

# DQ11

MISC RX/TX PLUS BCC TEST  
MD-11-DZDQE-D

EP-DZDQD-D-DL-B  
COPYRIGHT © 1977  
FICHE 1 OF 1

APR 1977  
**digital**  
MADE IN USA

11 12 13 14 15 16 17 18 19 20



B01

EOF1DZDQ00SEQ

00010000

770325

PDP10 411

4 HDR1DZDQEDSEQ

00010000

770325



CO1

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 2  
DZDQED.P11 16-DEC-76 13:29

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDQE-D-D  
PRODUCT NAME: MISC. RX AND TX TEST PLUS BCC TESTS  
DATE: MARCH 1977  
MAINTAINER: DIAGNOSