEST OF DQ11 HARDWARE!!!!!!

8.4 KEY LOCATIONS

RETURN (1210) CONTAINS THE ADDRESS WHERE PROGRAM WILL
RETURN WHEN ITERATION COUNT IS REACHED
OR IF LOOP ON TEST IS ASSERTED.

NEXT (1212) CONTAINS THE ADDRESS OF THE NEXT TEST
TO BE PEFORMED.

TSTNO (1222) CONTAINS THE NUMBER OF THE TEST NOW
BEING PEFORMED.

RUN (1272 THE BIT IN "RUN" ALWAYS POINTS ONE
PAST THE DQ11 CURRENTLY BEING TESTED.
EXAMPLE:
(RUN) 1272/0000000001000000
MEANS THAT DQ11 NO.05 IS THE DQ11 NOW
RUNNING.

DQCR00-DQCR17
DQST00-DQST17
(1400)-(1476)

THESE LOCATIONS CONTAIN THE INFORMATION
NEEDED TO TEST UP TO 16 (DECIMAL) DQ11S
SEQUENTIALY. THEY CONTAIN THE CSR,VECTOR

AND STATUS CONCERNING THE CONFIGURATION
OF EACH DQ11.

DQACTV (1500) EACH BIT SET IN THIS LOCATION INDICATES
THAT THE ASSOCIATED DQ11 WILL BE TESTED
IN TURN.
EXAMPLE:
(DQACTV) 1500/0000000000011111
MEANS THAT DQ11 NO. 00,01,02,03,04
WILL BE TESTED.
EXAMPLE:
(DQACTV) 1500/0000000000010001
MEANS THAT DQ11 NO. 00,04
WILL BE TESTED.

DQCSR (1506) CONTAINS THE RECEIVER CSR OF THE
CURRENT DQ11 UNDER TEST.

DQSTAT (1510) CONTAINS THE STATUS OF THE CURRENT
DQ11 UNDER TEST.

BIT 15 SET: TWO SYNC CHARS/ONE SYNC CHAR
BIT 14 SET: TEST JUMPER INSTALLED/NOT INSTALLED
BIT 13 SET: B8 OPTION INSTALLED/NOT INSTALLED
BIT 12 SET: BA OPTION INSTALLED/NOT INSTALLED
BIT 11 SET: ACTIVE ON FIRST NON-SYNC/ACTIVE AFTER NO. OF SYNC
BIT 10 SET: AB OPTION INSTALLED/NOT INSTALLED
BIT 09 SET: ODD VRC/EVEN VRC
BIT 00-08 VECTOR "A" OF DEVICE

8.5 *** METHOD OF AUTO SIZING ***

8.5.1 FINDING THE CONTROL STATUS REGISTER.

WHEN LOOKING FOR THE CSR IT IS NECESSARY TO TAKE CARE
THAT WHEN A CSR IS FOUND THAT IT IS INDEED A DQ11. THAT
IS THE METHOD OF MY MADNESS FOR THIS ROUTINE.
AN ATTEMPT TO CLEAR THE MISC. REGISTER IS TRIED
IF A TIME-OUT TRAP OCCURES POINTERS ARE UPDATED
AND ATTEMPTED AGAIN. IF NO TIME-OUT; THE RECEIVER "ACTIVE BIT" (BIT 12)
IS SET AND A *COMPARE* FOR BOTH SYNC1 AND SYNC 2 IS DONE
AT THE MISC. REGISTER. IF THEY ARE THERE THIS IS
A DQ11. THE INFORMATION IS STORED AWAY.

8.5.2 ONE SYNC BIT OR TWO?

SINCE TOO MUCH HARDWARE MUST BE TURNED ON TO SENSE THE
PRESENTS OF ONE SYNC OR TWO. THE PROGRAM ASSUMES TWO SYNC
CHARS. NOTE: THIS ASSUMPTION MAY BE ALTERED AFTER AUTO SIZING
BY ALTERING BIT 15 IN APPRICATE DQSTXX: LOCATION.

8.5.3 "BB" OPTION INSTALLED?

TO SENSE FOR THE "BB" OPTION THE PROGRAM SELECTS THE
CHARACTER DET. REGISTER AND THE LOADS IN ALL 1'S; IF
ANY ONE OR COMBINATION OF BITS ARE SET THE BB OPTION
IS ASSUMED TO EXIST.

8.5.4 "AB" OPTION INSTALLED?

TO SENSE FOR THE "AB" OPTION THE PROGRAM SELECTS THE
POLYNOMIAL REGISTER AND WRITES ALL 1'S INTO IT; IF ANY
ONE OR COMBINATION OF BITS ARE SET THE AB OPTION IS ASSUMED
TO EXIST.

8.5.5 "BA" OPTION INSTALLED?

TO SENSE FOR "BA" OPTION REQUEST TO SEND AND DATA TERMINAL
READY ARE SET; IF EITHER ONE OR BOTH ARE SET THE PROGRAM
ASSUMES THE BA OPTION EXISTES

8.5.6 JUMPER ON END OF CABLE?

THE PROGRAM CHECKS TO SEE IF EITHER OR BOTH CLEAR TO
SEND AND CARRIER ARE SET; IF SO THE PROGRAM ASSUMES THE
TEST JUMPER IS ON THE END OF THE CABLE.

8.5.7 ACTIVE ON FIRST NON-SYNC?

SINCE TOO MUCH HARDWARE MUST BE TURNED ON TO SENSE FOR
WHEN THE DQ11 GOES ACTIVE THE PROGRAM ASSUMES "ACTIVE
ON FIRST NON-SYNC". NOTE: THIS CAN BE CHANGED BY ALTERING
BIT 11 IN THE APPRIOATE DQSTXX: AFTER AUTO SIZING

8.5.8 SET FOR ODD OR EVEN PARITY?

AS ABOVE TOO MUCH HARDWARE IS NEED TO SENSE WHICH PARITY
WAS SELECTED.SO THE PROGRAM ASSEMES ODD PARITY.
NOTE: THIS CAN BE CHANGED BY ALTERING BIT 9 IN APPRIO-
ATE DQSTXX: LOCATION. AFTER AUTO SIZING

8.5.9 FINDING THE VECTOR.

THE PROGRAM SETS "PRIMARY DONE","SECONDAY DONE", AND "INTERUPT ENABLE"
AND LOOKS FOR AN INTERUPT. IF IT INTERUPTS IT IS PICKED
UP AND STORED AWAY. IF NO INTERUPT OCCURES THE PROGRAM
ASSUMES VECTOR =300. THIS PROBLEM WILL BE FIXED IN ONE
OF THE DIAGNOSTICS AND *AUTO SIZING* SHOULD BE REDONE TO
GET THE CORRECT VECTOR.

9. PROGRAM DESCRIPTION

CONTAINED WITHIN LISTING

10. LISTING

FOLLOWING

```
                                .ENABLE AMA

                                :MAINDEC-11-DZDQE-C/(377)/TX AND RX MISC. AND BCC TESTS
                                :COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

                                :REVISED 21-JUNE-76 BY S. CARPENTER
                                :       A)SUPPORTS SOFTWARE SWITCH REGISTER
                                :       B)SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER
                                :         BY (↑G).
                                        ;STARTING PROCEDURE
                                        :LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        :PRESS START
                                        ;PROGRAM WILL TYPE "MAINDEC-11-DZDQE-C/(377)/TX AND RX MISC. AND BCC TESTS"
                                        :PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        :AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
                                        ;AND THEN RESUME TESTING


                                :SWITCH REGISTER OPTIONS

          100000                SW15=100000             ;=1,HALT ON ERROR
          040000                SW14=40000              ;=1,LOOP ON CURRENT TEST
          020000                SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
          010000                SW12=10000              ;=1,DELETE TYPEOUT/BELL ON ERROR.
          004000                SW11=4000               ;=1,INHIBIT ITERATIONS
          002000                SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
          001000                SW09=1000               ;=1,LOOP WITH CURRENT DATA
          000400                SW08=400                ;=1,LOOP ON ERROR
          000100                SW06=100
          000040                SW05=40
          000020                SW04=20
          000010                SW03=10
          000004                SW02=4                  ;LOCK ON TEST SELECT
          000002                SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
          000001                SW00=1                  ;RESELECT DQ11 DESIRED ACTIVE
                                                        ;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT
```

```
                                ;REGISTER DEFINITIONS

000000                          R0=%0           ;GENERAL REGISTER
000001                          R1=%1           ;GENERAL REGISTER
000002                          R2=%2           ;GENERAL REGISTER
000003                          R3=%3           ;GENERAL REGISTER
000004                          R4=%4           ;GENERAL REGISTER
000005                          R5=%5           ;GENERAL REGISTER
000006                          SP=%6           ;PROCESSOR STACK POINTER
000007                          PC=%7           ;PROGRAM COUNTER

                                ;LOCATION EQUIVALENCIES

177570                          DSWR=   177570  ;HARDWARE SWITCH REGISTER LOC.
177570                          DLIGHTS=177570  ;HARDWARE DISPLAY REGISTER LOC.
177776                          PS=177776       ;PROCESSOR STATUS WORD
001200                          STACK=1200      ;START OF PROCESSOR STACK

                                ;INSTRUCTION DEFINITIONS

005746                          PUSH1SP=5746    ;DECREMENT PROCESSOR STACK 1 WORD
005726                          POP1SP=5726     ;INCREMENT PROCESSOR STACK 1 WORD
010046                          PUSHR0=10046    ;SAVE R0 ON STACK
012600                          POPR0=12600     ;RESTORE R0 FROM STACK
024646                          PUSH2SP=24646   ;DECREMENT STACK TWICE
022626                          POP2SP=22626    ;INCREMENT STACK TWICE
                                .EQUIV  EMT,HLT ;BASIC DEFINITION OF ERROR CALL


100000                          BIT15=100000
040000                          BIT14=40000
020000                          BIT13=20000
010000                          BIT12=10000
004000                          BIT11=4000
002000                          BIT10=2000
001000                          BIT9=1000
000400                          BIT8=400
000200                          BIT7=200
000100                          BIT6=100
000040                          BIT5=40
000020                          BIT4=20
000010                          BIT3=10
000004                          BIT2=4
000002                          BIT1=2
000001                          BIT0=1


                                ;DQ11 OPTIONAL DEFINITIONS

002000                          ABBIT=2000
004000                          ACTBIT=4000
010000                          BABIT=10000
020000                          BBBIT=20000
040000                          JUMBIT=40000
```

```
DZDQE   MACY11 27(732)  24-MAY-76  13:14  PAGE 14
DZDQEC.P11      GENERAL DEFINATIONS AND EQUIVALENCIES

                001000                  ODDBIT=1000
                100000                  SYNBIT=100000


                                        ;DQ11 SECONDARY REGISTER DEFINATIONS

                000000                  RXBA.P=0        ;RECEIVER BUS ADDRESS PRIMARY.
                000001                  RXWC.P=1        ;RECEIVER WORD COUNT PRIMARY.
                000002                  TXBA.P=2        ;TRANSMITTER BUS ADDRESS PRIMARY.
                000003                  TXWC.P=3        ;TRANSMITTER BUS ADDRESS PRIMARY.
                000004                  RXBA.S=4        ;RECEIVER BUS ADDRESS SECONDARY.
                000005                  RXWC.S=5        ;RECEIVER WORD COUNT SECONDARY.
                000006                  TXBA.S=6        ;TRANSMITTER BUS ADDRESS SECONDARY.
                000007                  TXWC.S=7        ;TRANSMITTER WORD COUNT SECONDARY.

                000010                  CHARDT=10       ;CHARACTER DETECT REGISTER.
                000011                  SYNC.=11        ;SYNC REGISTER.
                000012                  MISC.=12        ;MISCELLANEOUS REGISTER.
                000013                  TX.MUX=13       ;TRANSMITTER MUX REGISTER.
                000014                  SEQ.=14         ;SEQUENCE REGISTER.
                000015                  RX.BCC=15       ;RECEIVER BCC REGISTER.
                000016                  TX.BCC=16       ;TRANSMITTER BCC REGISTER.
                000017                  POLY.=17        ;POLYNOMIAL REGISTER.
```

```
                                        ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
         000000                 .=0
000000   000002                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000002   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000004   000006                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000006   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000010   000012                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000012   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000014   000016                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000016   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000020   000022                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000022   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000024   000026                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000026   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000030   000032                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000032   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000034   000036                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000036   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000040   000042                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000042   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000044   000046                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000046   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000050   000052                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000052   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000054   000056                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000056   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000060   000062                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000062   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000064   000066                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000066   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000070   000072                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000072   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000074   000076                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000076   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000100   000102                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000102   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000104   000106                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000106   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000110   000112                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000112   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000114   000116                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000116   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000120   000122                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000122   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000124   000126                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000126   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000130   000132                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000132   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000134   000136                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000136   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000140   000142                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000142   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000144   000146                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000146   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
000150   000152                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000152   000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
000154  000156          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000156  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000160  000162          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000162  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000164  000166          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000166  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000170  000172          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000172  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000174  000176          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000176  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000200  000202          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000202  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000204  000206          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000206  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000210  000212          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000212  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000214  000216          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000216  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000220  000222          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000222  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000224  000226          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000226  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000230  000232          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000232  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000234  000236          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000236  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000240  000242          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000242  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000244  000246          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000246  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000250  000252          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000252  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000254  000256          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000256  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000260  000262          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000262  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000264  000266          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000266  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000270  000272          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000272  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000274  000276          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000276  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000300  000302          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000302  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000304  000306          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000306  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000310  000312          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000312  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000314  000316          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000316  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000320  000322          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000322  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000324  000326          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000326  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
000330  000332          .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000332  000000          HALT         ;EXAMINE STACK TO FIND CAUSE
```

```
                000334  000336                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000336  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000340  000342                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000342  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000344  000346                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000346  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000350  000352                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000352  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000354  000356                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000356  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000360  000362                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000362  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000364  000366                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000366  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000370  000372                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000372  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000374  000376                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000376  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000400  000402                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000402  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000404  000406                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000406  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000410  000412                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000412  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000414  000416                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000416  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000420  000422                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000422  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000424  000426                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000426  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000430  000432                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000432  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000434  000436                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000436  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000440  000442                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000442  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000444  000446                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000446  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000450  000452                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000452  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000454  000456                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000456  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000460  000462                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000462  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000464  000466                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000466  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000470  000472                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000472  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000474  000476                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000476  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000500  000502                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000502  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000504  000506                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000506  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
                000510  000512                  .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                000512  000000                  HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
    830  000514  000516                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    831  000516  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    832  000520  000522                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    833  000522  000000                    HALT         EXAMINE STACK TO FIND CAUSE
    834  000524  000526                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    835  000526  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    836  000530  000532                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    837  000532  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    838  000534  000536                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    839  000536  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    840  000540  000542                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    841  000542  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    842  000544  000546                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    843  000546  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    844  000550  000552                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    845  000552  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    846  000554  000556                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    847  000556  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    848  000560  000562                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    849  000562  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    850  000564  000566                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    851  000566  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    852  000570  000572                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    853  000572  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    854  000574  000576                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    855  000576  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    856  000600  000602                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    857  000602  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    858  000604  000606                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    859  000606  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    860  000610  000612                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    861  000612  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    862  000614  000616                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    863  000616  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    864  000620  000622                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    865  000622  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    866  000624  000626                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    867  000626  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    868  000630  000632                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    869  000632  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    870  000634  000636                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    871  000636  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    872  000640  000642                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    873  000642  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    874  000644  000646                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    875  000646  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    876  000650  000652                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    877  000652  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    878  000654  000656                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    879  000656  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    880  000660  000662                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    881  000662  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    882  000664  000666                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    883  000666  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
    884  000670  000672                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
    885  000672  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
```

```
  896  000674  000676                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  897  000676  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  898  000700  000702                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  899  000702  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  890  000704  000706                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  891  000706  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  892  000710  000712                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  893  000712  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  894  000714  000716                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  895  000716  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  896  000720  000722                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  897  000722  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  898  000724  000726                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  899  000726  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  900  000730  000732                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  901  000732  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  902  000734  000736                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  903  000736  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  904  000740  000742                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  905  000742  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  906  000744  000746                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  907  000746  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  908  000750  000752                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  909  000752  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  910  000754  000756                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  911  000756  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  912  000760  000762                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  913  000762  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  914  000764  000766                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  915  000766  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  916  000770  000772                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  917  000772  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  918  000774  000776                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  919  000776  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
```

```
                                                ;STANDARD INTERRUPT VECTORS

                000024                  .=24
000024  014256                                  .PFAIL                  ;POWER FAIL HANDLER
000026  000340                                  340                     ;SERVICE AT LEVEL 7
000030  013726                                  .HLT                    ;ERROR HANDLER
000032  000340                                  340                     ;SERVICE AT LEVEL 7
000034  013674                                  .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                                  340                     ;SERVICE AT LEVEL 7
                000046                  .=46
000046  012454                                  LOGICAL                 ;ACT HOOKS
                000052                  .=52
000052  000000                                  .WORD 0
                                        ;THIS ROUTINE TRIES TO FORCE THE RECEIVER TO INTERUPT
                                        ;TO ITS VECTOR WHERE IT WILL PICK UP THE STATUS LOCATION
                                        ;FOR ITS NEW PC; AND PICK UP AN IOT INSTRUCTION FOR ITS
                                        ;NEW PS. WHEN THE NEW PC IS FETCHED AN IOT INSTRUCTION IS
                                        ;EXECUTED, TRAPPING TO LOCATION 20 WHERE A ROUTINE IS EXECUTED
                                        ;TO TAKE THE PC FROM THE STACK AND US IT AS THE VECTOR ADDRESS
                000056                  .=56

000056                                  VECMAP:
000056  010120                  1$:     MOV     R1,(R0)+        ;START FILLING THE VECTOR AREA
000060  012721  000004                  MOV     #4,(R1)+        ;WITH .+2; IOT (4)
000064  022021                          CMP     (R0)+,(R1)+     ;UPDATE THE POINTERS
000066  020127  001000                  CMP     R1,#1000        ;IS ALL FLOATING VECTOR AREA DONE
000072  101771                          BLOS    1$              ;BR IF NOT ALL DONE
000074  012737  000146  000020          MOV     #4$,@#20        ;SET FOR IOT TRAP BY DQ11
000102  013737  001500  001244          MOV     DQACTV,TEMP1    ;GET THE ACTIVE DQ11 S
000110  006037  001244          2$:     ROR     TEMP1           ;ARE YOU ACTIVE.. DQ11
000114  103023                          BCC     5$              ;IF CARRY CLEAR.. NO MORE DQ11S
000116  005037  177776                  CLR     PS              ;CLEAR PS
000122  005722                          TST     (R2)+           ;PUT POINTER TO STATUS TABLE
000124  012772  000340  177776          MOV     #340,@-2(R2)    ;TRY AND SET PRI/SEC DONE AND IE
000132  105200                          INCB    R0              ;DELAY.....
000134  001376                          BNE     .-2             ;.....DELAY
000136  112712  000300                  MOVB    #300,(R2)       ;NO INTERUPT ASSUME 300 FIX IN TEST C
000142  005722                  3$:     TST     (R2)+           ;UPDATE POINTERS
000144  000761                          BR      2$              ;GO DO IT AGAIN
000146  051612                  4$:     BIS     (SP),(R2)       ;ENTERD  BY IOT TRAP BY DQ11
000150  042712  000007                  BIC     #7,(R2)         ;CLEAR UNWANTED BITS
000154  022626                          CMP     (SP)+,(SP)+     ;POP IOT JUNK OFF STACK
000156  012716  000142                  MOV     #3$,(SP)        ;SET RETURN PC ON STACK
000162  000002                          RTI                     ;GO HOME.
000164  000207                  5$:     RTS     PC              ;ALL SIZING IS DONE

                                        ;****SOFTWARE SWITCH REGISTER****
                000174                  .=174
000174  000000                  DISPREG:        0               ;SOFTWARE DISPLAY REGISTER
000176  000000                  SWREG:          0               ;SOFTWARE SWITCH REGISTER

                                        ;PROGRAM START

                000200                  .=200
000200  000137  001512                  JMP     .START          ;GO TO START OF PROGRAM
```

```
 976            000220                          .=220
 977  000220   012702  001400          CSRMAP: MOV     #1400,R2        ;CLEAR ALL STATUS TABLE
 978  000224   005022                          CLR     (R2)+           ;DO CLEAR
 979  000226   022702  001512                  CMP     #1512,R2        ;ALL TABLE DONE
 980  000232   001374                          BNE     .-6             ;BR IF MORE TO GO
 981  000234   005037  001504                  CLR     DQNUM           ;SET NUMBER OF DQ11S TO 0
 982  000240   012702  001400                  MOV     #1400,R2        ;SET TABLE POINTER
 983  000244   012701  160000                  MOV     #160000,R1      ;GET FIRST FLOATING ADDRESS
 984  000250   012737  000614  000004          MOV     #5$,@#4         ;SET FOR TIME OUT TRAP--NO DEVICE--
 985  000256   112761  000012  000005  1$:     MOVB    #12,5(R1)       ;TRY AND SEL MISC REGISTER
 986  000264   005061  000006                  CLR     6(R1)           ;TRY AND CLEAR MISC REG
 987  000270   012711  010000                  MOV     #10000,(R1)     ;TRY AND SET RX ACTIVE
 988  000274   022761  030000  000006          CMP     #30000,6(R1)    ;LOOK FOR SYNC 1 AND SYNC 2
 989  000302   001071                          BNE     2$              ;THIS IS NOT A DQ11 IF I BRANCH
 990  000304   010122                          MOV     R1,(R2)+        ;NOW THIS IS A DQ11 --STORE CSR
 991  000306   052712  100000                  BIS     #SYNBIT,(R2)    ;SET FOR TWO SYNC CHARS
 992  000312   005011                          CLR     (R1)            ;CLEAR DQ ACTIVE BIT
 993  000314   112761  000010  000005          MOVB    #10,5(R1)       ;SEL CHAR DET REGISTER
 994  000322   012761  177777  000006          MOV     #-1,6(R1)       ;WRITE INTO CHAR DET REG
 995  000330   005761  000006                  TST     6(R1)           ;WAS THE REGISTER WRITTEN?
 996  000334   001402                          BEQ     .+6             ;APPERENTLY NO BB OPTION.
 997  000336   052712  020000                  BIS     #BBBIT,(R2)     ;SET FOR BB OPTION
 998  000342   112761  000017  000005          MOVB    #17,5(R1)       ;SEL POLYNO. REGISTER
 999  000350   012761  177777  000006          MOV     #-1,6(R1)       ;WRITE POLYNO.REGISTER
1000  000356   005761  000006                  TST     6(R1)           ;WAS REG WRITTEN??
1001  000362   001402                          BEQ     .+6             ;BR IF NO AB OPTION
1002  000364   052712  002000                  BIS     #ABBIT,(R2)     ;SET FOR AB OPTION
1003  000370   012761  001400  000002          MOV     #1400,2(R1)     ;TRY TO SET .DTR. .RS.
1004  000376   032761  001400  000002          BIT     #1400,2(R1)     ;DID ANY OF THEM SET
1005  000404   001402                          BEQ     .+6             ;BR IF NO BA OPTION
1006  000406   052712  010000                  BIS     #BABIT,(R2)     ;SET FOR BA OPTION
1007  000412   032761  030000  000002          BIT     #30000,2(R1)    ;DID .CS. .CO. SET
1008  000420   001402                          BEQ     .+6             ;BR IF NO JUMPER
1009  000422   052712  040000                  BIS     #JUMBIT,(R2)    ;SET FOR JUMPER
1010  000426   052712  004000                  BIS     #ACTBIT,(R2)    ;SET FOR ACTIVE ON FIRST NON-SYNC
1011  000432   052712  001000                  BIS     #ODDBIT,(R2)    ;SET FOR ODD VRC.......
1012  000436   005722                          TST     (R2)+           ;POP POINTER
1013  000440   005011                          CLR     (R1)            ;CLEAR RCSR
1014  000442   005061  000002                  CLR     2(R1)           ;CLEAR TCSR
1015  000446   005061  000002                  CLR     2(R1)           ;CLEAR AGAIN
1016  000452   005061  000004                  CLR     4(R1)           ;CLEAR ERROR REG
1017  000456   005061  000006                  CLR     6(R1)           ;CLEAR SEC REG
1018  000462   005237  001504                  INC     DQNUM           ;UPDATE NUMBER OF DQ11S
1019  000466   062701  000010          2$:     ADD     #10,R1          ;UPDATE CSR POINTER BY 10 (8)
1020  000472   022701  164000                  CMP     #164000,R1      ;HAVE ALL FLOATING ADDRESSES BEEN CHECKED??
1021  000476   001267                          BNE     1$              ;BR IF NOT ALL DONE
1022  000500   005037  001500                  CLR     DQACTV          ;ZERO ACTIVE DQ11S
1023  000504   005737  001504                  TST     DQNUM           ;WERE ANY DQ11S FOUND
1024  000510   001434                          BEQ     4$              ;HEY BUDDY. NO DQ11S FOUND IN SYSTEM
1025  000512   013701  001504                  MOV     DQNUM,R1        ;SAVE NUMBER OF DQ11S
1026  000516   010137  001276                  MOV     R1,SAVNUM       ;SAVE NUMBER FOR ACT11
1027  000522   000241                  3$:     CLC                     ;CLEAR CARRY
1028  000524   006137  001500                  ROL     DQACTV          ;<<<<< ACTIVE ADDRESS
1029  000530   005237  001500                  INC     DQACTV          ;SET BIT 0
1030  000534   005301                          DEC     R1              ;DEC NUMBER OF DQ11S
1031  000536   001371                          BNE     3$              ;BR IF MORE TO GO
```

```
1032  000540  012737  000006  000004          MOV     #6,@#4          ;RESET TIME OUT VECTOR
1033  000546  013737  001500  001502          MOV     DQACTV,SAVACT   ;SAVE ACTIVE
1034  000554  012737  000340  000022          MOV     #340,@#22       ;SET IOT TRAP PRIO: TO 7
1035  000562  012702  001400                  MOV     #1400,R2        ;SET TABLE POINTER
1036  000566  012700  000300                  MOV     #300,R0         ;SET VECTOR START
1037  000572  012701  000302                  MOV     #302,R1         ;SET VECTOR+2 START
1038  000576  000137  000056                  JMP     VECMAP          ;GO FIND THE VECTORS
1039  000602  104402                  4$:     TYPE                    ;TYPE MESSAGE
1040  000604  014617                          MERR2                   ;I DIDN'T FIND ANY DQ11S. DON'T USE AUTO SIZE.
1041  000606  005000                          CLR     R0              ;
1042  000610  000000                          HALT                    ;HOW CAN I TEST NO DQ11S
1043  000612  000776                          BR      .-2             ;DON'T LET OPR HIT CONT. SW
1044  000614  012716  000466          5$:     MOV     #2$,(SP)        ;ENTERED BY TIME OUT TRAP
1045  000620  000002                          RTI                     ;GO HOME.
1046
1047
1048          001000                  .=1000
1049  001000  005377  040515  047111  MTITLE: .ASCIZ  <377><12>/MAINDEC-11-DZDQE-C/<377>/TX AND RX MISC. AND BCC TESTS/<377>
1050  001006  042504  026503  030461
1051  001014  042055  042132  042521
1052  001022  041455  052377  020130
1053  001030  047101  020104  054122
1054  001036  046440  051511  027103
1055  001044  040440  042116  041040
1056  001052  041503  052040  051505
1057  001060  051524  000377
1058
1059          001200                  .=1200
1060                                  ;INDIRECT POINTERS
1061
1062  001200  177570                  SWR:    177570                  ;SWITCH REGISTER POINTER
1063  001202  177570                  LIGHTS: 177570                  ;DISPLAY REGISTER POINTER
1064  001204  177560                  TKCSR:  177560                  ;TELETYPE KEYBOARD CONTROL REGISTER
1065  001206  177562                  TKDBR:  177562                  ;TELETYPE KEYBOARD DATA BUFFER
1066  001210  177564                  TPCSR:  177564                  ;TELEPRINTER CONTROL REGISTER
1067  001212  177566                  TPDBR:  177566                  ;TELEPRINTER DATA BUFFER
1068
1069                                  ;PROGRAM CONTROL PARAMETERS
1070
1071  001214  000000                  RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
1072  001216  000000                  NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
1073  001220  000000                  LOCK:   0                       ;ADDRESS FOR LOCK ON CURRENT DATA
1074  001222  000003                  ICOUNT: 3                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
1075  001224  000000                  LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
1076  001226  000000                  TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
1077  001230  000000                  PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
1078  001232  000000                  ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
1079  001234  000000                  LSTERR: 0                       ;PC OF LAST ERROR CALL
1080
1081                                  ;PROGRAM VARIABLES
1082
1083  001236  000000                  CHAR1:  0
1084  001240  000000                  CHAR2:  0
1085  001242  000000                  CHAR3:  0
1086  001244  000000                  TEMP1:  0                       ;TEMPORARY STORAGE
1087  001246  000000                  TEMP2:  0                       ;TEMPORARY STORAGE
```

```
  1088  001250  000000                TEMP3:  0                       ;TEMPORARY STORAGE
  1089  001252  000000                TEMP4:  0                       ;TEMPORARY STORAGE
  1090  001254  000000                TEMP5:  0                       ;TEMPORARY STORAGE
  1091  001256  000000                SAVR0:  0                       ;R0 STORAGE
  1092  001260  000000                SAVR1:  0                       ;R1 STORAGE
  1093  001262  000000                SAVR2:  0                       ;R2 STORAGE
  1094  001264  000000                SAVR3:  0                       ;R3 STORAGE
  1095  001266  000000                SAVR4:  0                       ;R4 STORAGE
  1096  001270  000000                SAVR5:  0                       ;R5 STORAGE
  1097  001272  000000                SAVSP:  0                       ;STACK POINTER STORAGE
  1098  001274  000000                SAVPC:  0                       ;PROGRAM COUNTER STORAGE
  1099  001276  000000                SAVNUM: 0
  1100  001300  000001                CREAM:  .BLKW   1
  1101  001302  000000                RUNFLG: 0
  1102  001304  000000                RUN:    0
  1103  001306  000000                RUNCNT: 0
```

```
                                        ;PROGRAM CONTROL FLAGS

001310  000             INIFLG: .BYTE   0               ;PROGRAM INITIALIZATION FLAG
001311  000             STFLG:  .BYTE   0               ;TEST START FLAG
001312  000             ERRFLG: .BYTE   0               ;ERROR OCCURED FLAG
001313  000             LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
        000000          $Y=0

                        ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
                        ;POINTERS TO SUBROUTINES CAN BE FOUND
                        ;IN THE TABLE IMMEDIATLY FOLLOWING THE DEFINITIONS

                        ;*****************************************************************
                        ;*****************************************************************
001314                  .TRPTAB:
        104400          SCOPE=TRAP+0            ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001314  012530                  .SCOPE
        104401          SCOP1=TRAP+1            ;CALL TO LOOP ON CURRENT DATA HANDLER
001316  012642                  .SCOP1
        104402          TYPE=TRAP+2             ;CALL TO TELETYPE OUTPUT ROUTINE
001320  012662                  .TYPE
        104403          INSTR=TRAP+3            ;CALL TO ASCII STRING INPUT ROUTINE
001322  012770                  .INSTR
        104404          INSTER=TRAP+4           ;CALL TO INPUT ERROR HANDLER
001324  013106                  .INSTER
        104405          PARAM=TRAP+5            ;CALL TO NUMERICAL DATA INPUT ROUTINE
001326  013140                  .PARAM
        104406          SAVO5=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
001330  013354                  .SAVO5
        104407          RESO5=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
001332  013414                  .RESO5
        104410          CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
001334  013446                  .CONVRT
        104411          CNVRT=TRAP+11           ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
001336  013452                  .CNVRT
        104412          MSTCLR=TRAP+12          ;CALL TO ISSUE MASTER CLEAR
001340  012324                  .MSTCLR
        104413          MEMCLR=TRAP+13          ;CALL TO CLEAR ALL SCRATCH PAD MEMORIES
001342  012200                  .MEMCLR
        104414          CKSWR=TRAP+14           ;CALL TO ALLOW SWREG TO BE LOADED FROM TTY
001344  014354                  .CKSWR
        104415          CNTLU=TRAP+15           ;CALL TO ALLOW LOADING OF SWREG FROM TTY
001346  014430                  .CNTLU

                        ;*****************************************************************
                        ;*****************************************************************

                        ;DQ11 VECTOR AND REGISTER INDIRECT POINTERS

001350  000C00  DQRVEC: 0               ;POINTER TO DQ11 RECEIVER INTERRUPT VECTOR
001352  000000  DQRLVL: 0               ;POINTER TO DQ11 RECEIVER INTERRUPT SERVICE PS
001354  000C00  DQTVEC: 0               ;POINTER TO DQ11 TRANSMITTER INTERRUPT VECTOR
001356  000000  DQTLVL: 0               ;POINTER TO DQ11 TRANSMITTER INTERRUPT SERVICE PS
001360  000000  DQRCSR: 0               ;POINTER TO DQ11 RECEIVER CONTROL REGISTER
001362  000000  DQRCSH: 0               ;POINTER TO HIGH BYTE OF DQ11 RECEIVER CONTROL REGISTER
```

```
1160  001364  000000                DQTCSR: 0               ;POINTER TO DQ11 TRANSMITTER CONTROL REGISTER
1161  001366  000000                DQERR:  0               ;POINTER TO DQ11 ERROR REGISTER
1162  001370  000000                DQREG:  0               ;POINTER TO HIGH BYTE OF ERROR REGISTER
1163  001372  000000                DQSEC:  0               ;POINTER TO DQ11 SECONDARY REGISTER
1164  001374  000000                DQSECH: 0               ;POINTER TO HIGH BYTE OF DQ11 SECONDARY REGISTER
1165
1166
1167
1168                                        ;DQ11 STATUS TABLE AND ADDRESS ASSIGNMENTS
1169
1170          001400                .=1400
1171  001400  000001                DQCR00: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 00
1172  001402  000001                DQST00: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 00
1173  001404  000001                DQCR01: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 01
1174  001406  000001                DQST01: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 01
1175  001410  000001                DQCR02: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 02
1176  001412  000001                DQST02: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 02
1177  001414  000001                DQCR03: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 03
1178  001416  000001                DQST03: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 03
1179  001420  000001                DQCR04: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 04
1180  001422  000001                DQST04: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 04
1181  001424  000001                DQCR05: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 05
1182  001426  000001                DQST05: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 05
1183  001430  000001                DQCR06: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 06
1184  001432  000001                DQST06: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 06
1185  001434  000001                DQCR07: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 07
1186  001436  000001                DQST07: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 07
1187  001440  000001                DQCR10: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 10
1188  001442  000001                DQST10: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 10
1189  001444  000001                DQCR11: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 11
1190  001446  000001                DQST11: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 11
1191  001450  000001                DQCR12: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 12
1192  001452  000001                DQST12: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 12
1193  001454  000001                DQCR13: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 13
1194  001456  000001                DQST13: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 13
1195  001460  000001                DQCR14: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 14
1196  001462  000001                DQST14: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 14
1197  001464  000001                DQCR15: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 15
1198  001466  000001                DQST15: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 15
1199  001470  000001                DQCR16: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 16
1200  001472  000001                DQST16: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 16
1201  001474  000001                DQCR17: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 17
1202  001476  000001                DQST17: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 17
1203  001500  000001                DQACTV: .BLKW   1       ;HOLD ACTIVE BITS FOR TESTING
1204  001502  000001                SAVACT: .BLKW   1       ;SAVE NUMBER OF ACTIVE DQ11S
1205  001504  000001                DQNUM:  .BLKW   1       ;OCTAL NUMBER OF TOTAL NUMBER OD DQ11S
1206  001506  000001                DQCSR:  .BLKW   1       ;CSR OF DQ11 UNDER TEST
1207  001510  000001                DQSTAT: .BLKW   1       ;VECTOR AND CONFIGURATION STATJS OF DQ11 UNDER TEST
1208
1209                                        ;PROGRAM INITIALIZATION
1210                                        ;LOCK OUT INTERRUPTS
1211                                        ;SET UP PROCESSOR STACK
1212                                        ;SET UP POWER FAIL VECTOR
1213                                        ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
1214                                        ;TYPE TITLE MESSAGE
1215
```

```
1216  001512  012737  000340  177776  .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1217  001520  012706  001200                  MOV     #STACK,SP               ;SET UP STACK
1218  001524  012737  014256  000024          MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
1219  001532  013737  001504  001276          MOV     DQNUM,SAVNUM
1220  001540  105037  001311                  CLRB    STFLG                   ;CLEAR START FLAG
1221  001544  005037  001230                  CLR     PASCNT                  ;CLEAR PASS COUNT
1222  001550  105037  001312                  CLRB    ERRFLG                  ;CLEAR ERROR FLAG
1223  001554  005037  001302                  CLR     RUNFLG
1224  001560  012737  001400  001300          MOV     #1400,CREAM
1225  001566  005037  001232                  CLR     ERRCNT                  ;CLEAR ERROR COUNT
1226  001572  005037  001234                  CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
1227  001576  012737  000001  001226          MOV     #1,TSTNO                ;SET UP FOR TEST 1
1228  001604  012737  001512  001214          MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
1229                                                                          ;TESTING STARTS
1230  001612  105737  001310                  TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
1231  001616  001075                          BNE     12$
1232  001620  104402  001000                  TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
1233  001624  105137  001310                  COMB    INIFLG                  ;IF NOT SET FLAG AND DO
1234
1235  001630  012737  177570  001200          MOV     #DSWR,SWR               ;MOV HARDWARE SWR TO SWR
1236  001636  012737  177570  001202          MOV     #DLIGHTS,LIGHTS         ;MOV DISPLAY LIGHTS TO LIGHTS
1237  001644  013746  000006                  MOV     @#6,-(SP)               ;SAVE VECTORS
1238  001650  013746  000004                  MOV     @#4,-(SP)
1239  001654  012737  001674  000004          MOV     #64$,@#4                        ;SET UP FOR TIMEOUT
1240  001662  022777  177777  177310          CMP     #-1,@SWR                ;REFERENCE HARDWARE SWITCH REGISTER
1241  001670  001402                          BEQ     65$
1242  001672  000407                          BR      66$
1243  001674  022626                  64$:    CMP     (SP)+,(SP)+             ;ADJUST STACK
1244  001676  012737  000176  001200  65$:    MOV     #SWREG,SWR              ;POINT TO SOFTWARE SWITCH REG
1245  001704  012737  000174  001202          MOV     #DISPREG,LIGHTS         ;POINT TO SOFT DISPLAY REG
1246  001712  012637  000004          66$:    MOV     (SP)+,@#4               ;RESTORE VECTORS
1247  001716  012637  000006                  MOV     (SP)+,@#6
1248  001722  005737  000042                  TST     @#42                    ;UNDER MONITOR
1249  001726  001005                          BNE     67$
1250  001730  022737  000176  001200          CMP     #SWREG,SWR              ;IS SWREG USED
1251  001736  001001                                  BNE     67$
1252  001740  104415                          CNTLU
1253  001742  105777  177232          67$:    TSTB    @SWR
1254  001746  100402                          BMI     .+6
1255  001750  004737  000220                  JSR     PC,CSRMAP
1256  001754  104402  015104                  TYPE    ,XHEAD
1257  001760  012737  001400  001244          MOV     #1400,TEMP1
1258  001766  017737  177252  001246          MOV     @TEMP1,TEMP2
1259  001774  001406                          BEQ     .+16
1260  001776  104410                          CONVRT
1261  002000  015132                          XSTATQ
1262  002002  062737  000002  001244          ADD     #2,TEMP1
1263  002010  000766                          BR      .-22
1264  002012  032777  000001  177160  12$:    BIT     #SW00,@SWR
1265  002020  001424                          BEQ     1$
1266  002022  104402                          TYPE
1267  002024  015025                          MNEW
1268  002026  005000                          CLR     R0
1269  002030  000000                          HALT
1270  002032  104414                          CKSWR
1271  002034  027737  177140  001502          CMP     @SWR,SAVACT
```

```
002042  101404                          BLOS    11$
002044  104402                          TYPE
002046  014666                          MERR3
002050  000000                          HALT
002052  000776                          BR      .-2
002054  017737  177120  001500  11$:    MOV     @SWR,DQACTV
002062  013700  001500                  MOV     DQACTV,R0
002066  000000                          HALT
002070  104414                          CKSWR
002072  012700  000300          1$:     MOV     #300,R0
002076  012701  000302                  MOV     #302,R1
002102  010120                  2$:     MOV     R1,(R0)+
002104  005021                          CLR     (R1)+
002106  022021                          CMP     (R0)+,(R1)+
002110  022700  001000                  CMP     #1000,R0
002114  001372                          BNE     2$

                                        ;TEST START AND RESTART

002116  012737  000340  177776  .BEGIN: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
002124  012706  001200                  MOV     #STACK,SP               ;SET UP STACK
002130  005737  000042                  TST     @#42                    ;IS PROGRAM UNDER MONITOR CONTROL
002134  001040                          BNE     3$
002136  104414                          CKSWR                           ;CHECK FOR <↑G>
002140  032777  000004  177032          BIT     #BIT2,@SWR              ;CHECK FOR LOCK ON TEST
002146  001411                          BEQ     1$
002150  104402  014724                  TYPE    .MLOCK
002154  012737  000240  012540          MOV     #NOP,TTST
002162  012737  000240  012542          MOV     #NOP,TTST+2             ;SET UP TO LOCK
002170  000406                          BR      2$
002172  013737  012636  012540  1$:     MOV     BRW,TTST
002200  013737  012640  012542          MOV     BRX,TTST+2              ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
002206  032777  000002  [illegible]  2$:     BIT     #SWC1,@SWR              ;IF SWC1=1, GET STARTING PC
002214  001410                          BEQ     3$
002216  104403                          INSTR
002220  [illegible]                          #TSTPC
002222  104405                          PARAM
002224  002254                          TST1
002226  [illegible]                          #LAST
002230  [illegible]                          RETURN
002232     001                  .BYTE   1
002233     001                  .BYTE   1
002234  000403                          BR      4$
002236  012737  002254  001214  3$:     MOV     #TST1,RETURN            ;START AT TEST 1
002244  104402  014614          4$:     TYPE    .MR                     ;TYPE R
002250  000177  176740                  JMP     @RETURN                 ;START TESTING
                                ;       TEST 1
                                ;*************************************************************************
002254  012737  000001  001226  TST1:   MOV     #1,TSTNO
002262  012737  002646  001214          MOV     #TST2,RETURN
002270  012737  002646  001216          MOV     #TST2,NEXT
002276  105737  001302                  TSTB    RUNFLG          ;IS THIS MY FIRST TIME HERE?
[illegible]                             [illegible]             ;[illegible]
002304  012737  000001  001304          MOV     #BIT0,RUN       ;SET RUN POINTER.
002312  012737  000020  001306          MOV     #16.,RUNCNT     ;SET FOR MAX OF 16 DQ11'S PER SYSTEM
002320  105137  001302                  COMB    RUNFLG          ;SET RUN FLAG
```

```
 [illegible]  002324  033737  001304  001500  1$:     BIT     RUN,DQACTV      ;FIND AN ACTIVE DQ11 TO TEST.
 [illegible]  002332  001032                          BNE     3$              ;BR IF I FOUND ONE TO TEST.
 [illegible]  002334  005737  001500                  TST     DQACTV          ;FIND OUT IF THERE ARE NO DQ11 ACTIVE.
 [illegible]  002340  001423                          BEQ     2$              ;BR TO FATAL ERROR. WHY AM I HERE IF NO ACTIVE DQ11'S???
 [illegible]  002342  000257                          CCC                     ;CLEAR ALL THE CONDITION CODES OF CPU
 [illegible]  002344  006137  001304                  ROL     RUN             ;UPDATE RUN POINTER
 [illegible]  002350  062737  000004  001300          ADD     #4,CREAM        ;UPDATE ADDRESS POINTER.
 [illegible]  002356  005337  001306                  DEC     RUNCNT          ;DEC NUMBER OF TIMES I LOOKED AT ACTIVE.
 [illegible]  002362  001360                          BNE     1$              ;BR AND KEEP LOOKING.
 [illegible]  002364  012737  000020  001306          MOV     #16.,RUNCNT     ;START RESTORING MY POINTERS.
 [illegible]  002372  012737  001400  001300          MOV     #1400,CREAM     ;RESTORE  ADDRESS POINTER
 [illegible]  002400  012737  000001  001304          MOV     #1,RUN          ;RESTORE RUN POINTER.
 [illegible]  002406  000746                          BR      1$              ;KEEP ON TESTING.
 [illegible]  002410  104402                  2$:     TYPE                    ;ALLERT OPERATOR OF FATAL ERROR
 [illegible]  002412  014617                          MERR2                   ;NO DQ11 ACTIVE. WHY AM I HERE???
 [illegible]  002414  000000                          HALT                    ;YOU MUST RELOAD DQ11 DIAGNOSTIC!!
 [illegible]  002416  000776                          BR      .-2             ;STICK HERE ON CONT.
 [illegible]  002420  000257                  3$:     CCC                     ;CLEAR CPU COND. CODES
 [illegible]  002422  006137  001304                  ROL     RUN             ;UPDATE RUN. ACTIVE DQ11 FOUND.
 [illegible]  002426  017737  176646  001506          MOV     @CREAM,DQCSR    ;PLACE ADDRESSS OF DQ11 AT DQCSR
 [illegible]  002434  062737  000002  001300          ADD     #2,CREAM        ;UPDATE ADDRESS POINTER
 [illegible]  002442  017737  176632  001510          MOV     @CREAM,DQSTAT   ;PLACE STATUS OF DQ11 AT DQSTAT
 [illegible]  002450  062737  000002  001300          ADD     #2,CREAM        ;UPDATE ADDRESS POINTER
 [illegible]  002456  013737  001506  001360          MOV     DQCSR,DQRCSR
 [illegible]  002464  013737  001510  001350          MOV     DQSTAT,DQRVEC
 [illegible]  002472  042737  177007  001350          BIC     #177007,DQRVEC
 [illegible]  002500  013737  001350  001352          MOV     DQRVEC,DQRLVL   ;GENERATE ADDRESS OF RECEIVER INTERRUPT SERVICE PS
 [illegible]  002506  062737  000002  001352          ADD     #2,DQRLVL
 [illegible]  002514  013737  001352  001354          MOV     DQRLVL,DQTVEC   ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT VECTOR
 [illegible]  002522  062737  000002  001354          ADD     #2,DQTVEC
 [illegible]  002530  013737  001354  001356          MOV     DQTVEC,DQTLVL   ;GENERATE ADDRESS  OF TRANSMITTER INTERRUPT SERVICE PS
 [illegible]  002536  062737  000002  001356          ADD     #2,DQTLVL
 [illegible]  002544  013737  001360  001362          MOV     DQRCSR,DQRCSH
 [illegible]  002552  005237  001362                  INC     DQRCSH          ;GENERATE ADDRESS OF HIGH BYTE
 [illegible]  002556  013737  001360  001364          MOV     DQRCSR,DQTCSR   ;GENERATE ADDRESS OF TRANSMITTER CONTROL REGISTER
 [illegible]  002564  062737  000002  001364          ADD     #2,DQTCSR
 [illegible]  002572  013737  001364  001366          MOV     DQTCSR,DQERR    ;GENERATE ADDRESS OF ERROR REGISTER
 [illegible]  002600  062737  000002  001366          ADD     #2,DQERR
 [illegible]  002606  013737  001366  001370          MOV     DQERR,DQREG     ;GENERATE ADDRESS OF HIGH BYTE OF ERROR REGISTER
 [illegible]  002614  005237  001370                  INC     DQREG
 [illegible]  002620  013737  001370  001372          MOV     DQREG,DQSEC     ;GENERATE ADDRESS OF SECONDARY REGISTER
 [illegible]  002625  005237  001372                  INC     DQSEC
 [illegible]  002632  013737  001372  001374          MOV     DQSEC,DQSECH    ;GENERATE ADDRESS OF HIGH BYTE
 [illegible]  002640  005237  001374                  INC     DQSECH
 [illegible]  002644  000240                  NOP
```

```
                                        ;
                                        ;CABLE TEST.
                                        ;TEST OF DATA REALIBILITY THROUGH
                                        ;CABLE AND LEVEL CONVERTERS.

                                        ;NOTE: IF JUMPER IS NOT INSTALLED
                                        ;      AT END OF CABLE THIS TEST IS NOT
                                        ;       DONE
                                        ;


                                ;  TEST 2
                                ;*************************************************************************
002646  012737  000002  001226  TST2:   MOV     #2,TSTNO
002654  012737  003226  001216          MOV     #TST3,NEXT

002662  032737  100000  001510  CKSYN1: BIT     #SYNBIT,DQSTAT  ;ADJUST POINTER FOR NUMBER OF SYNC CHARS.
002670  001003                          BNE     1$              ;BR IF TWO SYNC CHARS SELECTED.
002672  105037  015352                  CLRB    SYNC            ;SET FIRST SYNC TO 0 IF ONE SYNC SEL.
002676  000403                          BR      CKDN            ;BR TO CONT.
002700  112737  000026  015352  1$:     MOVB    #26,SYNC        ;RESET FIRST SYNC TO 26
002706  000240                  CKDN:   NOP                     ;CONTINUE TEST.

002710  032737  040000  001510          BIT     #JUMBIT,DQSTAT  ;DOES THE TEST JUMPER EXIST.
002716  001005                          BNE     .+14            ;BR IF YES
002720  013737  001216  001214          MOV     NEXT,RETURN     ;PREPARE TO DO NEXT TEST.
002726  000177  176262                  JMP     @RETURN         ;GOTO NEXT TEST
002732  005000                          CLR     R0              ;ZERO DATA POINTER.
002734  012704  015354                  MOV     #TXBUFF,R4      ;SET BUFFER POINTER
002740  110024                  1$:     MOVB    R0,(R4)+        ;FILL TX BUFFER WITH BINARY COUNT PATTERN
002742  105200                          INCB    R0              ;UPDATE CHAR.
002744  001375                          BNE     1$              ;BR IF MORE TO DO
002746  104413                  2$:     MEMCLR                  ;CLEAR DC11 MEMORIES.
002750  005000                          CLR     R0              ;ZERO COUNTER POINTER
002752  012704  015756                  MOV     #RXBUFF,R4      ;PREPARE TO ZERO ALL RX BUFFER.
002756  105024                  3$:     CLRB    (R4)+           ;START CLEARING.
002760  105200                          INCB    R0              ;UPDATE
002762  001375                          BNE     3$              ;BR IF NOT ALL CLEARED.
002764  105077  176400                  CLRB    @DQREG          ;SELECT RX BA PRI.
002770  012777  015755  176374          MOV     #RXBUFF,@DQSEC  ;LOAD IT.
002776  105277  176366                  INCB    @DQREG          ;SELECT RX WC PRI.
003002  012777  177400  176362          MOV     #-400,@DQSEC    ;LOAD IT FOR 400(8) CHARS.
003010  105277  176354                  INCB    @DQREG          ;SEL TX BA PRI.
003014  012777  015352  176350          MOV     #SYNC,@DQSEC    ;LOAD IT.
003022  105277  176342                  INCB    @DQREG          ;SEL TX WC PR.
003026  012777  177376  176336          MOV     #-402,@DQSEC    ;SET 400(8) CHARS AND TWO SYNC.
003034  112777  000011  176326          MOVB    #11,@DQREG      ;SEL SYNC REGISTER
003042  013777  015350  176322          MOV     SYNC,@DQSEC     ;LOAD IT.
003050  105277  176314                  INCB    @DQREG          ;GET MISC REGISTER
003054  012777  004000  176310          MOV     #4000,@DQSEC    ;SET FOR EIGHTBITS.
003062  005037  001244                  CLR     TEMP1           ;SET DELAY.....
003066  012737  000020  001246          MOV     #20,TEMP2       ;" " "
003074  005277  176260                  INC     @DQRCSR         ;SET RX GO!!
003100  005277  176260                  INC     @DQTCSR         ;SET TX GO!!
003104  005777  176256          4$:     TST     @DQERR          ;ANY ERRORS
003110  100007                          BPL     7$              ;BR IF NO ERRORS
```

```
 1434  003112  012700  176242          MOV     @DQRCSR,R0
 1435  003116  012701  176242          MOV     @DQTCSR,R1
 1436  003122  012702  176240          MOV     @DQERR,R2
 1437  003126  104007                  HLT     7                       ;THE DQ11 ERROR FLAG IS SET.
 1438  003130  105777  176224  7$:     TSTB    @DQRCSR                 ;IS RX PRI DONE SET?
 1439  003134  100407                  BMI     5$                      ;BR IF YES
 1440  003136  005237  001244          INC     TEMP1                   ;DELAY.....
 1441  003142  001360                  BNE     4$                      ;
 1442  003144  005337  001246          DEC     TEMP2                   ;" " "
 1443  003150  001355                  BNE     4$                      ;
 1444  003152  104000                  HLT                             ;RX PRI. DONE FAILED TO SET.
 1445  003154  005000          5$:     CLR     R0                      ;ZERO COUNTER.
 1446  003156  005037  001252          CLR     TEMP4                   ;CLEAR STORAGE
 1447  003162  005037  001254          CLR     TEMP5                   ;SAME..
 1448  003166  012704  015354          MOV     #TXBUFF,R4              ;GET TX BUFFER AREA
 1449  003172  012705  015756          MOV     #RXBUFF,R5              ;GET RX BUFFER AREA
 1450  003176  112437  001254  6$:     MOVB    (R4)+,TEMP5             ;LOAD FOR ERROR CALL
 1451  003202  112537  001252          MOVB    (R5)+,TEMP4             ;" " " "
 1452  003206  023737  001254  001252  CMP     TEMP5,TEMP4             ;DOES DATA CHECK OUT OK?
 1453  003214  001401                  BEQ     .+4                     ;BR IF GOOD DATA.
 1454  003216  104004                  HLT     4                       ;DATA COMPARISON ERROR.
 1455  003220  105200                  INCB    R0                      ;UPDATE COUNTER
 1456  003222  001365                  BNE     6$                      ;BR IF MORE DATA TO CHECK.
 1457  003224  104420                  SCOPE                           ;SCOPE THE TEST.
 1458
 1459
 1460                                  ;  CEIVER STRIP SYNC TEST.
                                       ;EST THAT THE RECEIVER CAN STRIP SYNC.
                                       ;CHARACTERS FROM 000-025 WILL BE TRANSFERED
                                       ;WITH A TRAIL OF SYNC CHARS. FOLLOWING (026).
                                       ;THE TRANSMITTER AND RECEIVER BUFFERS ARE BOTH
                                       ;CLEARED BEFORE THE TEST IS EXECUTED.
                                       ;A TOTAL OF 400 CHARS. WILL BE TRANSMITTER INTO
                                       ;THE RECEIVER. WHEN RX PRI. DONE SETS;
                                       ;THE RECEIVER BUFFER IS CHECKED FOR ANY SYNC
                                       ;CHARACTERS (026). IF NONE ARE FOUND THEN THE
                                       ;RECEIVER DID INDEED STRIP SYNC.


                                  ;  TEST 3
                                  ;*********************************************************************
 1475  003226  012737  000003  001226  TST3:   MOV     #3,TSTNO
 1476  003234  012737  003520  001216          MOV     #TST4,NEXT
 1477  003242  005000                          CLR     R0                      ;CLEAR POINTER
 1478  003244  012704  015756                  MOV     #RXBUFF,R4              ;SET THE RX BUFFER
 1479  003250  105024                  1$:     CLRB    (R4)+                   ;BEGIN TO CLEAR THE RX BUFFER
 1480  003252  105200                          INCB    R0                      ;ALL DONE?
 1481  003254  001375                          BNE     1$                      ;BR IF NO
 1482  003256  005000                          CLR     R0                      ;RESET R0 TO ZERO
 1483  003260  012704  015354                  MOV     #TXBUFF,R4              ;GET TX BUFFER
 1484  003264  105024                  8$:     CLRB    (R4)+                   ;BEGIN TO CLEAR THE TX BUFFER
 1485  003266  105200                          INCB    R0                      ;DONE YET?
 1486  003270  001375                          BNE     8$                      ;BR IF NO
 1487
 1488  003272  005000                          CLR     R0                      ;RESET R0
 1489  003274  012704  015354                  MOV     #TXBUFF,R4              ;GET TX BUFFER
```

```
1490  003300  110024                  7$:   MOVB    R0,(R4)+        ;START FILLING WITH CHARS.
1491  003302  105200                        INCB    R0              ;UPDATE POINTER
1492  003304  022700  000026                CMP     #26,R0          ;SYNC YET?
1493  003310  001373                        BNE     7$              ;BR IF NO
1494  003312  012702  000026                MOV     #26,R2          ;SET FOR SYNC CHAR.
1495  003316  110024                  6$:   MOVB    R0,(R4)+        ;FILL TX BUFFER WITH SYNC CHARS.
1496  003320  105202                        INCB    R2              ;MORE TO GO?
1497  003322  100375                        BPL     6$              ;BR IF YES
1498
1499  003324  104413                        MEMCLR                  ;CLEAR ALL DQ11
1500  003326  105077  176036                CLRB    @DQREG          ;SEL RX BA PRI.
1501  003332  012777  015756  176032        MOV     #RXBUFF,@DQSEC  ;SET BUFFER.
1502  003340  105277  176024                INCB    @DQREG          ; RX WC PRI.
1503  003344  012777  177400  176020        MOV     #-400,@DQSEC    ;256. CHARS
1504  003352  105277  176012                INCB    @DQREG          ;TX BA PRI.
1505  003356  012777  015352  176006        MOV     #SYNC,@DQSEC    ;SET TO XMIT SYNC CHARS.
1506  003364  105277  176000                INCB    @DQREG          ;TX WC PRI.
1507  003370  012777  177400  175774        MOV     #-400,@DQSEC    ;256. CHARS
1508  003376  112777  000011  175764        MOVB    #11,@DQREG      ;SYNC REGISTER
1509  003404  013777  015350  175760        MOV     .SYNC,@DQSEC    ;LOAD SYNC REGISTER
1510  003412  105277  175752                INCB    @DQREG          ;MISC. REGISTER.
1511  003416  012777  004010  175746        MOV     #4010,@DQSEC    ;EIGHT BITS AND TEST LOOP
1512  003424  012777  000003  175726        MOV     #00003,@DQRCSR  ;GO AND STRIP SYNC.
1513  003432  005277  175726                INC     @DQTCSR         ;SET TX GO.
1514  003436  005000                        CLR     R0              ;PREPARE TO DELAY.
1515  003440  005001                        CLR     R1              ; SAME
1516  003442  105777  175712          2$:   TSTB    @DQRCSR         ;RX DONE?
1517  003446  100406                        BMI     9$              ;BR IF YES
1518  003450  062700  000001                ADD     #1,R0           ;D
1519  003454  001372                        BNE     2$              ; E
1520  003456  105201                        INCB    R1              ;  L
1521  003460  100370                        BPL     2$              ;   A
1522                                                                ;    Y
1523  003462  104000                        HLT                     ;RX PRI DONE NOT SET.
1524  003464  005077  175670          9$:   CLR     @DQRCSR         ;DISABLE RX
1525  003470  005077  175670                CLR     @DQTCSR         ;DISABLE TX
1526  003474  005000                        CLR     R0              ;SET COUNTER
1527  003476  012704  015756                MOV     #RXBUFF,R4      ;GET RX BUFFER
1528  003502  122724  000026          3$:   CMPB    #26,(R4)+       ;ANY SYNC CHARS?
1529  003506  001001                        BNE     .+4             ;BR IF NONE
1530  003510  104000                        HLT                     ;RECEIVER DID NOT STRIP SYNC.
1531  003512  105200                        INCB    R0              ;UPDATE COUNTER
1532  003514  001372                        BNE     3$              ;BR IF MORE TO GO.
1533  003516  104400                        SCOPE                   ;SCOPE THIS TEST.
1534
1535
1536
1537
1538
1539                                        ;DQ11   MEMORY TRANSFER TESTS
1540                                        ;
1541                                        ;IF THE PROCESSOR HAS AT LEAST 9K OF MEMORY
1542                                        ;THIS TEST WILL BE EXECUTED.
1543                                        ;
1544                                        ;THE FIRST PART EXERCISES THE TRANSMITTER ALONE
1545                                        ;TRANSMITTING FROM ADD.20000 TO LAST MEMORY ADD.
```

```
1546                                                ;UNDER 28K MINUS 400 (SAVE ABL). THE DATA IS
1547                                                ;VERIFIED ONE CHAR AT A TIME. TWO EIGHT BIT
1548                                                ; CHARS ARE TRANSFERED AT ONE TIME.
1549                                                ;
1550                                                ;AFTER THE TX ALONE THEN THE TRANSMITTER AND RECEIVER
1551                                                ;ARE EXERCISED TOGETHER IN THE SAME
1552                                                ;MANNER AS DESCRIBED ABOVE.
1553                                                ;
1554
1555
1556                                        ;  TEST 4
1557                                        ;*************************************************************************
1558  003520  012737  000004  001226  TST4:  MOV     #4,TSTNO
1559  003526  012737  004456  001216         MOV     #TST5,NEXT
1560  003534  104413                        MEMCLR
1561  003536  013705  000004                MOV     @#4,R5          ;SAVE THE TIME OUT VECTOR
1562  003542  012737  004432  000004        MOV     #10$,@#4        ;LOAD TRAP VECTOR
1563  003550  005737  020000                TST     @#20000         ;CHECK FOR 8K OF MEMORY.
1564  003554  000240                        NOP                     ;
1565  003556  012737  003606  000004        MOV     #2$,@#4         ;CPU DOES HAVE AT LEAST 8K.
1566  003564  012700  020000                MOV     #20000,R0       ;PREPARE TO SIZE MEMORY TO 28K
1567  003570  062700  020000          1$:   ADD     #20000,R0       ;CHECK MEMORY.
1568  003574  005710                        TST     (R0)            ;EXIST?
1569  003576  000240                        NOP                     ;
1570  003600  022700  160000                CMP     #160000,R0      ;28K HIT YET.
1571  003604  001371                        BNE     1$              ;BR IF NO
1572  003606  012706  001200          2$:   MOV     #STACK,SP       ;ADJUST STACK
1573  003612  012737  003706  001220        MOV     #4$,LOCK        ;SET FOR LOCK (SW09=1)
1574  003620  010537  000004                MOV     R5,@#4          ;RESET TRAP VECTOR.
1575  003624  162700  005000                SUB     #5000,R0        ;ALLOW ROOM FOR DDP2 MONITOR.
1576  003630  010037  004454                MOV     R0,LIMIT.HI     ;SAVE LAST MEMORY ADDRESS
1577  003634  005002                        CLR     R2              ;ZERO DATA CHAR POINTER
1578  003636  012700  020000                MOV     #20000,R0       ;PREPARE TO FILL MEMORY WITH BINARY COUNT
1579  003642  110220          3$:   MOVB    R2,(R0)+        ;START FILL
1580  003644  005202                        INC     R2              ;UPDATE CHAR.
1581  003646  122702  000026                CMPB    #26,R2          ;DOES IT EQUAL THE SYNC CHAR?
1582  003652  001001                        BNE     .+4             ;BR IF NO
1583  003654  005202                        INC     R2              ;BUMP ONE HIGHER
1584  003656  020037  004454                CMP     R0,LIMIT.HI     ;IS ALL OF MEMORY FULL?
1585  003662  001367                        BNE     3$              ;BR IF NO.
1586  003664  112777  000002  175476        MOVB    #2,@DQREG       ;SEL TX BA PRI.
1587  003672  012777  020000  175472        MOV     #20000,@DQSEC   ;SET TX BA TO FIRST ADD IN 8K
1588  003700  012700  020000                MOV     #20000,R0       ;SET SOFTWARE POINTER.
1589  003704  000406                        BR      5$              ;CONTINUE TEST
1590  003706  112777  000002  175454  4$:   MOVB    #2,@DQREG       ;SEL TX BA PRI.
1591  003714  162777  000002  175450        SUB     #2,@DQSEC       ;GO BACKWARDS FOR SCOPE ROUTINE
1592  003722  112777  000002  175440  5$:   MOVB    #2,@DQREG       ;SEL TX BA PRI.
1593  003730  017737  175436  001252        MOV     @DQSEC,TEMP4    ;SAVE THE TX BA PRI. ADDRESS
1594  003736  104412                        MSTCLR                  ;DO INIT DQ11
1595  003740  112777  000002  175422        MOVB    #2,@DQREG       ;SEL TX BA PRI.
1596  003746  013777  001252  175416        MOV     TEMP4,@DQSEC    ;RELOAD TX BA PRI.
1597  003754  105277  175410                INCB    @DQREG          ;SEL TX WC PRI.
1598  003760  012777  177776  175404        MOV     #-2,@DQSEC      ;SET FOR A TWO EIGHT BIT XFER
1599  003766  112777  000012  175374        MOVB    #MISC.,@DQREG   ;SEL MISC REGISTER
1600  003774  012777  004012  175370        MOV     #4012,@DQSEC    ;SET 8 BITS TEST LOOP AND AUTO STEP
1601  004002  005277  175356                INC     @DQTCSR         ;SET TX GO.
```

```
1602  004006  005777  175354                    TST     @DQERR          ;ANY ERRORS?
1603  004012  100001                            BPL     .+4             ;BR IF NO ERRORS
1604  004014  104000                            HLT                     ;DQ11 ERROR FLAG SET.
1605  004016  112777  000013  175344            MOVB    #13,@DQREG      ;SEL TX MUX REG
1606  004024  017737  175342  001252            MOV     @DQSEC,TEMP4    ;READ TX MUX
1607  004032  011037  001254                    MOV     (R0),TEMP5      ;READ SOFTWARE POINTER
1608  004036  023710  001252                    CMP     TEMP4,(R0)      ;IS THE DATA CORRECT
1609  004042  001401                            BEQ     .+4             ;BR IF GOOD
1610  004044  104005                            HLT     5               ;DATA COMPARISON ERROR.
1611
1612  004046  104401                            SCOP1                   ;LOCK ON CHARACTER (SW09=1)
1613  004050  005720                            TST     (R0)+           ;UPDATE SOFTWARE POINTER.
1614  004052  020037  004454                    CMP     R0,LIMIT.HI     ;ALL DONE?
1615  004056  001321                            BNE     5$              ;BR IF NO
1616
1617
1618                                            ;THE ABOVE WAS FOR THE TX ONLY
1619                                            ;
1620                                            ;THE BELOW ROUTINE EXECISES BOTH THE
1621                                            ;TX AND RX TOGETHER.
1622                                            ;
1623                                            ;NOTE THAT THE RX CA SHOULD BE
1624                                            ;400 (8) LOCATIONS HIGHER THAN THE TX CA.
1625
1626
1627
1628  004060  012737  004106  001220            MOV     #6$,LOCK        ;SET FOR LOCK ON CHAR(SW09=1)
1629  004066  012700  020000                    MOV     #20000,R0       ;SET FIRT ADD IN 8K
1630  004072  112777  000006  175270            MOVB    #6,@DQREG       ;SEL TX BA SEC
1631  004100  010077  175266                    MOV     R0,@DQSEC       ;LOAD TX BA SEC
1632  004104  000413                            BR      7$              ;CONT TEST.
1633  004106  112777  000006  175254  6$:       MOVB    #6,@DQREG       ;SEL TX BA SEC.
1634  004114  162777  000002  175250            SUB     #2,@DQSEC       ;KILL LAST XFER
1635  004122  105077  175242                    CLRB    @DQREG          ;SEL RX BA PRI.
1636  004126  162777  000002  175236            SUB     #2,@DQSEC       ;KILL LAST XFER
1637  004134  112777  000006  175226  7$:       MOVB    #6,@DQREG       ;SEL TX BA SEC
1638  004142  017737  175224  001252            MOV     @DQSEC,TEMP4    ;SAVE IT
1639  004150  104412                            MSTCLR                  ;INIT DQ11
1640  004152  105077  175212                    CLRB    @DQREG          ;SEL RX BA PRI
1641  004156  013777  001252  175206            MOV     TEMP4,@DQSEC    ;LOAD IT
1642  004164  062777  000400  175200            ADD     #400,@DQSEC     ;UPDATE IT
1643  004172  105277  175172                    INCB    @DQREG          ;SEL RX WC PRI
1644  004176  012777  177776  175166            MOV     #-2,@DQSEC      ;SET FOR TWO CHARS.
1645  004204  105277  175160                    INCB    @DQREG          ;SEL TX BA PRI
1646  004210  012777  015352  175154            MOV     #SYNC,@DQSEC    ;TX SYNC ON PRI.
1647  004216  105277  175146                    INCB    @DQREG          ;SEL TX WC PRI.
1648  004222  012777  177776  175142            MOV     #-2,@DQSEC      ;SET FOR TWO SYNC.
1649  004230  112777  000006  175132            MOVB    #6,@DQREG       ;LOAD TX BA SEC.
1650  004236  013777  001252  175126            MOV     TEMP4,@DQSEC    ;LOAD IT
1651  004244  105277  175120                    INCB    @DQREG          ;SEL TX WC SEC
1652  004250  012777  177776  175114            MOV     #-2,@DQSEC      ;SET FOR TWO CHAR XFER
1653  004256  112777  000011  175104            MOVB    #11,@DQREG      ;SEL THE SYNC REG
1654  004264  013777  015350  175100            MOV     .SYNC,@DQSEC    ;LOAD IT
1655  004272  105277  175072                    INCB    @DQREG          ;SEL MISC REG.
1656  004276  012777  004010  175066            MOV     #4010,@DQSEC    ;EIGHT BITS TEST LOOP
1657  004304  005277  175050                    INC     @DQRCSR         ;SET RX GO..
```

```
1658  004310  005277  175050              INC     @DQTCSR         ;SET TX GO..
1659  004314  005037  001244              CLR     TEMP1           ;DELAY COUNTER
1660  004320  012737  000005  001246      MOV     #5,TEMP2        ;" " " "
1661  004326  105777  175036        8$:   TSTB    @DQRCSR         ;RX PRI DONE?
1662  004332  100407                     BMI     9$              ;BR IF YES
1663  004334  005237  001244              INC     TEMP1           ;DELAY
1664  004340  001372                     BNE     8$              ;
1665  004342  005337  001246              DEC     TEMP2           ;" " "
1666  004346  001367                     BNE     8$              ;
1667  004350  104000                     HLT                     ;RX PRI DONE NOT SET.
1668  004352  005777  175010        9$:   TST     @DQERR          ;ANY ERRORS
1669  004356  100001                     BPL     .+4             ;BR IF NO.
1670  004360  104000                     HLT                     ;DQ11 ERROR FLAG SET.
1671  004362  011037  001254              MOV     (R0),TEMP5      ;SET EXPECTED
1672  004366  105077  174776              CLRB    @DQREG          ;SELECT RX BA PRI.
1673  004372  017701  174774              MOV     @DQSEC,R1       ;GET RX BA
1674  004376  162701  000002              SUB     #2,R1           ;GET LAST XFER
1675  004402  011137  001252              MOV     (R1),TEMP4      ;GET ACTUAL DATA
1676  004406  021037  001252              CMP     (R0),TEMP4      ;IS DATA OF?
1677  004412  001401                     BEQ     .+4             ;BR IF GOOD
1678  004414  104006                     HLT     6               ;DATA COMPARISON ERROR
1679
1680  004416  104401                     SCOP1                   ;LOCK ON DATA (SW09=1)
1681  004420  005720                     TST     (R0)+           ;UPDATE SOFTWARE POINTER
1682  004422  020037  004454              CMP     R0,LIMIT.HI     ;ALL DONE?
1683  004426  001242                     BNE     7$              ;BR IF NO
1684  004430  000410                     BR      11$             ;END TEST
1685  004432  022626              10$:   POP2SP                  ;ADJUST STACK POINTER
1686  004434  010537  000004              MOV     R5,@#4          ;RESET TRAP VECTOR
1687  004440  013737  001216  001214      MOV     NEXT,RETURN     ;DO NEXT TEST
1688  004446  000177  174542              JMP     @RETURN         ;
1689  004452  104400              11$:   SCOPE                   ;SCOPE THIS TEST
1690
1691  004454                     LIMIT.HI:
1692  004454  000000                     0
1693
```

```
1694
1695                                                    ;
1696                                                    ;TEST OF "ENTER T" AND "EXIT T"
1697                                                    ;TRANSMITTER TRIGGERED
1698                                                    ;
1699                                                    ;TEST TO TRANSMITT ONE CHARACTER ENTERING T
1700                                                    ;CHECKING THE BCC THEN OVERFLOWING
1701                                                    ;CAUSING AN EXIT T THEN MAKING SURE
1702                                                    ;THAT THE BCC WENT TO ZERO.
1703                                                    ;
1704                                                    ;NOTE: IF THE BCC DOES NOT EXIST THESE TESTS WILL NOT BE DONE
1705
1706                                            ;  TEST 5
1707                                            ;*****************************************************************
1708  004456  012737  000005  001226  TST5:     MOV     #5,TSTNO
1709  004464  012737  005000  001216            MOV     #TST6,NEXT
1710  004472  032737  002000  001510            BIT     #ABBIT,@DQSTAT
1711  004500  001005                            BNE     .+14
1712  004502  012737  012342  001214            MOV     #.EOP,RETURN
1713  004510  000177  174500                    JMP     @RETURN
1714  004514  104412                            MSTCLR
1715  004516  104412                            MSTCLR                          ;ISSUE A MASTER CLEAR
1716  004520  012737  000351  004776            MOV     #351,WORD               ;SET CHAR
1717  004526  112777  000017  174634            MOVB    #17,@DQREG              ;SELECT POLY REGISTER
1718  004534  012777  000200  174630            MOV     #200,@DQSEC             ;SET POLY FOR LRC 8
1719  004542  012737  000011  015330            MOV     #11,COUNT               ;SET COUNT TO 11
1720  004550  112777  000002  174612            MOVB    #2,@DQREG               ;SET TX BA PRI
1721  004556  012777  004776  174606            MOV     #WORD,@DQSEC            ;LOAD TX BA
1722  004564  112777  000123  174576            MOVB    #123,@DQREG             ;SEL TX CC-WRITE EN ENTER T
1723  004572  012777  177777  174572            MOV     #-1,@DQSEC              ;SEL TX CC TO -1
1724  004600  112777  000067  174562            MOVB    #67,@DQREG              ;SEL TX CC SEC-WRITE EN EXIT T
1725  004606  005077  174560                    CLR     @DQSEC                  ;SET TX CC TO ZERO
1726  004612  112777  000012  174550            MOVB    #MISC.,@DQREG           ;SEL MISC REG
1727  004620  012777  004012  174544            MOV     #4012,@DQSEC            ;EIGHT BITS TEST LOOP AUTO SET
1728  004626  005277  174532                    INC     @DQTCSR                 ;SET TX GO
1729  004632  027777  174526  174524            CMP     @DQTCSR,@DQTCSR         ;WAIST TIME.
1730  004640  027777  174520  174516            CMP     @DQTCSR,@DQTCSR         ;WAIST TIME
1731  004646  027777  174512  174510            CMP     @DQTCSR,@DQTCSR         ;WAIST TIME
1732  004654  005277  174512                    INC     @DQSEC                  ;CLOCK UP
1733  004660  005377  174506                    DEC     @DQSEC                  ;CLOCK DOWN
1734  004664  005277  174502          1$:       INC     @DQSEC                  ;START THE CHAR UP
1735  004670  005377  174476                    DEC     @DQSEC                  ;       .       DOWN
1736  004674  005337  015330                    DEC     COUNT                   ;DONE YET?
1737  004700  001371                            BNE     1$                      ;BR IF NO
1738  004702  112777  000016  174460            MOVB    #16,@DQREG              ;SET TX BCC REG
1739  004710  017705  174456                    MOV     @DQSEC,R5               ;STORE IT
1740  004714  022705  000351                    CMP     #351,R5                 ;DID CHAR GET INTO BCC
1741  004720  001401                            BEQ     .+4                     ;BR IF YES
1742  004722  104000                            HLT                             ;TX BCC FAILED
1743  004724  112777  000012  174436            MOVB    #MISC.,@DQREG           ;SEL MISC REG
1744  004732  012737  000010  015330            MOV     #10,COUNT               ;SET COUNT TO 10
1745  004740  005277  174426          2$:       INC     @DQSEC                  ;START CLOCKING BCC OUT UP
1746  004744  005377  174422                    DEC     @DQSEC                  ;       .               DOWN
1747  004750  005337  015330                    DEC     COUNT                   ;DONE YET
1748  004754  001371                            BNE     2$                      ;BR IF NO
1749  004756  112777  000016  174404            MOVB    #16,@DQREG              ;SEL TX BCC
```

```
1750  004764  005777  174402                  TST     @DQSEC          ;DID BCC GET SHIFTED OUT?
1751  004770  001401                          BEQ     .+4             ;BR IF YES
1752  004772  104000                          HLT                     ;TX BCC NOT ZERO
1753  004774  104400                          SCOPE                   ;SCOPE TEST
1754  004776  000000              WORD:       0
1755
1756
1757                                          ;
1758                                          ;TEST TO FORCE
1759                                          ;RECEIVER BCC ERROR
1760                                          ;
1761                                          ;THE TRANSMITTER CHARACTER COUNT
1762                                          ;WILL BE SET TO -300
1763                                          ;AND THE RECEIVER CHARACTER COUNT
1764                                          ;WILL BE SET TO -400
1765                                          ;THUS THE RECEIVER WILL RECEIVE
1766                                          ;MORE CHARACTERS THAN THE TRANSMITER
1767                                          ;TRANSMITTED.   *BCC ERROR*
1768                                          ;
1769                                          ;
1770
1771                                  ;  TEST 6
1772                                  ;*************************************************************************
1773  005000  012737  000006  001226  TST6:   MOV     #6,TSTNO
1774  005006  012737  005040  001214          MOV     #2$,RETURN
1775  005014  012737  005106  001216          MOV     #TST7,NEXT
1776  005022  104413                          MEMCLR                  ;CLEAR THE DEVICE
1777  005024  005000                          CLR R0                  ;SET R0 TO ZERO
1778  005026  012704  015354                  MOV     #TXBUFF,R4      ;SET POINTER FOR BUFFER
1779  005032  110024                  1$:     MOVB    R0,(R4)+        ;START FILLING THE BUFFER
1780  005034  105200                          INCB    R0              ;UPDATE THE DATA
1781  005036  001375                          BNE     1$              ;HAS THE BUFFER BEEN FILLED
1782  005040  104412                  2$:     MSTCLR
1783  005042  012737  120001  015326          MOV     #120001,XPOLY   ;SELECT CRC 16  FOR POLYNOMIAL
1784  005050  004537  010546                  JSR     R5,SYNBCC       ;GO PRIM THE DQ11
1785  005054  177400                          -400                    ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER
1786  005056  177500                          -300                    ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER
1787  005060  017705  174302                  MOV     @DQERR,R5       ;SAVE THE DQ ERROR REGISTER
1788  005064  005705                          TST     R5              ;DID AN ERROR OCCUR??
1789  005066  100401                          BMI     .+4             ;BR IF THE ERROR DID OCCUR
1790  005070  104001                          HLT     1               ;HALT THE DQ ERROR BIT IS NOT SET
1791  005072  032777  000100  174266          BIT     #BIT6,@DQERR    ;MAKE SURE IT WAS A RX BCC THAT CAUSED THE ERROR
1792  005100  001001                          BNE     .+4             ;BR IF THE RX BCC BIT IS SET
1793  005102  104001                          HLT     1               ;RX BCC ERROR BIT NOT SET
1794  005104  104400                          SCOPE                   ;SCOPE THIS TEST
1795
1796
1797                                          ;
1798                                          ;TEST OF TRANSMITTER BCC
1799                                          ;WITH POLYNOMIAL EQUAL TO 177777
1800                                          ;
1801                                          ;A FOUR HUNDRED BINARY COUNT
1802                                          ;DATA PATTERN IS RUN THROUGH
1803                                          ;THE BCC WITH A SHIFT BY SHIFT
1804                                          ;CHECK OF THE HARDWARE BY THE SOFTWARE.
1805                                          ;AT THE END THE TRANSMITTER IS ALSO
```

```
1806                                            ;CHECKED TO SEE IF THE BCC WAS SHIFTED
1807                                            ;OUT AND THAT THE BCC WENT TO ZERO.
1808                                            ;
1809                                            ;NOTE: THERE IS A TWO SHIFT DELAY
1810                                                   ;BEFORE THE TX BCC STARTS.
1811                                            ;
1812                                        ;  TEST 7
1813                                        ;*****************************************************************************
1814  005106  012737  000007  001226    TST7: MOV     #7,TSTNO
1815  005114  012737  005154  001214          MOV     #2$,RETURN
1816  005122  012737  005614  001216          MOV     #TST10,NEXT
1817  005130  104413                         MEMCLR                  ;CLEAR ALL THE DQ11
1818  005132  005000                         CLR     R0              ;SET POINTER TO ZERO
1819  005134  012704  015354                 MOV     #TXBUFF,R4      ;GET TX BUFFER
1820  005140  110024                  1$:    MOVB    R0,(R4)+        ;START FILLING TX BUFFER
1921  005142  105200                         INCB    R0              ;WITH A BINARY
1922  005144  001375                         BNE     1$              ;COUNT PATTERN
1823  005146  012737  177777  015326         MOV     #177777,XPOLY   ;SET POLYNOMIAL TO 177777
1824  005154  104412                  2$:    MSTCLR                  ;ISSUE MASTER CLEAR
1825  005156  104412                         MSTCLR
1826  005160  005037  015322                 CLR     CALBCC          ;SET CALCULATED BCC TO ZERO
1827  005164  012737  000016  001254         MOV     #16,TEMPS       ;SET TYPE OUT ERROR REG TO 16
1928  005172  005037  015332                 CLR     DATA            ;SET DATA OF BCC SIMULATOR TO ZERO
1829  005176  112777  000022  174164         MOVB    #22,@DQREG      ;WRITE EN TX BA PRI
1830  005204  012777  015354  174160         MOV     #TXBUFF,@DQSEC  ;SET TX BUFFER
1831  005212  112777  000123  174150         MOVB    #123,@DQREG     ;ENTER T WRITE EN ,TX CC PRI
1832  005220  012777  177400  174144         MOV     #-400,@DQSEC    ;SET FOR 400 CHARS
1833  005226  112777  000067  174134         MOVB    #67,@DQREG      ;EXIT "T",WRITE EN,TX CC SEC
1834  005234  005077  174132                 CLR     @DQSEC          ;SET FOR ZERO CHARS ON SEC
1535  005240  112777  000017  174122         MOVB    #17,@DQREG      ;SEL POLYNOMIAL REGISTER
1836  005246  013777  015326  174116         MOV     XPOLY,@DQSEC    ;LOAD IT
1837  005254  112777  000012  174106         MOVB    #MISC.,@DQREG   ;SEL MISC REGISTER
1838  005262  012777  004012  174102         MOV     #4012,@DQSEC    ;EIGHT BITS,TEST LOOP,AUTO STEP
1839  005270  005277  174070                 INC     @DQTCSR         ;SET TX GO
1840  005274  027777  174064  174062         CMP     @DQTCSR,@DQTCSR ;WAIST TIME.
1841  005302  027777  174056  174054         CMP     @DQTCSR,@DQTCSR ;WAIST TIME
1842  005310  027777  174050  174046         CMP     @DQTCSR,@DQTCSR ;WAIST TIME
1843  005316  005277  174050                 INC     @DQSEC          ;CLOCK UP---
1844  005322  005377  174044                 DEC     @DQSEC          ;CLOCK DOWN---
1845  005326  005277  174040                 INC     @DQSEC          ;CLOCK UP---
1846  005332  005377  174034                 DEC     @DQSEC          ;CLOCK DOWN---
1847  005336  005037  015330          3$:    CLR     COUNT           ;SET COUNT TO 0
1848  005342  013737  015332  005364         MOV     DATA,6$         ;SET DATA FOR SUBROUTINE
1849  005350  013737  015322  005366  4$:    MOV     CALBCC,7$       ;SET CALCULATED BCC FOR SUB ROUTINE
1850  005356  004537  012042                 JSR     R5,SIMBCC       ;GO TO BCC SIMULATOR ROUTINE
1851  005362  000001                   5$:    1                       ;THIS IS THE NUMBER OF SHIFTS FOR ROUTINE TO DO
1852  005364  000001                   6$:    .BLKW   1               ;THIS IS WHERE THE CHAR IS PLACED
1853  005366  000001                   7$:    .BLKW   1               ;THIS IS THE PREVIOUS BCC CALCULATED
1854  005370  112777  000012  173772         MOVB    #MISC.,@DQREG   ;RESELECT THE MISC REG
1855  005376  005277  173770                 INC     @DQSEC          ;CLOCK UP---
1856  005402  005377  173764                 DEC     @DQSEC          ;CLOCK DOWN---
1857  005406  112777  000016  173754         MOVB    #16,@DQREG      ;SEL THE TX BCC REGISTER
1858  005414  017737  173752  001252         MOV     @DQSEC,TEMP4    ;SAVE IT IN TEMP4
1859  005422  023737  015322  001252         CMP     CALBCC,TEMP4    ;ARE THE CALCULATED AND RECEIVED RESULTS THE SAME??
1860  005430  001401                         BEQ     .+4             ;BR IF GOOD (SAME)
1861  005432  104003                         HLT     3               ;BCC(S) ARE DIFFERNT..
```

```
1862  005434  000241                          CLC                     ;CLEAR THE CARRY BIT OF PSW
1863  005436  006037  005364                  ROR     5$              ;UPDATE MY DATA LOCATION
1864  005442  005237  015330                  INC     COUNT           ;UPDATE THE COUNT OF BITS PER CHAR
1865  005446  023727  015330  000010          CMP     COUNT,#8.       ;IS THIS CHARACTER DONE YET??
1866  005454  001335                          BNE     4$              ;BR IF CHAR NOT DONE
1867  005456  105237  015332                  INCB    DATA            ;GET NEW CHAR
1868  005462  001325                          BNE     3$              ;HAVE ALL CHARACTERS BEEN DONE
1869                                          ;
1870  005464  005037  015330                  CLR     COUNT           ;INIT COUNT
1871  005470  112777  000012  173672  8$:     MOVB    #MISC..@DQREG   ;RESELECT THE MISC REG
1872  005476  000241                          CLC                     ;CLEAR CARRY
1873  005500  006037  015322                  ROR     CALBCC          ;SHIFT OUT CALCULATED BCC
1874  005504  005277  173662                  INC     @DQSEC          ;CLOCK UP---
1875  005510  005377  173656                  DEC     @DQSEC          ;CLOCK DOWN---
1876  005514  112777  000016  173646          MOVB    #16,@DQREG      ;SEL TX BCC REGISTER
1877  005522  017737  173644  001252          MOV     @DQSEC,TEMP4    ;SAVE IT IN TEMP4
1878  005530  023737  015322  001252          CMP     CALBCC,TEMP4    ;ARE THEY THE SAME??
1879  005536  001401                          BEQ     .+4             ;BR IF SAME(GOOD)
1880  005540  104003                          HLT     3               ;BCC DIFFERENT
1881  005542  005237  015330                  INC     COUNT           ;UPDATE COUNT
1882  005546  022737  000020  015330          CMP     #16..COUNT      ;HAS ALL THE BCC BEEN SHIFED OUT
1883  005554  001345                          BNE     8$              ;BR IF MORE TO DO
1884  005556  112777  000012  173604          MOVB    #MISC.,@DQREG   ;SELECT THE MISC REGISTER
1885  005564  005277  173602                  INC     @DQSEC          ;GIVE ONE LAST CLOCK UP--
1886  005570  005377  173576                  DEC     @DQSEC          ;AND CLOCK DOWN--
1887  005574  112777  000016  173566          MOVB    #16,@DQREG      ;SEL THE TX BCC REGISTER
1888  005602  005777  173564                  TST     @DQSEC          ;DID THE TX BCC GO TO ZERO
1889  005606  001401                          BEQ     .+4             ;BR IF GOOD
1890  005610  104003                          HLT                     ;TX BCC NOT ZERO
1891  005612  104400                          SCOPE                   ;SCOPE TEST
1892
1893
1894
1895                                          ;
1896                                          ;TEST OF RECEIVER BCC
1897                                          ;WITH POLYNOMIAL EQUAL TO 177777
1898                                          ;
1899                                          ;A FOUR HUNDRED BINARY COUNT
1900                                          ;DATA PATTERN IS RUN THROUGH
1901                                          ;THE BCC WITH A SHIFT BY SHIFT
1902                                          ;CHECK OF THE HARDWARE COMPARE
1903                                          ;WITH THE SOFTWARE.
1904                                          ;NOTE   THERE IS ONE CHARACTER TIME DELAY
1905                                                  ;FOR THE BCC TO START.
1906                                                  ;ALSO THE IS ONE PAD CHAR
1907                                                  ;NEEDED AT THE END OF THE DATA.
1908                                          ;
1909                                    ;  TEST 10
1910                                    ;*****************************************************************************
1911  005614  012737  000010  001226  TST10:  MOV     #10,TSTNO
1912  005622  012737  005654  001214          MOV     #2$,RETURN
1913  005630  012737  006640  001216          MOV     #TST11,NEXT
1914  005636  104413                  1$:     MEMCLR                  ;CLEAR THE DEVICE
1915  005640  012737  177777  015326          MOV     #177777,XPOLY   ;SELXPOLY TO 177777
1916  005646  012737  000015  001254          MOV     #15,TEMP5       ;SET TYPE OUT REG TO 15
1917  005654  104412                  2$:     MSTCLR                  ;ISSUE A MASTER CLEAR
```

```
 1918  005656  104412                          MSTCLR
 1919  005660  005037  015322                  CLR     CALBCC          ;SET CALBCC TO ZERO
 1920  005664  005037  015332                  CLR     DATA            ;SET DATA TO ZERO
 1921  005670  112777  000020  173472          MOVB    #20,@DQREG      ;WRITE EN,RX BA PRI
 1922  005676  012777  015756  173466          MOV     #RXBUFF,@DQSEC  ;LOAD THE RX BA
 1923  005704  112777  000121  173456          MOVB    #121,@DQREG     ;ENTER "T",WRITE EN,RX CC PRI
 1924  005712  012777  177400  173452          MOV     #-400,@DQSEC    ;SET FOR FOUR HUNDRED CHARS
 1925  005720  112777  000024  173442          MOVB    #24,@DQREG      ;SEL THE RX BA SEC
 1926  005726  012777  015756  173436          MOV     #RXBUFF,@DQSEC  ;SET FOR THE PAD CHAR.
 1927  005734  112777  000065  173426          MOVB    #65,@DQREG      ;EXIT "T",WRITE EN,RX CC SEC
 1928  005742  012777  177777  173422          MOV     #-1,@DQSEC      ;SEL RX CC SEC FOR ONE PAD CHAR.
 1929  005750  112777  000017  173412          MOVB    #17,@DQREG      ;SEL THE POLYNO REGISTER
 1930  005756  013777  015326  173406          MOV     XPOLY,@DQSEC    ;LOAD IT WITH THE POLY
 1931  005764  112777  000012  173376          MOVB    #MISC.,@DQREG   ;SEL THE MISC REGISTER
 1932  005772  012777  004012  173372          MOV     #4012,@DQSEC    ;EIGHT BITS TEST LOOP AND AUTO STEP
 1933  006000  012777  010001  173352          MOV     #10001,@DQRCSR  ;SET RX ACTIVE AND GO!!
 1934  006006  012737  000010  015330  3$:     MOV     #8.,COUNT       ;SET FOR ONE CHAR TIME DELAY
 1935  006014  013737  015332  015316          MOV     DATA,TMPDAT     ;SAVE THE DATA
 1936  006022  005137  015316                  COM     TMPDAT          ;COMPLIMENT IT FOR BIT WINDOW USE.
 1937  006026  005037  001250          4$:     CLR     TEMP3           ;INIT LOC
 1938  006032  006037  015316                  ROR     TMPDAT          ;SHIFT OUT ONE BIT OF DATA
 1939  006036  106037  001250                  RORB    TEMP3           ;BRING IT IN FROM CARRY
 1940  006042  042777  000200  173322          BIC     #BIT7,@DQSEC    ;CLEAR THE BIT WINDOW
 1941  006050  053777  001250  173314          BIS     TEMP3,@DQSEC    ;PLACE DATA ON BIT WINDOW
 1942  006056  005277  173310                  INC     @DQSEC          ;CLOCK UP---
 1943  006062  005377  173304                  DEC     @DQSEC          ;CLOCK DN---
 1944  006066  005337  015330                  DEC     COUNT           ;CHAR DONE??
 1945  006072  001355                          BNE     4$              ;BR IF NOT DONE
 1946  006074  105237  015332                  INCB    DATA            ;UPDATE DATA
 1947  006100  005037  015330          5$:     CLR     COUNT           ;INIT COUNT
 1948  006104  013737  015332  006144          MOV     DATA,8$         ;MOV DATA TO SUB ROUTINE USE AREA
 1949  006112  005337  006144                  DEC     8$              ;SET SUBROUTINE TO ONE LESS THAT RX GETS
 1950  006116  013737  015332  015316          MOV     DATA,TMPDAT     ;SAVE DATA
 1951  006124  005137  015316                  COM     TMPDAT          ;COMPLIMENT DATA FOR BIT WINDOW USE
 1952  006130  013737  015322  006146  6$:     MOV     CALBCC,9$       ;MOV CALCULATED BCC TO SUB ROUTINE USE
 1953  006136  004537  012042                  JSR     R5,SIMBCC       ;GO AND CALCULATE BCC (SOFTWARE)
 1954  006142  000001                  7$:     1                       ;THIS IS NUMBER OF SHIFTS TO BE DONE
 1955  006144  000001                  8$:     .BLKW   1               ;THIS IS WHERE THE DATA IS PLACED
 1956  006146  000001                  9$:     .BLKW   1               ;THIS IS WHERE THE PREVIOUS BCC IS PLACED
 1957  006150  112777  000012  173212          MOVB    #MISC.,@DQREG   ;RESELECT THE MISC REGISTER
 1958  006156  005037  001250                  CLR     TEMP3           ;INIT LOC
 1959  006162  006037  015316                  ROR     TMPDAT          ;SHIFT OUT DATA BIT
 1960  006166  106037  001250                  RORB    TEMP3           ;CATCH IT IN TEMP3
 1961  006172  042777  000200  173172          BIC     #BIT7,@DQSEC    ;CLEAR THE BIT WINDOW
 1962  006200  053777  001250  173164          BIS     TEMP3,@DQSEC    ;LOAD THE DATA
 1963  006206  005277  173160                  INC     @DQSEC          ;CLOCK UP---
 1964  006212  005377  173154                  DEC     @DQSEC          ;CLOCK DN---
 1965  006216  112777  000015  173144          MOVB    #15,@DQREG      ;SEL RX BCC REGISTER
 1966  006224  017737  173142  001252          MOV     @DQSEC,TEMP4    ;SAVE THE BCC
 1967  006232  023737  015322  001252          CMP     CALBCC,TEMP4    ;IS IT CORRECT??
 1968  006240  001401                          BEQ     .+4             ;BR IF GOOD
 1969  006242  104003                          HLT     3               ;BCC NOT WHAT EXPECTED
 1970  006244  000241                          CLC                     ;CLEAR THE CARRY BIT
 1971  006246  006037  006144                  ROR     8$              ;SHIFT THE DATA
 1972  006252  005237  015330                  INC     COUNT           ;UPDATE THE COUNT
 1973  006256  022737  000010  015330          CMP     #8.,COUNT       ;IS THE CHARACTER DONE??
```

```
 1974  006264  001321                          BNE     6$              ;BR IF CHAR NOT DONE
 1975  006266  105237  015332                  INCB    DATA            ;UPDATA DATA
 1976  006272  001302                          BNE     5$              ;BR IF NOT ALL CHARS DONE.
 1977  006274  012737  000003  015346  10$:    MOV     #3,LOC1         ;POINTER****
 1978  006302  013737  015322  006326          MOV     CALBCC,21$      ;SAVE CALBCC
 1979  006310  013737  015322  015344          MOV     CALBCC,STORE1   ;
 1980  006316  004537  012042                  JSR     R5,SIMBCC       ;GO FINISH THE BCC
 1981  006322  000010                          6.                      ;SHIFTS REQUIRED
 1982  006324  000377                          377                     ;DATA CHARACTER
 1983  006326  000001                  21$:    .BLKW   1               ;PREVIOS BCC
 1984  006330  013737  015322  015334          MOV     CALBCC,SAVBCC   ;SAVE THE BCC
 1985  006336  013737  015344  015322          MOV     STORE1,CALBCC   ;RESTORE THE BCC
 1986  006344  012737  000377  015332          MOV     #377,DATA       ;DATA =377
 1987  006352  013737  015334  015316          MOV     SAVBCC,TMPDAT   ;PUSH IN THE LOW BYTE OF THE BCC
 1988  006360  005137  015316                  COM     TMPDAT          ;INTO THE RX
 1989  006364  000421                          BR      12$             ;
 1990  006366  013737  015334  015332  11$:    MOV     SAVBCC,DATA     ;MOVE THE CALBCC TO DATA FOR SUBROUTINE
 1991  006374  113737  015335  015316          MOVB    SAVBCC+1,TMPDAT ;MOVE THE HIGH BYTE OF CALBCC TO PLACED INTO THE RECEIVE
 1992  006402  005137  015316                  COM     TMPDAT          ;PREPARE IT FOR THE BIT WINDOW
 1993  006406  000410                          BR      12$             ;GO TO MAIN PART OF TEST
 1994  006410  113737  015335  015332  16$:    MOVB    SAVBCC+1,DATA   ;MOVE THE HIGH BYTE OF THE CALBCC TO DATA
 1995  006416  012737  000377  015316          MOV     #377,TMPDAT     ;PAD CHAR FOR RX
 1996  006424  005137  015316                  COM     TMPDAT          ;PREPARE IT FOR THE BIT WINDOW
 1997  006430  005037  015230          12$:    CLR     COUNT           ;INIT COUNT (THIS IS FOR NUMBER OF BITS PER CHAR)
 1998  006434  113737  015332  006456          MOVB    DATA,13$        ;LOAD THE CHAR FOR SUBROUTINE
 1999  006442  013737  015322  006460  15$:    MOV     CALBCC,14$      ;LOAD THE CALBCC FOR THE SUBROUTINE
 2000  006450  004537  012042                  JSR     R5,SIMBCC       ;GO TO THE SUBROUTINE
 2001  006454  000001                          1                       ;THIS IS THE NUMBER OF SHIFTS TO BE DONE
 2002  006456  000001                  13$:    .BLKW   1               ;THIS IS THE CHAR FOR THE SUBROUTINE
 2003  006460  000001                  14$:    .BLKW   1               ;THIS IS THE PREVIOUS BCC
 2004  006462  112777  000012  172700          MOVB    #MISC..@DQREG   ;SEL THE MISC REGISTER
 2005  006470  005037  001250                  CLR     TEMP3           ;INIT LOC       ;
 2006  006474  006037  015316                  ROR     TMPDAT          ;SHIFT OUT A BIT OF DATA
 2007  006500  106037  001250                  RORB    TEMP3           ;BRING IT FROM CARRY INTO TEMP3
 2008  006504  042777  000200  172660          BIC     #BIT7,@DQSEC    ;CLEAR THE BIT WINDOW
 2009  006512  053777  001250  172652          BIS     TEMP3,@DQSEC    ;PLACE DATA ON THE BIT WINDOW
 2010  006520  005277  172646                  INC     @DQSEC          ;CLOCK UP---
 2011  006524  005377  172642                  DEC     @DQSEC          ;CLOCK DN---
 2012  006530  112777  000015  172632          MOVB    #15,@DQREG      ;SEL THE RX BCC REGISTER
 2013  006536  017737  172630  001252          MOV     @DQSEC,TEMP4    ;SAVE IT IN TEMP4
 2014  006544  023737  015322  001252          CMP     CALBCC,TEMP4    ;IS THE BCC CORRECT??
 2015  006552  001401                          BEQ     .+4             ;BR IF GOOD
 2016  006554  104003                          HLT     3               ;BCC ERROR IN RX
 2017  006556  000241                          CLC                     ;CLEAR THE CARRY BIT
 2018  006560  006037  006456                  ROR     13$             ;SHIFT THE CHARACTER TO THE RIGHT
 2019  006564  005237  015330                  INC     COUNT           ;UPDATE THE COUNT
 2020  006570  022737  000010  015330          CMP     #8.,COUNT       ;IS THIS CHARACTER DONE
 2021  006576  001321                          BNE     15$             ;BR IF NOT DONE
 2022  006600  005337  015346                  DEC     LOC1            ;ALTER THE RETURN POINTER
 2023  006604  022737  000002  015346          CMP     #2,LOC1         ;WHERE SHOULD I GO??
 2024  006612  001665                          BEQ     11$             ;IF LOC1=2 GOTO 11$
 2025  006614  022737  000001  015346          CMP     #1,LOC1 :
 2026  006622  001672                          BEQ     16$             ;IF LOC1=1 GOTO 16$
 2027  006624  017737  172636                  MOV     @DQSEC,R5       ;SAVE THE ERROR REGISTER
 2028  006630  005705                          TST     R5              ;DID AN ERROR OCCUR??
 2029  006632  100001                          BPL     .+4             ;BR IF NO ERROR
```

```
 2030  006634  104001                          HLT     1               ;DQ11 ERROR FLAG SET
 2031  006636  104400                          SCOPE                   ;SCOPE THIS TEST
 2032
 2033
 2034
 2035
 2036
 2037
 2038
 2039
 2040
 2041
 2042
 2043
 2044
 2045
 2046
 2047
 2048
 2049
 2050
 2051
 2052
 2053
 2054
 2055
 2056
 2057
 2058
 2059
 2060
 2061
 2062
 2063
 2064
 2065
 2066
 2067
 2068
 2069
 2070
 2071
 2072
 2073
 2074
 2075
 2076
 2077
 2078
 2079
 2080
 2081
 2082
 2083
 2084
 2085
                                                ;TEST OF TRANSMITTER AND RECEIVER
                                                ;BCC WITH A POLYNOMIAL OF
                                                ;CRC 16: X16+X15+X2+1
                                                ;NOTE: IN THIS TEST IT IS UP TO
                                                ;      THE HARDWARE TO DISCOVER
                                                ;      AN ERROR IF ONE OCCURS.

                                        ;  TEST 11
                                        ;*****************************************************************
 006640  012737  000011  001226 TST11:  MOV     #11.TSTNO
 006646  012737  006704  001216         MOV     #TST12,NEXT
 006654  104412                         MSTCLR                  ;CLEAR DQ11 WITH A MASTER CLEAR
 006656  012737  120001  015326         MOV     #120001,XPOLY   ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
 006664  004737  010160                 JSR     PC,STBCC        ;TRANSFER CHARACTERS.
 006670  017705  172472                 MOV     @DQERR,R5       ;SAVE THE ERROR REGISTER
 006674  005705                         TST     R5              ;DID AN ERROR OCCUR??
 006676  100001                         BPL     .+4             ;BR IF NO ERROR
 006700  104001                         HLT     1               ;AN ERROR OCCURED
 006702  104400                         SCOPE
                                                ;
                                                ;TEST OF TRANSMITTER AND RECEIVER
                                                ;BCC WITH A POLYNOMIAL OF
                                                ;CRC 12: X12+X11+X3+X2+X+1
                                                ;NOTE:  IN THIS TEST IT IS UP TO
                                                ;       THE HARDWARE TO DISCOVER
                                                ;       AN ERROR IF ONE OCCURS

                                        ;  TEST 12
                                        ;*****************************************************************
 006704  012737  000012  001226 TST12:  MOV     #12,TSTNO
 006712  012737  006750  001216         MOV     #TST13,NEXT
 006720  104412                         MSTCLR                  ;CLEAR DQ11 WITH A MASTER CLEAR
 006722  012737  007401  015326         MOV     #7401,XPOLY     ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
 006730  004737  010160                 JSR     PC,STBCC        ;TRANSFER CHARACTERS.
 006734  017705  172426                 MOV     @DQERR,R5       ;SAVE THE ERROR REGISTER
 006740  005705                         TST     R5              ;DID AN ERROR OCCUR??
 006742  100001                         BPL     .+4             ;BR IF NO ERROR
 006744  104001                         HLT     1               ;AN ERROR OCCURED
 006746  104400                         SCOPE
                                                ;
                                                ;TEST OF TRANSMITTER AND RECEIVER
                                                ;BCC WITH A POLYNOMIAL OF
                                                ;CRC-CCITT: X16+X12+X5+X+1
                                                ;NOTE:  IN THIS TEST IT IS UP TO
```

```
                                                        ;THE HARDWARE TO DISCOVER
                                                        ;AN ERROR IF ONE OCCURES


                                        ;  TEST 13
                                        ;*****************************************************************
007750  012737  000013  001226  TST13:  MOV     #13,TSTNO
007756  012737  007014  001216          MOV     #TST14,NEXT
007764  104412                          MSTCLR                  ;CLEAR DQ11 WITH A MASTER CLEAR
007766  012737  102010  015326          MOV     #102010,XPOLY   ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
007774  004737  010160                  JSR     PC,$TBCC        ;TRANSFER CHARACTERS.
008000  017705  172362                  MOV     @DQERR,R5       ;SAVE THE ERROR REGISTER
008004  005705                          TST     R5              ;DID AN ERROR OCCUR??
008006  100001                          BPL     .+4             ;BR IF NO ERROR
008010  104001                          HLT     1               ;AN ERROR OCCURED
008012  104400                          SCOPE


                                                        ;
                                                        ;TEST OF TRANSMITTER AND RECEIVER
                                                        ;BCC WITH A POLYNOMIAL OF
                                                        ;LRC 8: X8+1
                                                        ;NOTE:  IN THIS TEST IT IS UP TO
                                                        ;       THE HARDWARE TO DISCOVER
                                                        ;       AN ERROR IF ONE OCCURES.

                                        ;  TEST 14
                                        ;*****************************************************************
007014  012737  000014  001226  TST14:  MOV     #14,TSTNO
007022  012737  007060  001216          MOV     #TST15,NEXT
007030  104412                          MSTCLR                  ;CLEAR DQ11 WITH A MASTER CLEAR
007032  012737  000200  015326          MOV     #200,XPOLY      ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
007040  004737  010160                  JSR     PC,$TBCC        ;TRANSFER CHARACTERS.
007044  017705  172316                  MOV     @DQERR,R5       ;SAVE THE ERROR REGISTER
007050  005705                          TST     R5              ;DID AN ERROR OCCUR??
007052  100001                          BPL     .-4             ;BR IF NO ERROR
007054  104001                          HLT     1               ;AN ERROR OCCURED
007056  104400                          SCOPE


                                                        ;
                                                        ;TEST OF TRANSMITTER AND RECEIVER
                                                        ;BCC WITH A POLYNOMIAL OF
                                                        ;LRC 16: X16+1
                                                        ;NOTE:  IN THIS TEST IT IS UP TO
                                                        ;       THE HARDWARE TO DISCOVER
                                                        ;       AN ERROR IF ONE OCCURES

                                        ;  TEST 15
                                        ;*****************************************************************
007060  012737  000015  001226  TST15:  MOV     #15,TSTNO
007066  012737  007124  001216          MOV     #TST16,NEXT
007074  104412                          MSTCLR                  ;CLEAR DQ11 WITH A MASTER CLEAR
```

```
2142  007076  012737  100000  015326          MOV     #100000,XPOLY   ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
2143  007104  004737  010160                  JSR     PC,STBCC        ;TRANSFER CHARACTERS.
2144  007110  017705  172252                  MOV     @#DQERR,R5      ;SAVE THE ERROR REGISTER
2145  007114  005705                          TST     R5              ;DID AN ERROR OCCUR??
2146  007116  100001                          BPL     .+4             ;BR IF NO ERROR
2147  007120  104001                          HLT     1               ;AN ERROR OCCURED
2148  007122  104400                          SCOPE
2149
2150
2151
2152
2153                                          ;
2154                                          ;TEST OF RECEIVER AND TRANSMITTER
2155                                          ;BCC USING CRC 16 FOR POLYNOMIAL.
2156                                          ;
2157                                          ;THIS TEST USES IDLE MODE TO
2158                                          ;GET INTO TRANSPARENCY
2159                                          ;AND IF AN ERROR SHOULD OCCUR
2160                                          ;IT MUST BE REPORTED BY THE HARDWARE.
2161                                          ;
2162
2163                                  ;  TEST 16
2164                                  ;*************************************************************************
2165  007124  012737  000016  001226  TST16:  MOV     #16.,TSTNO
2166  007132  012737  007164  001214          MOV     #2$,RETURN
2167  007140  012737  007220  001216          MOV     #TST17,NEXT
2168  007146  104413                          MEMCLR                  ;CLEAR THE DEVICE
2169  007150  005000                          CLR R0                  ;SET R0 TO ZERO
2170  007152  012704  015354                  MOV     #TXBUFF,R4      ;SET POINTER FOR BUFFER
2171  007156  110024                  1$:     MOVB    R0,(R4)+        ;START FILLING THE BUFFER
2172  007160  105200                          INCB    R0              ;UPDATE THE DATA
2173  007162  001375                          BNE     1$              ;HAS THE BUFFER BEEN FILLED
2174  007164  104412                  2$:     MSTCLR
2175  007166  012737  120001  015326          MOV     #120001,XPOLY   ;SELECT CRC 16  FOR POLYNOMIAL
2176  007174  004537  010546                  JSR     R5,SYNBCC       ;GO PRIM THE DQ11
2177  007200  177400                          -400                    ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER
2178  007202  177400                          -400                    ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER
2179  007204  017705  172156                  MOV     @#DQERR,R5      ;SAVE THE DQ ERROR REGISTER
2180  007210  005705                          TST     R5              ;DID AN ERROR OCCUR??
2181  007212  100001                          BPL     .+4             ;BR IF THE ERROR DID NOT OCCUR
2182  007214  104001                          HLT     1               ;HALT THE DQ ERROR BIT IS SET
2183  007216  104400                          SCOPE                   ;SCOPE THIS TEST
2184
2185
2186
2187                                          ;
2188                                          ;TEST OF TRANSMITTER AND RECEIVER
2189                                          ;BCC WITH A POLYNOMIAL OF 177777
2190                                          ;
2191                                          ;THIS TEST USES IDLE MODE TO
2192                                          ;GET INTO TRANSPARENCY. IF AN
2193                                          ;ERROR SHOULD HAPPEN, THE HARDWARE
2194                                          ;MUST FLAG IT.
2195                                          ;
2196                                  ;  TEST 17
2197                                  ;*************************************************************************
```

```
 2198  007220  012737  000017  001226  TST17:  MOV     #17,TSTNO
 2199  007226  012737  007260  001214          MOV     #2$,RETURN
 2200  007234  012737  007314  001216          MOV     #TST20,NEXT
 2201  007242  104413                          MEMCLR                  ;CLEAR THE DEVICE
 2202  007244  005000                          CLR RO                  ;SET RO TO ZERO
 2203  007246  012704  015354                  MOV     #TXBUFF,R4      ;SET POINTER FOR BUFFER
 2204  007252  110024                  1$:     MOVB    RO,(R4)+        ;START FILLING THE BUFFER
 2205  007254  105200                          INCB    RO              ;UPDATE THE DATA
 2206  007256  001375                          BNE     1$              ;HAS THE BUFFER BEEN FILLED
 2207  007260  104412                  2$:     MSTCLR                  ;ISSUE MASTER CLEAR
 2208  007262  012737  177777  015326          MOV     #177777,XPOLY   ;SELECT 177777  FOR POLYNOMIAL
 2209  007270  004537  010546                  JSR     R5,SYNBCC       ;GO PRIM THE DQ11
 2210  007274  177400                          -400                    ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER
 2211  007276  177400                          -400                    ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER
 2212  007300  017705  172062                  MOV     @DQERR,R5       ;SAVE THE DQ ERROR REGISTER
 2213  007304  005705                          TST     R5              ;DID AN ERROR OCCUR??
 2214  007306  100001                          BPL     .+4             ;BR IF THE ERROR DID NOT OCCUR
 2215  007310  104001                          HLT     1               ;HALT THE DQ ERROR BIT IS SET
 2216  007312  104400                          SCOPE                   ;SCOPE THIS TEST
 2217
 2218                                          ;
 2219                                          ;TEST OF TRANSMITTER AND RECEIVER
 2220                                          ;BCC WITH ALL POLYNOMIALS
 2221                                          ;BETWEEN 000000 AND 177777
 2222                                          ;SENDING ONE CHARACTER (351)
 2223                                          ;AND CALCLLATING WHAT THE BCC
 2224                                          ;SHOULD BE AND COMPARING IT
 2225                                          ;WITH THE TRANSMITTER BCC AND RECEIVER B C
 2226                                          ;
 2227                                          ;NOTE:  SW 09=1 WILL FREEZE THE POLYNOMIAL
 2228                                                  ;WHEN SW 09 IS ASSERTED.
 2229
 2230                                  ;  TEST 20
 2231                                  ;*****************************************************************
 2232  007314  012737  000020  001226  TST20:  MOV     #20,TSTNO
 2233  007322  012737  000003  001222          MOV     #3,ICOUNT
 2234  007330  012737  007450  001216          MOV     #TST21,NEXT
 2235  007336  012737  007350  001220          MOV     #1$,LOCK
 2236  007344  005037  015326                  CLR     XPOLY           ;INIT SET XPOLY TO ZERO
 2237  007350  104412                  1$:     MSTCLR
 2238  007352  004737  011066                  JSR     PC,TYBCC        ;GO AND PRIM THE DQ11
 2239  007356  004537  012042                  JSR     R5,SIMBCC       ;GO AND CALCULATE WHAT THE BCC SHOULD BE
 2240  007362  000010                          8.                      ;THIS IS FOR EIGHT BITS PER CHAR (NUMBER OF SHIFTS)
 2241  007364  000351                          351                     ;THIS IS THE DATA CHARACTER
 2242  007366  000000                          0                       ;THIS IS THE PREVIOUS BCC
 2243  007370  112777  000015  171772          MOVB    #15,@DQREG      ;SEL THE RX BCC REGISTER
 2244  007376  012705  000015                  MOV     #15,R5          ;SAVE THE REGISTER
 2245  007402  017701  171764                  MOV     @DQSEC,R1       ;SAVE THE BCC
 2246  007406  023701  015322                  CMP     CALBCC,R1       ;IS THE BCC RIGHT??
 2247  007412  001401                          BEQ     .+4             ;BR IF YES
 2248  007414  104002                          HLT     2               ;BCC ERROR
 2249  007416  012705  000016                  MOV     #16,R5          ;SEL THE TX BCC REGISTER
 2250  007422  013701  015340                  MOV     SEC16,R1        ;THIS IS WHERE THE TX BCC WAS STORED
 2251  007426  023701  015322                  CMP     CALBCC,R1       ;IS IT RIGHT??
 2252  007432  001401                          BEQ     .+4             ;BR IF GOOD
 2253  007434  104002                          HLT     2               ;TX BCC ERROR
```

```
                                        SCOP1                   ;DOES THE USER HAVE SW09=1??
007436  104401
007440  005237  015326                  INC     XPOLY           ;UPDATE THE POLYNO
007444  001341                          BNE     1$              ;BE IF NOT ALL POLYNO HAVE BEEN DONE
007446  104400                          SCOPE                   ;SCOPE THIS TEST
                                        ;
                                        ;TEST OF BIT 06 OF M    REGISTER
                                        ;POLYNOMIAL 16-24
                                        ;
                                        ;TEST WILL SEND ONE CHARACTER AT
                                        ;A TIME CHECKING THAT THE BCC
                                        ;OF RECEIVER AND TRANSMITTER
                                        ;ARE CORRECT.
                                        ;CHARACTER SENT: 000-377
                                        ;
                                ;  TEST 21
                                ;*************************************************************************
007450  012737  000021  001226  TST21:  MOV     #21,TSTNO
007456  012737  012342  001216          MOV     #.EOP,NEXT
                                        ;PART 1         READ/WRITE TEST OF POLY 16-24
                                        ;TEST OF ALL READ WRITE BITS IN POLY 16-24
                                        ;BY RUNNING A BINARY COUNT PATTERN TROUGH
                                        ;THE REGISTER.
007464  012737  007476  001220          MOV     #1$,LOCK        ;SET FOR LOCK ON TEST(SW09=1)
007472  005037  001254                  CLR     TEMP5           ;ZERO POINTER
007476  104412                  1$:     MSTCLR                  ;INIT DQ11
007500  112777  000012  171662          MOVB    #MISC.,@DQREG   ;SEL MISC REG
007506  012777  000100  171656          MOV     #BIT6,@DQSEC    ;SEL POLY 16-24
007514  112777  000017  171646          MOVB    #17,@DQREG      ;SEL POLY REGISTER
007522  053777  001254  171642          BIS     TEMP5,@DQSEC    ;LOAD WITH CHAR.
007530  017737  171636  001252          MOV     @DQSEC,TEMP4    ;READ CHAR BACK.
007536  023737  001254  001252          CMP     TEMP5,TEMP4     ;IS IT CORRECT?
007544  001401                          BEQ     .+4             ;BR IF YES.
007546  104006                          HLT     6               ;POLY READ/WRITE ERROR.
007550  104401                          SCOP1                   ;LOCK ON CHAR (SW09=1)
007552  105237  001254                  INCB    TEMP5           ;UPDATE CHAR.
007556  001347                          BNE     1$              ;BR IF MORE TO GO
                                        ;PART 2         RX AND TX BCC TESTS
007560  005037  015322                  CLR     CALBCC          ;ZERO EXPECTED BCC
007564  012737  007572  001220          MOV     #2$,LOCK        ;SET FOR SW09=1
007572  104412                  2$:     MSTCLR                  ;INIT DQ11
007574  012737  000200  015326          MOV     #200,XPOLY      ;SEL "LRC 24"
007602  004737  011434                  JSR     PC,TXBCC        ;GOTO SUBROUTINE
007606  012705  000015                  MOV     #15,R5          ;SEL  BCC REG
007612  112777  000015  171550          MOVB    #15,@DQREG      ;" " "
007620  017701  171546                  MOV     @DQSEC,R1       ;READ BCC REG.
007624  023701  015322                  CMP     CALBCC,R1       ;IS BCC CORRECT?
007630  001401                          BEQ     .+4             ;BR IF GOOD.
007632  104002                          HLT     2               ;BCC ERROR.
007634  012705  000016                  MOV     #16,R5          ;SEL BCC REG.
```

```
2310  007640  013701  015340                  MOV     SEC16,R1        ;GET SAVED BCC
2311  007644  023701  015322                  CMP     CALBCC,R1       ;DID IT COMPARE?
2312  007650  001401                          BEQ     .+4             ;BR IF GOOD.
2313  007652  104002                          HLT     2               ;BCC ERROR
2314  007654  013704  015322                  MOV     CALBCC,R4       ;SAVE GOOD BCC
2315  007660  005037  015322                  CLR     CALBCC          ;ZERO SOFTWARE BCC
2316  007664  112777  000012  171476          MOVB    #MISC.,@DQREG   ;SEL MISC REGISTER
2317  007672  042777  000100  171472          BIC     #BIT6,@DQSEC    ;SEL POLY 0-15
2318  007700  012705  000015                  MOV     #15,R5          ;SET FOR ERROR
2319  007704  110577  171460                  MOVB    R5,@DQREG       ;SEL BCC REG
2320  007710  017701  171456                  MOV     @DQSEC,R1       ;READ BCC
2321  007714  005701                          TST     R1              ;IS IT 0
2322  007716  001401                          BEQ     .+4             ;BR IF YES.
2323  007720  104002                          HLT     2               ;BCC NOT 0
2324  007722  012705  000016                  MOV     #16,R5
2325  007726  013701  015342                  MOV     SEC16X,R1
2326  007732  005701                          TST     R1
2327  007734  001401                          BEQ     .+4
2328  007736  104002                          HLT     2
2329  007740  010437  015322                  MOV     R4,CALBCC
2330  007744  112777  000012  171416          MOVB    #MISC.,@DQREG
2331  007752  005277  171414                  INC     @DQSEC
2332  007756  005377  171410                  DEC     @DQSEC
2333  007762  012705  000016                  MOV     #16,R5
2334  007766  110577  171376                  MOVB    R5,@DQREG
2335  007772  017701  171374                  MOV     @DQSEC,R1
2336  007776  000301                          SWAB    R1
2337  010000  023701  015322                  CMP     CALBCC,R1
2338  010004  001401                          BEQ     .+4
2339  010006  104002                          HLT     2
2340  010010  012737  000007  015330          MOV     #7,COUNT
2341  010016  112777  000012  171344          MOVB    #MISC.,@DQREG
2342  010024  005277  171342          3$:     INC     @DQSEC
2343  010030  005377  171336                  DEC     @DQSEC
2344  010034  005337  015330                  DEC     COUNT
2345  010040  001371                          BNE     3$
2346  010042  012705  000015                  MOV     #15,R5
2347  010046  110577  171316                  MOVB    R5,@DQREG
2348  010052  017701  171314                  MOV     @DQSEC,R1
2349  010056  000301                          SWAB    R1
2350  010060  023701  015322                  CMP     CALBCC,R1
2351  010064  001401                          BEQ     .+4
2352  010066  104002                          HLT     2
2353  010070  104401                          SCOP1
2354  010072  105237  015322                  INCB    CALBCC
2355  010076  001235                          BNE     2$
2356  010100  104413                          MEMCLR
2357  010102  104413                          MEMCLR
2358  010104  104403                          SCOPE
2359                                          ;
2360                                          ;USERS BCC RECEIVER TRANSMITTER TEST
2361                                          ;THIS TEST ALLOWS THE USER TO
2362                                          ;USE ANY POSSIBLE POLYNOMIAL
2363                                          ;OF HIS CHOICE.
2364                                          ;THE TEST USES THAT POLYNOMIAL
2365                                          ;ON A FOUR HUNDRED, EIGHT BIT,BINARY COUNT
```

```
                                                        ;DATA PATTERN
                                                        ;NOTE: THE BCC OF THE TRANSMITTER AND RECEIVER
                                                        ;ARE NOT CHECKED FOR CORRECT CONTENT
                                                        ;ONLY THAT THE HARDWARE DID NOT
                                                        ;FLAG AN ERROR.

010106  012737  010106  001216  USEBCC: MOV     #USEBCC,NEXT
010114  012737  010106  001214          MOV     #USEBCC,RETURN
010122  104402                        TYPE
010124  016440                        MPOLY
010126  000000                        HALT
010130  104414                        CKSWR                           ;CHECK FOR <^G>
010132  017737  171042  015326        MOV     @SWR,XPOLY
010140  004737  010160          2$:   JSR     PC,$TBCC
010144  017705  171216                MOV     @DQE   ,R5
010150  005705                        TST     R5
010152  100001                        BPL     .+4
010154  104001                        HLT     1
010156  000770                        BR      2$
```

```
2388  010160  011637  015336          STBCC:  MOV     (SP),SAVEPC     ;SAVE PC OF ENTERING ROUTINE
2389  010164  104413                          MEMCLR
2390  010166  005000                          CLR     R0
2391  010170  012704  015354                  MOV     #TXBUFF,R4
2392  010174  110024                  1$:     MOVB    R0,(R4)+
2393  010176  105200                          INCB    R0
2394  010200  001375                          BNE     1$
2395  010202  105077  171162                  CLRB    @DQREG          ;SELECT THE RX BA PRI.
2396  010206  012777  015756  171156          MOV     #RXBUFF,@DQSEC  ;LOAD THE RX BA
2397  010214  112777  000121  171146          MOVB    #121,@DQREG     ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2398  010222  012777  177400  171142          MOV     #-400,@DQSEC    ;SET RX CC FOR A TRANSFER OF 400 CHARS.
2399  010230  112777  000022  171132          MOVB    #22,@DQREG      ;WRITE ENABLE, TX BA PRI.
2400  010236  012777  015352  171126          MOV     #SYNC,@DQSEC    ;LOAD THE TX BA PRI.
2401  010244  112777  000023  171116          MOVB    #23,@DQREG      ;ENTER T,WRITE ENABLE,TX CC PRI.
2402  010252  012777  177776  171112          MOV     #-2,@DQSEC      ;SET TX CC FOR A TRANSFER OF 2 CHARS.
2403  010260  112777  000024  171102          MOVB    #24,@DQREG      ;WRITE ENABLE,RX BA SEC.
2404  010266  005077  171100                  CLR     @DQSEC          ;CLEAR THE RX BA SEC
2405  010272  112777  000065  171070          MOVB    #65,@DQREG      ;EXIT T,WRITE ENABLE,RX CC SEC.
2406  010300  012777  177777  171064          MOV#-1,@DQSEC           ;SET THE RX CC SEC FOR ONE PAD CHAR.
2407  010306  112777  000026  171054          MOVB    #26,@DQREG      ;WRITE ENABLE,TX BA SEC.
2408  010314  012777  015354  171050          MOV     #TXBUFF,@DQSEC
2409  010322  112777  000127  171040          MOVB    #127,@DQREG     ;EXIT T,WRITE ENABLE,TX CC SEC.
2410  010330  012777  177400  171034          MOV     #-400,@DQSEC
2411  010336  112777  000011  171024          MOVB    #11,@DQREG
2412  010344  013777  015350  171020          MOV     .SYNC,@DQSEC
2413  010352  112777  000017  171010          MOVB    #17,@DQREG
2414  010360  013777  015326  171004          MOV     XPOLY,@DQSEC
2415  010366  112777  000012  170774          MOVB    #MISC.,@DQREG
2416  010374  012777  004010  170770          MOV     #4010,@DQSEC
2417  010402  005037  001244                  CLR     TEMP1
2418  010406  005037  001246                  CLR     TEMP2
2419  010412  012777  010462  170734          MOV     #TXISR1,@DQTVEC
2420  010420  005077  170732                  CLR     @DQTLVL
2421  010424  012777  000001  170726          MOV     #1,@DQRCSR
2422  010432  012777  000041  170724          MOV     #41,@DQTCSR
2423  010440  005037  177776                  CLR     PS
2424  010444  105237  001244          2$:     INCB    TEMP1
2425  010450  001375                          BNE     2$
2426  010452  105237  001246                  INCB    TEMP2
2427  010456  001372                          BNE     2$
2428  010460  104000                          HLT
2429  010462  022626                    4.7   TXISR1: CMP     (SP)+,(SP)+
2430  010464  042777  000040  170672    8.2           BIC     #BIT5,@DQTCSR
2431  010472  112777  000063  170670    7.6           MOVB    #63,@DQREG
2432  010500  005077  170666            6.1           CLR     @DQSEC
2433  010504  000240                    1.5           NOP
2434  010506  005037  001244            3.7           CLR     TEMP1
2435  010512  005037  001246            3.7           CLR     TEMP2
2436  010516  032777  000100  170634    7.7   1$:     BIT     #BIT6,@DQRCSR
2437  010524  001007                    2.6           BNE     ENDBCC
2438  010526  105237  001244            3.7           INCB    TEMP1
2439  010532  001371                    2.6           BNE     1$
2440  010534  105237  001246            3.7           INCB    TEMP2
2441  010540  001366                    2.6           BNE     1$
2442  010542  104000                    9.3           HLT                     ;RX SECONDARY DONE NOT SET.
2443  010544  000207                    3.5   ENDBCC: RTS     PC
```

```
 2444
 2445
 2446
 2447  010546  010537  015336          SYNBCC: MOV     R5,SAVEPC       ;SAVE PC OF ENTERING ROUTINE
 2448  010552  104412                          MSTCLR                  ;CLEAR THE DQ11
 2449  010554  105077  170610                  CLRB    @DQREG          ;SELECT THE RX BA PRI.
 2450  010560  012777  015756  170604          MOV     #RXBUFF,@DQSEC  ;LOAD THE RX BA
 2451  010566  112777  000121  170574          MOVB    #121,@DQREG     ;*ENTER T,WRITE ENABLE,RX CC,PRI.
 2452  010574  012577  170572                  MOV     (R5)+,@DQSEC    ;SET RX CC
 2453  010600  112777  000022  170562          MOVB    #22,@DQREG      ;WRITE ENABLE, TX BA PRI.
 2454  010606  012777  015354  170556          MOV     #TXBUFF,@DQSEC  ;LOAD THE TX BA PRI.
 2455  010614  112777  000123  170546          MOVB    #123,@DQREG     ;ENTER T,WRITE ENABLE,TX CC PRI.
 2456  010622  012577  170544                  MOV     (R5)+,@DQSEC    ;SET TX CC
 2457  010626  112777  000024  170534          MOVB    #24,@DQREG      ;WRITE ENABLE,RX BA SEC.
 2458  010634  012777  015756  170530          MOV     #RXBUFF,@DQSEC  ;LOAD THE PX BA SEC.
 2459  010642  112777  000065  170520          MOVB    #65,@DQREG      ;EXIT T,WRITE ENABLE,RX CC SEC.
 2460  010650  012777  177777  170514          MOV     #-1,@DQSEC      ;SET FOR ONE PAD CHARACTER.
 2461  010656  112777  000026  170504          MOVB    #26,@DQREG      ;WRITE ENABLE,TX BA SEC.
 2462  010664  005077  170502                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
 2463  010670  112777  000067  170472          MOVB    #67,@DQREG      ;EXIT T,WRITE ENABLE,TX CC SEC.
 2464  010676  005077  170470                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
 2465  010702  112777  000011  170460          MOVB    #11,@DQREG
 2466  010710  013777  015350  170454          MOV     .SYNC,@DQSEC
 2467  010716  112777  000017  170444          MOVB    #17,@DQREG
 2468  010724  013777  015326  170440          MOV     XPOLY,@DQSEC
 2469  010732  112777  000012  170430          MOVB    #MISC.,@DQREG
 2470  010740  012777  004010  170424          MOV     #4010,@DQSEC
 2471  010746  005037  001244                  CLR     TEMP1
 2472  010752  005037  001246                  CLR     TEMP2
 2473  010756  052777  000003  170374          BIS     #3,@DQRCSR
 2474  010764  052777  000002  170372          BIS     #2,@DQTCSR
 2475  010772  032777  020000  170372  2$:     BIT     #BIT13,@DQSEC
 2476  011000  001006                          BNE     3$
 2477  011002  005237  001244                  INC     TEMP1
 2478  011006  001371                          BNE     2$
 2479  011010  005237  001246                  INC     TEMP2
 2480  011014  001366                          BNE     2$
 2481  011016  012777  000001  170340  3$:     MOV     #BIT0,@DQTCSR
 2482  011024  005037  001244                  CLR     TEMP1
 2483  011030  012737  000005  001246          MOV     #5,TEMP2
 2484  011036  032777  000100  170314  4$:     BIT     #BIT6,@DQRCSR
 2485  011044  001007                          BNE     ENDSYN
 2486  011046  005237  001244                  INC     TEMP1
 2487  011052  001371                          BNE     4$
 2488  011054  005337  001246                  DEC     TEMP2
 2489  011060  001366                          BNE     4$
 2490  011062  104000                          HLT                     ;RECEIVER DONE SECONDARY NOT SET.
 2491  011064  000205          ENDSYN: RTS     R5
 2492
 2493
 2494
 2495  011066  011637  015336          TYBCC:  MOV     (SP),SAVEPC     ;SAVE PC OF ENTERING ROUTINE
 2496  011072  104412                          MSTCLR          ;CLEAR THE DQ11
 2497  011074  000240                          NOP
 2498  011076  104412                          MSTCLR
 2499  011100  012737  000351  015354          MOV     #351,TXBUFF
```

```
2500  011106  012737  000011  015330          MOV     #9.,COUNT
2501  011114  105077  170250          1$:     CLRB    @DQREG          ;SELECT THE RX BA PRI.
2502  011120  012777  015756  170244          MOV     #RXBUFF,@DQSEC  ;LOAD THE RX BA
2503  011126  112777  000121  170234          MOVB    #121,@DQREG     ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2504  011134  012777  177600  170230          MOV     #-200,@DQSEC    ;SET RX CC FOR A TRANSFER OF 1 CHARACTER.
2505  011142  112777  000022  170220          MOVB    #22,@DQREG      ;WRITE ENABLE, TX BA PRI.
2506  011150  012777  015354  170214          MOV     #TXBUFF,@DQSEC  ;LOAD THE TX BA PRI.
2507  011156  112777  000123  170204          MOVB    #123,@DQREG     ;ENTER T,WRITE ENABLE,TX CC PRI.
2508  011164  012777  177600  170200          MOV     #-200,@DQSEC    ;SET TX CC FOR A TRANSFER OF 1 CHARACTER.
2509  011172  112777  000024  170170          MOVB    #24,@DQREG      ;WRITE ENABLE,RX BA SEC.
2510  011200  005077  170166                  CLR     @DQSEC          ;CLEAR THE RX BA SEC
2511  011204  112777  000065  170156          MOVB    #65,@DQREG      ;EXIT T,WRITE ENABLE,RX CC SEC.
2512  011212  005077  170154                  CLR     @DQSEC          ;CLEAR THE RX CC SEC.
2513  011216  112777  000026  170144          MOVB    #26,@DQREG      ;WRITE ENABLE,TX BA SEC.
2514  011224  005077  170142                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
2515  011230  112777  000067  170132          MOVB    #67,@DQREG      ;EXIT T,WRITE ENABLE,TX CC SEC.
2516  011236  005077  170130                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
2517  011242  112777  000017  170120          MOVB    #17,@DQREG
2518  011250  013777  015326  170114          MOV     XPOLY,@DQSEC
2519  011256  112777  000012  170104          MOVB    #MISC.,@DQREG
2520  011264  012777  004012  170100          MOV     #4012,@DQSEC
2521  011272  052777  000001  170064          BIS     #BIT0,@DQTCSR           ;SET TRANSMITTER GO
2522  011300  027777  170060  170056          CMP     @DQTCSR,@DQTCSR ;WAIST TIME.
2523  011306  027777  170052  170050          CMP     @DQTCSR,@DQTCSR ;WAIST TIME
2524  011314  027777  170044  170042          CMP     @DQTCSR,@DQTCSR ;WAIST TIME
2525  011322  005277  170044                  INC     @DQSEC          ;PRIM THE
2526  011326  005377  170040                  DEC     @DQSEC                  ;  TRANSMITTER.
2527  011332  042777  000200  170032          BIC     #BIT7,@DQSEC    ;CLEAR THE BIT WINDOW.
2528  011340  052777  010001  170012          BIS     #10001,@DQRCSR
2529  011346  005277  170020          2$:     INC     @DQSEC
2530  011352  005377  170014                  DEC     @DQSEC
2531  011356  005337  015330                  DEC     COUNT
2532  011362  001371                          BNE     2$
2533  011364  112777  000016  167776          MOVB    #16,@DQREG
2534  011372  017737  167774  015340          MOV     @DQSEC,SEC16
2535  011400  112777  000012  167762          MOVB    #MISC.,@DQREG
2536  011406  012737  000007  015330          MOV     #7.,COUNT
2537  011414  005277  167752          3$:     INC     @DQSEC
2538  011420  005377  167746                  DEC     @DQSEC
2539  011424  005337  015330                  DEC     COUNT
2540  011430  001371                          BNE     3$
2541  011432  000207                  ENDTY:  RTS     PC
2542
2543
2544
2545
2546
2547  011434  011637  015336          TXBCC:  MOV     (SP),SAVEPC     ;SAVE PC OF ENTERING ROUTINE
2548  011440  104412                          MSTCLR                  ;CLEAR THE DQ11
2549  011442  013737  015322  015354          MOV     CALBCC,TXBUFF
2550  011450  012737  000011  015330          MOV     #9.,COUNT
2551  011456  105077  167706          1$:     CLRB    @DQREG          ;SELECT THE RX BA PRI.
2552  011462  012777  015756  167702          MOV     #RXBUFF,@DQSEC  ;LOAD THE RX BA
2553  011470  112777  000121  167672          MOVB    #121,@DQREG     ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2554  011476  012777  177777  167666          MOV     #-1,@DQSEC      ;SET RX CC FOR A TRANSFER OF 1 CHARACTER.
2555  011504  112777  000022  167656          MOVB    #22,@DQREG      ;WRITE ENABLE, TX BA PRI.
```

```
2556  011512  012777  015354  167652          MOV     #TXBUFF,@DQSEC  ;LOAD THE TX BA PRI.
2557  011520  112777  000123  167642          MOVB    #123,@DQREG     ;ENTER T,WRITE ENABLE,TX CC PRI.
2558  011526  012777  177777  167636          MOV     #-1,@DQSEC      ;SET TX CC FOR A TRANSFER OF 1 CHARACTER.
2559  011534  112777  000024  167626          MOVB    #24,@DQREG      ;WRITE ENABLE,RX BA SEC.
2560  011542  005077  167624                  CLR     @DQSEC          ;CLEAR THE RX BA SEC
2561  011546  112777  000065  167614          MOVB    #65,@DQREG      ;EXIT T,WRITE ENABLE,RX CC SEC.
2562  011554  005077  167612                  CLR     @DQSEC          ;CLEAR THE RX CC SEC.
2563  011560  112777  000026  167602          MOVB    #26,@DQREG      ,WRITE ENABLE,TX BA SEC.
2564  011566  005077  167600                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
2565  011572  112777  000067  167570          MOVB    #67,@DQREG      ;EXIT T,WRITE ENABLE,TX CC SEC.
2566  011600  005077  167566                  CLR     @DQSEC          ;CLEAR THE TX CC SEC
2567  011604  112777  000012  167556          MOVB    #MISC.,@DQREG
2568  011612  012777  004112  167552          MOV     #4112,@DQSEC
2569  011620  112777  000017  167542          MOVB    #17,@DQREG
2570  011626  013777  015326  167536          MOV     XPOLY,@DQSEC
2571  011634  112777  000012  167526          MOVB    #MISC.,@DQREG
2572  011642  052777  000001  167514          BIS     #BIT0,@DQTCSR           ;SET TRANSMITTER GO
2573  011650  027777  167510  167506          CMP     @DQTCSR,@DQTCSR ;WAIST TIME.
2574  011656  027777  167502  167500          CMP     @DQTCSR,@DQTCSR ;WAIST TIME
2575  011664  027777  167474  167472          CMP     @DQTCSR,@DQTCSR ;WAIST TIME
2576  011672  005277  167474                  INC     @DQSEC          ;PRIM THE
2577  011676  005377  167470                  DEC     @DQSEC                  ;  TRANSMITTER.
2578  011702  042777  000200  167462          BIC     #BIT7,@DQSEC    ;CLEAR THE BIT WINDOW.
2579  011710  052777  010001  167442          BIS     #10001,@DQRCSR
2580  011716  005277  167450          2$:     INC     @DQSEC
2581  011722  005377  167444                  DEC     @DQSEC
2582  011726  005337  015330                  DEC     COUNT
2583  011732  001371                          BNE     2$
2584  011734  042777  000100  167430          BIC     #BIT6,@DQSEC
2585  011742  112777  000016  167420          MOVB    #16,@DQREG
2586  011750  017737  167416  015342          MOV     @DQSEC,SEC16X
2587  011756  112777  000012  167404          MOVB    #MISC.,@DQREG
2588  011764  052777  000100  167400          BIS     #BIT6,@DQSEC
2589  011772  112777  000016  167370          MOVB    #16,@DQREG
2590  012000  017737  167366  015340          MOV     @DQSEC,SEC16
2591  012006  112777  000012  167354          MOVB    #MISC.,@DQREG
2592  012014  012737  000007  015330          MOV     #7.,COUNT
2593  012022  005277  167344          3$:     INC     @DQSEC
2594  012026  005377  167340                  DEC     @DQSEC
2595  012032  005337  015330                  DEC     COUNT
2596  012036  001371                          BNE     3$
2597  012040  000207                  ENDTX:  RTS     PC
```

```
2598                                                    ;
2599                                                    ;BLOCK CHECK CHARACTER GENERATOR SIMULATOR.
2600                                                    ;
2601  012042  012537  001244          SIMBCC: MOV     (R5)+,TEMP1
2602  012046  012537  001246                  MOV     (R5)+,TEMP2
2603  012052  012537  001250                  MOV     (R5)+,TEMP3
2604  012056  005037  015324          1$:     CLR     BCCFBK
2605  012062  013700  001250                  MOV     TEMP3,R0
2606  012066  006037  001246                  ROR     TEMP2
2607  012072  005500                          ADC     R0
2608  012074  032700  000001                  BIT     #BIT0,R0
2609  012100  001402                          BEQ     2$
2610  012102  005137  015324                  COM     BCCFBK
2611  012106  013700  015326          2$:     MOV     XPOLY,R0
2612  012112  005100                          COM     R0
2613  012114  040037  015324                  BIC     R0,BCCFBK
2614  012120  000241                          CLC
2615  012122  006037  001250                  ROR     TEMP3
2616  012126  013700  015324                  MOV     BCCFBK,R0
2617  012132  013701  001250                  MOV     TEMP3,R1
2618  012136  010102                          MOV     R1,R2
2619  012140  040100                          BIC     R1,R0
2620  012142  043702  015324                  BIC     BCCFBK,R2
2621  012146  050200                          BIS     R2,R0
2622  012150  043737  015326  001250          BIC     XPOLY,TEMP3
2623  012156  050037  001250                  BIS     R0,TEMP3
2624  012162  005337  001244                  DEC     TEMP1
2625  012166  001333                          BNE     1$
2626  012170  013737  001250  015322          MOV     TEMP3,CALBCC
2627  012176  000205                          RTS     R5
2628
2629  012200                          .MEMCLR:
2630  012200  005077  167154                  CLR     @DQRCSR
2631  012204  005077  167154                  CLR     @DQTCSR
2632  012210  005077  167152                  CLR     @DQERR
2633  012214  012705  000020                  MOV     #16.,R5
2634  012220  152777  000020  167142  1$:     BISB    #BIT4,@DQREG
2635  012226  142777  000140  167134          BICB    #140,@DQREG
2636  012234  005077  167132                  CLR     @DQSEC
2637  012240  105277  167124                  INCB    @DQREG
2638  012244  005305                          DEC     R5
2639  012246  001364                          BNE     1$
2640  012250  105077  167114                  CLRB    @DQREG
2641  012254  105077  167102                  CLRB    @DQRCSH
2642  012260  012705  000020                  MOV     #16.,R5
2643  012264  112777  000010  167076  2$:     MOVB    #10,@DQREG
2644  012272  005077  167074                  CLR     @DQSEC
2645  012276  112777  000014  167064          MOVB    #14,@DQREG
2646  012304  005077  167062                  CLR     @DQSEC
2647  012310  105277  167046                  INCB    @DQRCSH
2648  012314  005305                          DEC     R5
2649  012316  001362                          BNE     2$
2650  012320  105077  167036                  CLRB    @DQRCSH
2651  012324                          .MSTCLR:
2652  012324  112777  000012  167036          MOVB    #MISC.,@DQREG
2653  012332  012777  000040  167032          MOV     #BIT5,@DQSEC
```

```
 2654  012340  000002                          RTI
 2655
 2656
 2657
 2658
 2659                                          ;END OF PASS
 2660                                          ;TYPE NAME OF TEST
 2661                                          ;UPDATE PASS COUNT
 2662                                          ;CHECK FOR EXIT TO ACT-11
 2663                                          ;RESTART TEST
 2664
 2665  012342  005037  001234          .EOP:  CLR     LSTERR                 ;CLEAR LAST ERROR PC
 2666  012346  005037  001312                 CLR     ERRFLG                 ;CLEAR ERROR FLAG
 2667  012352  005237  001230                 INC     PASCNT                 ;UPDATE PASS COUNT
 2668  012356  104402                         TYPE
 2669  012360  014572                         MEPASS
 2670  012362  104402                         TYPE
 2671  012364  014753                         MCSRX
 2672  012366  104411                         CNVRT
 2673  012370  012500                         XCSR
 2674  012372  104402                         TYPE
 2675  012374  014761                         MVECX
 2676  012376  104411                         CNVRT
 2677  012400  012506                         XVEC
 2678  012402  104402                         TYPE
 2679  012404  014767                         MPASSX
 2680  012406  104411                         CNVRT
 2681  012410  012514                         XPASS
 2682  012412  104402                         TYPE
 2683  012414  015000                         MERRX
 2684  012416  104411                         CNVRT
 2685  012420  012522                         XERR
 2686  012422  013777  001230  166552        MOV     PASCNT,@LIGHTS         ;DISPLAY PASS COUNT
 2687  012430  005337  001276                 DEC     SAVNUM
 2688  012434  001013                         BNE     RESTRT
 2689  012436  013737  001504  001276        MOV     DQNUM,SAVNUM
 2690  012444  013701  000042                 MOV     @#42,R1                ;CHECK FOR ACT-11 OR DDP
 2691  012450  001405                         BEQ     RESTRT                 ;IF NOT, CONTINUE TESTING
 2692  012452  000005                         RESET
 2693  012454                         LOGICAL:
 2694  012454  004711                         JSR     PC,(R1)
 2695  012456  000240                         NOP
 2696  012460  000240                         NOP
 2697  012462  000240                         NOP
 2698  012464  104414                 RESTRT: CKSWR
 2699  012466  012737  002254  001214        MOV     #TST1,RETURN
 2700  012474  000137  002254                 JMP     TST1
 2701  012500  000001                 XCSR:   1
 2702  012502     006     002                 .BYTE   6,2
 2703  012504  001360                         DQRCSR
 2704  012506  000001                 XVEC:   1
 2705  012510     003     002                 .BYTE   3,2
 2706  012512  001350                         DQRVEC
 2707  012514  000001                 XPASS:  1
 2708  012516     006     002                 .BYTE   6,2
 2709  012520  001230                         PASCNT
```

```
012522 000001                         $ERR:   1
012524    006    002                          .BYTE   6,2
012526 001232                                 ERRCNT

                                              ;SCOPE LOOP AND INTERATION HANDLER

012530 104414                         .SCOPE: CKSWR
012532 032777 040000 166440                   BIT     #BIT14,@SWR
012540 001407                         TTST:   BEQ     1$
012542 000432                                 BR      3$
012544 105777 166434                          TSTB    @TKCSR
012550 100027                                 BPL     3$
012552 017700 166430                          MOV     @TKDBR,R0
012556 000412                                 BR      2$
012560 032777 004000 166412           1$:     BIT     #SW11,@SWR
012566 001006                                 BNE     2$
012570 005237 001224                          INC     LPCNT
012574 023737 001224 001222                   CMP     LPCNT,ICOUNT
012602 001012                                 BNE     3$
012604 105037 001312                  2$:     CLRB    ERRFLG
012610 005037 001224                          CLR     LPCNT
012614 012737 000011 001222                   MOV     #9.,ICOUNT
012622 013737 00121[illegible] 001214         MOV     NEXT,RETURN
012630 013716 001214                  3$:     MOV     RETURN,(SP)
012634 000002                                 RTI
012636 001407                         BRW:    1407
012640 000432                         BRX:    432

                                              ;CHECK FOR FREEZE ON CURRENT DATA

012642 104414                         .SCOP1: CKSWR
012644 032777 001000 166326                   BIT     #SW09,@SWR
012652 001402                                 BEQ     1$
012654 013716 001220                          MOV     LOCK,(SP)
012660 000002                         1$:     RTI

                                              ;TELETYPE OUTPUT ROUTINE

012662 010546                         .TYPE:  MOV     R5,-(SP)
012664 017605 000002                          MOV     @2(SP),R5
012670 062766 000002 000002                   ADD     #2,2(SP)
012676 005737 014352                  1$:     TST     @#RDSW
012702 001004                                 BNE     300$
012704 032777 010000 166266                   BIT     #SW12,@SWR
012712 001024                                 BNE     3$
012714 105715                         300$:   TSTB    (R5)
012716 100014                                 BPL     2$
012720 105777 166264                          TSTB    @TPCSR
012724 100375                                 BPL     .-4
012726 012777 000015 166256                   MOV     #15,@TPDBR
012734 105777 166250                          TSTB    @TPCSR
012740 100375                                 BPL     .-4
012742 012777 000012 166242                   MOV     #12,@TPDBR
012750 105777 166234                  2$:     TSTB    @TPCSR
012754 100375                                 BPL     2$
012756 112577 166230                          MOVB    (R5)+,@TPDBR
```

```
012762  001345                          BNE     1$
012764  012605                  3$:     MOV     (SP)+,R5
012766  000002                          RTI

                                        ;ASCII STRING INPUT ROUTINE

012770  010346                  .INSTR: MOV     R3,-(SP)
012772  010446                          MOV     R4,-(SP)
012774  017637  000004  013012          MOV     @4(SP),.MSG
013002  062766  000002  000004          ADD     #2,4(SP)
013010  104402                  .INST1: TYPE
013012  000000                  .MSG:   0
013014  012704  015144                  MOV     #INBUF,R4
013020  012703  000007                  MOV     #7,R3
013024  105777  166154          1$:     TSTB    @TKCSR
013030  100375                          BPL     1$
013032  117714  166150                  MOVB    @TKDBR,(R4)
013036  142714  000200                  BICB    #200,(R4)
013042  121427  000025                  CMPB    (R4),#25                ;IS IT ^U
013046  001003                          BNE     200$
013050  104402  014532                  TYPE,MCRLF
013054  000755                          BR      .INST1
013056  122427  000015          200$:   CMPB    (R4)+,#15
013062  001423                          BEQ     INSTR2
013064  117777  166116  166120          MOVB    @TKDBR,@TPDBR
013072  105777  166112          2$:     TSTB    @TPCSR
013076  100375                          BPL     2$
013100  005303                          DEC     R3
013102  001350                          BNE     1$
013104  000402                          BR      .INSTG
013106  010346                  .INSTE: MOV     R3,-(SP)
013110  010446                          MOV     R4,-(SP)
013112  104402                  .INSTG: TYPE
013114  014526                          MQM
013116  005737  014352                  TST     @#RDSW
013122  001402                          BEQ     400$
013124  104402  014532                  TYPE,MCRLF
013130  000727                  400$:   BR      .INST1
013132  012604                  INSTR2: MOV     (SP)+,R4
013134  012603                          MOV     (SP)+,R3
013136  000002                          RTI

                                        ;CONVERT ASCII STRING TO OCTAL

013140  010546                  .PARAM: MOV     R5,-(SP)
013142  010446                          MOV     R4,-(SP)
013144  016605  000004                  MOV     4(SP),R5
013150  012537  013344                  MOV     (R5)+,LOLIM
013154  012537  013346                  MOV     (R5)+,HILIM
013160  012537  013350                  MOV     (R5)+,DEVADR
013164  112537  013352                  MOVB    (R5)+,LOBITS
013170  112537  013353                  MOVB    (R5)+,ADRCNT
013174  010566  000004                  MOV     R5,4(SP)
013200  005005                  PARAM1: CLR     R5
013202  012704  015144                  MOV     #INBUF,R4
013206  122714  000015                  CMPB    #15,(R4)
```

```
 2822  013212  001420                        BEQ     PARERR
 2823  013214  121427  000060        1$:     CMPB    (R4),#60
 2824  013220  002415                        BLT     PARERR
 2825  013222  121427  000067                CMPB    (R4),#67
 2826  013226  003012                        BGT     PARERR
 2827  013230  142714  000060                BICB    #60,(R4)
 2828  013234  152405                        BISB    (R4)+,R5
 2829  013236  122714  000015                CMPB    #15,(R4
 2830  013242  001414                        BEQ     LIMITS
 2831  013244  006305                        ASL     R5
 2832  013246  006305                        ASL     R5
 2833  013250  006305                        ASL     R5
 2834  013252  000760                        BR      1$
 2835  013254  122714  000015        PARERR: CMPB    #15,(R4)                ;IS FIRST CHARACTER A <CR>
 2836  013260  001003                        BNE     120$
 2837  013262  005737  014352                TST     @#RCSW                  ;IS CKSWR ROUTINE BEING USED
 2838  013266  001023                        BNE     PARTI
 2839  013270  104404                120$:   INSTER
 2840  013272  000742                        BR      PARAM1
 2841
 2842                                        ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
 2843
 2844  013274  020537  013346        LIMITS: CMP     R5,HILIM
 2845  013300  101365                        BHI     PARERR
 2846  013302  020537  013344                CMP     R5,LOLIM
 2847  013306  103762                        BLO     PARERR
 2848  013310  133705  013352                BITB    LOBITS,R5
 2849  013314  001357                        BNE     PARERR
 2850
 2851                                        ;STORE NUMBER AT SPECIFIED ADDRESS
 2852
 2853  013316  013704  013350                MOV     DEVADR,R4
 2854  013322  010524                1$:     MOV     R5,(R4)+
 2855  013324  062705  000002                ADD     #2,R5
 2856  013330  105337  013353                DECB    ADRCNT
 2857  013334  001372                        BNE     1$
 2858  013336  012604                PARTI:  MOV     (SP)+,R4
 2859  013340  012605                        MOV     (SP)+,R5
 2860  013342  000002                        RTI
 2861  013344  000000                LOLIM:  0
 2862  013346  000000                HILIM:  0
 2863  013350  000000                DEVADR: 0
 2864  013352  000000                LOBITS: 0
 2865          013353                ADRCNT=LOBITS+1
 2866
 2867                                        ;SAVE PC OF TEST THAT FAILED AND R0-R5
 2868
 2869  013354  016637  000004  001274 .SAV05: MOV     4(SP),SAVPC
 2870
 2871                                        ;SAVE R0-R5
 2872
 2873  013362  010537  001270        SV05:   MOV     R5,SAVR5
 2874  013366  010437  001266                MOV     R4,SAVR4
 2875  013372  010337  001264                MOV     R3,SAVR3
 2876  013376  010237  001262                MOV     R2,SAVR2
 2877  013402  010137  001260                MOV     R1,SAVR1
```

```
 2878  013406  010037  001256                  MOV     R0,SAVR0
 2879  013412  000002                          RTI
 2880
 2881                                          ;RESTORE R0-R5
 2882
 2883  013414  013700  001256          .RES05: MOV     SAVR0,R0
 2884  013420  013701  001260                  MOV     SAVR1,R1
 2885  013424  013702  001262                  MOV     SAVR2,R2
 2886  013430  013703  001264                  MOV     SAVR3,R3
 2887  013434  013704  001266                  MOV     SAVR4,R4
 2888  013440  013705  001270                  MOV     SAVR5,R5
 2889  013444  000002                          RTI
 2890
 2891                                          ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
 2892
 2893  013446  104402                  .CONVR: TYPE
 2894  013450  014532                          MCRLF
 2895  013452  010046                  .CNVRT: MOV     R0,-(SP)
 2896  013454  010146                          MOV     R1,-(SP)
 2897  013456  010346                          MOV     R3,-(SP)
 2898  013460  010446                          MOV     R4,-(SP)
 2899  013462  010546                          MOV     R5,-(SP)
 2900  013464  017601  000012                  MOV     @12(SP),R1
 2901  013470  013737  015206  001250          MOV     TEMP,TEMP3
 2902  013476  062766  000002  000012          ADD     #2,12(SP)
 2903  013504  012137  013666                  MOV     (R1)+,WRDCNT
 2904  013510  112137  013670          1$:     MOVB    (R1)+,CHRCNT
 2905  013514  112137  013671                  MOVB    (R1)+,SPACNT
 2906  013520  013137  013672                  MOV     @(R1)+,BINWRD
 2907  013524  013704  013672          2$:     MOV     BINWRD,R4
 2908  013530  113705  013670                  MOVB    CHRCNT,R5
 2909  013534  012700  015206                  MOV     #TEMP,R0
 2910  013540  010403                  3$:     MOV     R4,R3
 2911  013542  042703  177770                  BIC     #177770,R3
 2912  013546  062703  000060                  ADD     #060,R3
 2913  013552  110320                          MOVB    R3,(R0)+
 2914  013554  000241                          CLC
 2915  013556  006004                          ROR     R4
 2916  013560  000241                          CLC
 2917  013562  006004                          ROR     R4
 2918  013564  000241                          CLC
 2919  013566  006004                          ROR     R4
 2920  013570  005305                          DEC     R5
 2921  013572  001362                          BNE     3$
 2922  013574  012703  015250                  MOV     #MDATA,R3
 2923  013600  114023                  4$:     MOVB    -(R0),(R3)+
 2924  013602  105337  013670                  DECB    CHRCNT
 2925  013606  001374                          BNE     4$
 2926  013610  105737  013671                  TSTB    SPACNT
 2927  013614  001405                          BEQ     6$
 2928  013616  112723  000040          5$:     MOVB    #040,(R3)+
 2929  013622  105337  013671                  DECB    SPACNT
 2930  013626  001373                          BNE     5$
 2931  013630  105013                  6$:     CLRB    (R3)
 2932  013632  104402                          TYPE
 2933  013634  015250                          MDATA
```

```
2934  013636  005337  013666                  DEC     WRDCNT
2935  013642  001322                          BNE     1$
2936  013644  013737  001250  015206          MOV     TEMP3,TEMP
2937  013652  012605                          MOV     (SP)+,R5
2938  013654  012604                          MOV     (SP)+,R4
2939  013656  012603                          MOV     (SP)+,R3
2940  013660  012601                          MOV     (SP)+,R1
2941  013662  012600                          MOV     (SP)+,R0
2942  013664  000002                          RTI
2943  013666  000000              WRDCNT: 0
2944  013670  000000              CHRCNT: 0
2945          013671              SPACNT=CHRCNT+1
2946  013672  000000              BINWRD: 0
2947                                      ;TRAP DISPATCH SERVICE
2948                                      ;ARGUMENT OF TRAP IS EXTRACTED
2949                                      ;AND USED AS OFFSET TO OBTAIN POINTER
2950                                      ;TO SELECTED SUBROUTINE
2951
2952  013674  011646              .TRPSR: MOV     (SP),-(SP)          ;GET PC OF RETURN
2953  013676  162716  000002              SUB     #2,(SP)             ;=PC OF TRAP
2954  013702  017616  000000              MOV     @(SP),(SP)          ;GET TRP
2955  013706  006316              TRPOK:  ASL     (SP)                ;MULTIPLY TRAP ARG BY 2
2956  013710  042716  177001              BIC     #177000,(SP)        ;CLEAR UNWANTED BITS
2957  013714  062716  001314              ADD     #.TRPTAB,(SP)       ;POINTER TO SUBROUTINE ADDRESS
2958  013720  017616  000000              MOV     @(SP),(SP)          ;SUBROUTINE ADDRESS
2959  013724  000136                      JMP     @(SP)+              ;GO TO SUBROUTINE
2960
2961                                      ;ERROR HANDLER
2962
2963  013726  104414              .HLT:   CKSWR
2964  013730  032777  010000  165242      BIT     #SW12,@SWR
2965  013736  001406                      BEQ     XBX
2966  013740  105777  165244              TSTB    @TPCSR
2967  013744  100003                      BPL     XBX
2968  013746  112777  000207  165236      MOVB    #207,@TPDBR
2969  013754  032777  020000  165216  XBX:BIT     #SW13,@SWR
2970  013762  001074                      BNE     HALTS
2971  013764  021637  001234              CMP     (SP),LSTERR
2972  013770  001404                      BEQ     1$
2973  013772  011637  001234              MOV     (SP),LSTERR
2974  013776  105037  001312              CLRB    ERRFLG
2975  014002  004406              1$:     SAVOS
2976  014004  011605                      MOV     (SP),R5
2977  014006  162705  000002              SUB     #2,R5
2978  014012  011504                      MOV     (R5),R4
2979  014014  006304                      ASL     R4
2980  014016  061504                      ADD     (R5),R4
2981  014020  006304                      ASL     R4
2982  014022  042704  177001              BIC     #177001,R4
2983  014026  062704  016360              ADD     #.ERRTAB,R4
2984  014032  012437  014124              MOV     (R4)+,ERRMSG
2985  014036  012437  014136              MOV     (R4)+,DATAHD
2986  014042  011437  014150              MOV     (R4),DATABP
2987  014046  105737  001312              TSTB    ERRFLG
2988  014052  001403                      BEQ     TYPMSG
2989  014054  005737  014150              TST     DATABP
```

```
 2990  014060  001027                          BNE     TYPDAT
 2991  014062  104402                TYPMSG:   TYPE
 2992  014064  015011                          MTSTN
 2993  014066  104411                          CNVRT
 2994  014070  014250                          XTSTN
 2995  014072  104402                          TYPE
 2996  014074  015077                          MERRPC
 2997  014076  104411                          CNVRT
 2998  014100  014242                          ERTABO
 2999  014102  104402                          TYPE
 3000  014104  014532                          MCRLF
 3001  014106  112737  177777  001312          MOVB    #-1,ERRFLG
 3002  014114  005737  014124                  TST     ERRMSG
 3003  014120  001402                          BEQ     WRKO.FM
 3004  014122  104402                          TYPE
 3005  014124  000000                ERRMSG:   0
 3006  014126                        WRKO.FM:
 3007  014126  005737  014136                  TST     DATAHD
 3008  014132  001402                          BEQ     TYPDAT
 3009  014134  104402                          TYPE
 3010  014136  000000                DATAHD:   0
 3011  014140  005737  014150        TYPDAT:   TST     DATABP
 3012  014144  001402                          BEQ     RESREG
 3013  014146  104410                          CONVRT
 3014  014150  000000                DATABP:   0
 3015  014152  104407                RESREG:   RESO5
 3016  014154  005777  165020        HALTS:    TST     @SWR
 3017  014160  100005                          BPL     EXITER
 3018  014162  010046                          PUSHRO
 3019  014164  016600  000002                  MOV     2(SP),RO
 3020  014170  000000                          HALT
 3021  014172  012600                          POPRO
 3022  014174  104414                EXITER:   CKSWR
 3023  014176  005237  001232                  INC     ERRCNT
 3024  014202  032777  000400  164770          BIT     #SW09,@SWR
 3025  014210  001007                          BNE     1$
 3026  014212  032777  002000  164760          BIT     #SW10,@SWR
 3027  014220  001407                          BEQ     2$
 3028  014222  013737  001216  001214          MOV     NEXT,RETURN
 3029  014230  012706  001200        1$:       MOV     #STACK,SP
 3030  014234  000177  164754                  JMP     @RETURN
 3031  014240  000002                2$:       RTI
 3032  014242  000001                ERTABO:   1
 3033  014244     006     002                  .BYTE   6,2
 3034  014246  001274                          SAVPC
 3035  014250  000001                XTSTN:    1
 3036  014252     003     002                  .BYTE   3,2
 3037  014254  001226                          TSTNC
 3038                                          :ENTER HERE ON POWER FAILURE
 3039
 3040
 3041  014256                        .PFAIL:
 3042  014256  012737  014270  000024          MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
 3043  014264  000000                          HALT                            ;HALT ON POWER DOWN NORMAL
 3044  014266  000777                          BR      .
 3045
```

```
                                                ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
                                        RESTAR:
014270  012737  014256  000024                    MOV     #.PFAIL,24              ;SET UP FOR POWER FAILURE
014276  012706  001200                            MOV     #STACK,SP
014302  005037  015206                            CLR     TEMP
014306  005237  015206                            INC     TEMP
014312  001375                                    BNE     .-4
014314  104402                                    TYPE
014316  014534                                    MPFAIL
014320  104411                                    CNVRT
014322  014344                                    PFTAB
014324  005037  001312                            CLR     ERRFLG
014330  005037  001234                            CLR     LSTERR
014334  104412                                    MSTCLR
014336  104413                                    MEMCLR
014340  000177  164650                            JMP     @RETURN
014344  000001                          PFTAB:    1
014346     003     002                  .BYTE     3,2
014350  001226                                    TSTNO


                                                ;CHECK SWITCH REGISTER ROUTINE. CHECKS FOR ↑G TO ALLOW CHANGING
                                                ;OF LOC.176.
                                                ;LOCATIONS USED:
014352  000000                                    RDSW:   .WORD   0


014354  005737  000042                  .CKSWR:   TST     @#42
014360  001042                                    BNE     OUT
014362  022737  000176  001200                    CMP     #SWREG,SWR              ;SOFTWARE SWITCH REGISTER PRESENT
014370  001036                                    BNE     OUT                     ;NO, GET OUT
014372  105777  164606                            TSTB    @TKCSR                  ;YES,WAIT FOR
014376  100033                                    BPL     OUT                     ;READY, GET CHARACTER
014400  017737  164602  013012                    MOV     @TKDBR,.MSG             ;AND STRIP OFF
014406  042737  177600  013012                    BIC     #177600,.MSG            ;THE GARBAGE
014414  122737  000007  013012                    CMPB    #7,.MSG                 ;IS IT A <↑G>
014422  001021                                    BNE     OUT
014424  104402  014502                            TYPE,$CNTG
014430  005137  014352                  .CNTLU:   COM     @#RDSW
014434  104402  014506                            TYPE,$MSWR
014440  104411  014474                            CNVRT,SWREGC
014444  104403  014515                            INSTR,$MNEW
014450  104405                                    PARAM
014452  000000                                    0
014454  177777                                    177777
014456  000176                                    SWREG
014460     000     001                  .BYTE             0,1
014462  104402  014532                            TYPE,MCRLF
014466  005037  014352                  OUT:      CLR     @#RDSW
014472  000002                                    RTI
014474  000001                          SWREGC:   1
014476     006     002                  .BYTE     6,2
014500  000176                                    SWREG
014502  057377  000107                  $CNTG:    .ASCIZ  <377>/↑G/
014506  051777  051127  020075          $MSWR:    .ASCIZ  <377>/SWR= /
```

```
3102  014514     000
3103  014515     040  047040  053505  $MNEW:  .ASCIZ  /  NEW= /
3104  014522  020075     000
3105          014526                  .EVEN
3106  014526  020040  000077          MQM:    .ASCIZ  /  ?/
3107  014532  000377                  MCRLF:  .ASCIZ  <377>
3108  014534  050377  051127  043040  MPFAIL: .ASCIZ  <377>/PWR FAILED. RESTART AT TEST /
3109  014542  044501  042514  027104
3110  014550  051040  051505  040524
3111  014556  052122  040440  020124
3112  014564  042524  052123  000040
3113  014572  042777  042116  050040  MEPASS: .ASCIZ  <377>/END PASS DZDQE  /
3114  014600  051501  020123  055104
3115  014606  050504  020105  000040
3116  014614  051377     000          MR:     .ASCIZ  <377>/R/
3117  014617     377  051120  043517  MERR2:  .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
3118  014624  040522  020115  047111
3119  014632  044504  040503  042524
3120  014640  020123  047516  042040
3121  014646  053105  041511  051505
3122  014654  050040  042522  042523
3123  014662  052116  000056
3124  014666  044777  051516  043125  MERR3:  .ASCIZ  <377>/INSUFFICIENT DATA!/
3125  014674  044506  044503  047105
3126  014702  020124  040504  040524
3127  014710  000041
3128  014712  052377  051505  020124  MTSTPC: .ASCIZ  <377>/TEST PC-/
3129  014720  041520  000055
3130  014724  046377  041517  020113  MLOCK:  .ASCIZ  <377>/LOCK ON SELECTED TEST/
3131  014732  047117  051440  046105
3132  014740  041505  042524  020104
3133  014746  042524  052123     000
3134  014753     103  051123  020072  MCSRX:  .ASCIZ   /CSR: /
3135  014760     000
3136  014761     125  041505  020072  MVECX:  .ASCIZ  /VEC: /
3137  014766     000
3138  014767     120  051501  042523  MPASSX: .ASCIZ  /PASSES: /
3139  014774  035123  000040
3140  015000  051105  047522  051522  MERRX:  .ASCIZ  /ERRORS: /
3141  015006  020072     000
3142  015011     377  052377  051505  MTSTN:  .ASCIZ  <377><377> /TEST NO: /
3143  015016  020124  047516  020072
3144  015024     000
3145  015025     377  042523  020124  MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DQ11'S DESIRED ACTIVE /
3146  015032  053523  052111  044103
3147  015040  051040  043505  052040
3148  015046  020117  050504  030461
3149  015054  051447  042040  051505
3150  015062  051111  042105  040440
3151  015070  052103  053111  027105
3152  015076     000
3153  015077     120  035103  000040  MERRPC: .ASCIZ  /PC: /
3154  015104  046777  050101  047440  XHEAD:  .ASCIZ  <377>/MAP OF DQ11 STATUS/<377>
3155  015112  020106  050504  030461
3156  015120  051440  040524  052524
3157  015126  177523     000
```

```
3158            015132                  .EVEN
3159  015132    000002                  XSTATQ: 2
3160  015134       006     003                  .BYTE   6,3
3161  015136    001244                         TEMP1
3162  015140       006     002                  .BYTE   6,2
3163  015142    001246                         TEMP2
3164                                    .EVEN
3165
3166                                           ;BUFFERS FOR INPUT-OUTPUT
3167
3168  015144    000000                  INBUF:  0
3169            015206                  .=.+40
3170  015206    000000                  TEMP:   0
3171            015250                  .=.+40
3172  015250    000000                  MDATA:  0
3173            015312                  .=.+40
3174
3175
3176  015312    000010                  CHRLNG: 10
3177  015314    000351                  DATAIN: 351
3178  015316    000000                  TMPDAT: 0
3179  015320    000000                  BCCPRV: 0
3180  015322    000000                  CALBCC: 0
3181  015324    000000                  BCCFBK: 0
3182  015326    000000                  XPOLY:  0
3183  015330    000000                  COUNT:  0
3184  015332    000000                  DATA:   0
3185  015334    000000                  SAVBCC: 0
3186  015336    000000                  SAVEPC: 0
3187  015340    000000                  SEC16:  0
3188  015342    000000                  SEC16X: 0
3189  015344    000000                  STORE1: 0
3190  015346    000000                  LOC1:   0
3191  015350       026     026          .SYNC:  .BYTE   26,26
3192  015352       026     026          SYNC:   .BYTE   26,26
3193  015354    000000                  TXBUFF: 0
3194            015756                  .=.+400
3195  015756    000000                  RXBUFF: 0
3196            016360                  .=.+400
3197  016360    000000                  .ERRTA: 0
3198  016362    000000                          0       ;HALT 0
3199  016364    000000                          0
3200  016366    000000                          0
3201  016370    016742                          DH0     ;HALT 1
3202  016372    017316                          DT0
3203  016374    000000                          0
3204  016376    016766                          DH1     ;HALT 2
3205  016400    017330                          DT1
3206  016402    016605                          EM0
3207  016404    017036                          DH2     ;HALT 3
3208  016406    017352                          DT2
3209  016410    016700                          EM2
3210  016412    017150                          DH4     ;HALT 4
3211  016414    017422                          DT4
3212  016416    016634                          EM1
3213  016420    017115                          DH3     ;HALT 5
```

```
                                                    DT3
                                                    EM1
                                                    DH5     ;HALT 6
                                                    DT5
                                                    EM3
                                                    DH6     ;HALT   7
                                                    DT6
 3221  016440  050377  042514  051501  MPOLY:   .ASCII  <377>/PLEASE SET SWITCH REGISTER TO POLYNOMIAL YOU DESIRE /
       016525     377  047524  041040           .ASCIZ  <377>/TO BE PLACED INTO POLYNOMIAL REGISTER OF DQ11./
       016605     377  040503  041514  EM0:     .ASCIZ  <377>/CALCULATED BCC ERROR./
       016634  046777  046505  051117  EM1:     .ASCIZ  <377>/MEMORY TRANSFER TEST  *DATA ERROR*/
       016700  041777  041101  042514  EM2:     .ASCIZ  <377>/CABLE TEST /
       016715     377  050504  030461  EM3:     .ASCIZ  <377>/DQ11 ERROR FLAG SET/
       016742  050377  046117  047131  DH0:     .ASCIZ <377>/POLYNOMIAL   DQERR/
       016766  050377  046117  047131  DH1:     .ASCIZ <377>/POLYNO   EXPECTED   RECEIVED   SEC REG/
       017036  050377  046117  047131  DH2:     .ASCIZ  <377>/POLYNO  CHAR  SHIFTS  EXPECTED  RECEIVED  REG/
       017115     377  042101  051104  DH3:     .ASCIZ  <377>/ADDRESS  EXPECTED  FOUND /
       017150  042377  052101  020101  DH4:     .ASCII  <377>/DATA COMPARISON ERROR /
       017177     377  054105  042520           .ASCIZ  <377>/EXPECTED   RECEIVED/
       017224  052377  020130  042101  DH5:     .ASCIZ  <377>/TX ADD  RX ADD  EXPECTED  FOUND /
       017266  042377  051121  051503  DH6:     .ASCIZ  <377>/DQRCSR  DQTCSR  DQERR /
                                                .EVEN
       017316  000002                   DT0:     2
 3222  017320     006     007                    .BYTE   6,7
 3223  017322  015326                            XPOLY
 3224  017324     006     001                    .BYTE   6,1
 3225  017326  001270                            SAVR5
 3226  017330  000004                   DT1:     4
 3227  017332     006     003                    .BYTE   6,3
 3228  017334  015326                            XPOLY
 3229  017336     006     005                    .BYTE   6,5
 3230  017340  015322                            CALBCC
 3231  017342     006     005                    .BYTE   6,5
 3232  017344  001260                            SAVR1
 3233  017346     002     001                    .BYTE   2,1
 3234  017350  001270                            SAVR5
 3235  017352  000006                   DT2:     6
 3236  017354     006     002                    .BYTE   6,2
 3237  017356  015326                            XPOLY
 3238  017360     003     003                    .BYTE   3,3
 3239  017362  015332                            DATA
 3240  017364     002     006                    .BYTE   2,6
 3241  017366  015330                            COUNT
 3242  017370     006     004                    .BYTE   6,4
 3243  017372  015322                            CALBCC
 3244  017374     006     004                    .BYTE   6,4
 3245  017376  001252                            TEMP4
 3246  017400     002     002                    .BYTE   2,2
 3247  017402  001254                            TEMP5
 3248  017404  000003                   DT3:     3
 3249  017406     006     003                            .BYTE   6,3
 3250  017410  001256                                    SAVR0
 3251  017412     006     004                            .BYTE   6,4
 3252  017414  001254                                    TEMP5
 3253  017416     006     002                            .BYTE   6,2
 3254  017420  001252                                    TEMP4
```

```
3255  017422  000002                  DT4:    2
3256  017424     006      004         .BYTE   6,4
3257  017426  001254                          TEMP5
3258  017430     006      002         .BYTE   6,2
3259  017432  001252                          TEMP4
3260  017434  000004                  DT5:    4
3261  017436     006      002                 .BYTE   6,2
3262  017440  001256                          SAVR0
3263  017442     006      002                 .BYTE   6,2
3264  017444  001260                          SAVR1
3265  017446     006      004                 .BYTE   6,4
3266  017450  001254                          TEMP5
3267  017452     006      002                 .BYTE   6,2
3268  017454  001252                          TEMP4
3269  017456  000003                  DT6:    3
3270  017460     006      002                 .BYTE   6,2
3271  017462  001256                          SAVR0
3272  017464     006      002                 .BYTE   6,2
3273  017466  001260                          SAVR1
3274  017470     006      002                 .BYTE   6,2
3275  017472  001262                          SAVR2
3276          000001                  .END
```

```
A8BIT = 002000
ACTBIT= 004000
ADRCNT= 013353
BA8IT = 010000
B8BIT = 020000
BCCFBK  015324
BCCPRV  015320
BINWRD  013672
BIT0  = 000001
BIT1  = 000002
BIT10 = 002000
BIT11 = 004000
BIT12 = 010000
BIT13 = 020000
BIT14 = 040000
BIT15 = 100000
BIT2  = 000004
BIT3  = 000010
BIT4  = 000020
BIT5  = 000040
BIT6  = 000100
BIT7  = 000200
BIT8  = 000400
BIT9  = 001000
BRW     012636
BRX     012640
CALBCC  015322
CHARDT= 000010
CHAR1   001236
CHAR2   001240
CHAR3   001242
CHRCNT  013670
CHRLNG  015312
CKDN    002706
CKSWR = 104414
CKSYN1  002662
CNTLU = 104415
CNVRT = 104411
CONVRT= 104410
COUNT   015330
CREAM   001300
CSRMAP  000220
DATA    015332
DATABP  014150
DATAHD  014136
DATAIN  015314
DEVADR  013350
DH0     016742
DH1     016766
DH2     017036
DH3     017115
DH4     017150
DH5     017224
DH6     017266
DISPRE  000174
DLIGHT= 177570
DQACTV  001500
DQCR00  001400
DQCR01  001404
DQCR02  001410
DQCR03  001414
DQCR04  001420
DQCR05  001424
DQCR06  001430
DQCR07  001434
DQCR10  001440
DQCR11  001444
DQCR12  001450
DQCR13  001454
DQCR14  001460
DQCR15  001464
DQCR16  001470
DQCR17  001474
DQCSR   001506
DQERR   001366
DQNUM   001504
DQRCSH  001362
DQRCSR  001360
DQREG   001370
DQRLVL  001352
DQRVEC  001350
DQSEC   001372
DQSECH  001374
DQSTAT  001510
DQST00  001402
DQST01  001406
DQST02  001412
DQST03  001416
DQST04  001422
DQST05  001426
DQST06  001432
DQST07  001436
DQST10  001442
DQST11  001446
DQST12  001452
DQST13  001456
DQST14  001462
DQST15  001466
DQST16  001472
DQST17  001476
DQTCSR  001364
DQTLVL  001356
DQTVEC  001354
DSWR  = 177570
DT0     017316
DT1     017330
DT2     017352
DT3     017404
DT4     017422
DT5     017434
DT6     017456
EM0     016605
EM1     016634
EM2     016700
EM3     016715
ENDBCC  010544
ENDSYN  011064
ENDTX   012040
ENDTY   011432
ERRCNT  001232
ERRFLG  001312
ERRMSG  014124
ERTAB0  014242
EXITER  014174
HALTS   014154
HILIM   013346
ICOUNT  001222
INBUF   015144
INIFLG  001310
INSTER= 104404
INSTR = 104403
INSTR2  013132
JUMBIT= 040000
LIGHTS  001202
LIMITS  013274
LIMIT.  004454
LOBITS  013352
LOCK    001220
LOC1    015346
LOGICA  012454
LOKFLG  001313
LOLIM   013344
LPCNT   001224
LSTERR  001234
MCRLF   014532
MCSRX   014753
MDATA   015250
MEMCLR= 104413
MEPASS  014572
MERRPC  015077
MERRX   015000
MERR2   014617
MERR3   014666
MISC. = 000012
MLOCK   014724
MNEW    015025
MPASSX  014767
MPFAIL  014534
MPOLY   016440
MQM     014526
MR      014614
MSTCLR= 104412
MTITLE  001000
MTSTN   015011
MTSTPC  014712
MVECX   014761
NEXT    001216
ODDBIT= 001000
OUT     014466
PARAM = 104405
PARAM1  013200
PARERR  013254
PARTI   013336
PASCNT  001230
PC    =%000007
PFTAB   014344
POLY. = 000017
POPR0 = 012600
POP1SP= 005726
POP2SP= 022626
PS    = 177776
PUSHR0= 010046
PUSH1S= 005746
PUSH2S= 024646
RDSW    014352
RESREG  014152
RESTAR  014270
RESTRT  012464
RESUS = 104407
RETURN  001214
RUN     001304
RUNCNT  001306
RUNFLG  001302
RXBA.P= 000000
RXBA.S= 000004
RXBUFF  015756
RXWC.P= 000001
RXWC.S= 000005
RX.BCC= 000015
R0    =%000000
R1    =%000001
R2    =%000002
R3    =%000003
R4    =%000004
R5    =%000005
SAVACT  001502
SAVBCC  015334
SAVEPC  015336
SAVNUM  001276
SAVPC   001274
SAVR0   001256
SAVR1   001260
SAVR2   001262
SAVR3   001264
SAVR4   001266
SAVR5   001270
SAVSP   001272
SAVUS = 104406
SCOPE = 104400
SCOP1 = 104401
SEC16   015340
SEC16X  015342
SEQ.  = 000014
SIMBCC  012042
SP    =%000006
SPACNT= 013671
STACK = 001200
STBCC   010160
STFLG   001311
STORE1  015344
SVOS    013362
SWR     001200
SWREG   000176
SWREGC  014474
SW00  = 000001
SW01  = 000002
SW02  = 000004
SW03  = 000010
SW04  = 000020
SW05  = 000040
SW06  = 000100
SW08  = 000400
SW09  = 001000
SW10  = 002000
SW11  = 004000
SW12  = 010000
SW13  = 020000
SW14  = 040000
SW15  = 100000
SYNBCC  010546
SYNBIT= 100000
SYNC    015352
SYNC. = 000011
TEMP    015206
TEMP1   001244
TEMP2   001246
TEMP3   001250
TEMP4   001252
TEMP5   001254
TKCSR   001204
TKDBR   001206
TLAST = 007450
TMPDAT  015316
TPCSR   001210
TPDBR   001212
```

```
DZDQE   MACY11 27(732)  24-MAY-76  13:14  PAGE 67
DZDQEC.P11      SYMBOL TABLE

TRPOK   013706          TST5    004456          TYPMSG  014062          $E    = 000023          .INST1  013010
TSTNO   001226          TST6    005000          USEBCC  010106          $MNEW   014515          .MEMCL  012200
TST1    002254          TST7    005106          VECMAP  000056          $MSWR   014506          .MSG    013012
TST10   005614          TTST    012540          WORD    004776          $N    = 000021          .MSTCL  012324
TST11   006640          TXBA.P= 000002          WRDCNT  013666          $Y    = 000016          .PARAM  013140
TST12   006704          TXBA.S= 000006          WRKO.F  014126          .BEGIN  002116          .PFAIL  014256
TST13   006750          TXBCC   011424          XBX     013754          .CKSWR  014354          .RESOS  013414
TST14   007014          TXBUFF  015354          XCSR    012500          .CNTLU  014430          .SAVOS  013354
TST15   007060          TXISR1  010462          XERR    012522          .CNVRT  013452          .SCOPE  012530
TST16   007124          TXWC.P= 000003          XHEAD   015104          .CONVR  013446          .SCOPI  012642
TST17   007220          TXWC.S= 000007          XPASS   012514          .EOP    012342          .START  001512
TST2    002646          TX.BCC= 000016          XPOLY   015326          .ERRTA  016360          .SYNC   015350
TST20   007314          TX.MUX= 000013          XSTATQ  015132          .HLT    013726          .TRPSR  013674
TST21   007450          TYBCC   011066          XTSTN   014250          .INSTE  013106          .TRPTA  001314
TST3    003226          TYFDAT  014140          XVEC    012506          .INSTG  013112          .TYPE   012662
TST4    003520          TYPE  = 104402          $CNTG   014502          .INSTR  012770          .     = 017474


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *TUKE:DZDQEC,DZDQEC/SOL←UNIV.LIB.DZDQEC.P11
 RUN-TIME:  19 30 1 SECONDS
 RUN-TIME RATIO: 115/51=2.2
 CORE USED:  19K  (37 PAGES)
```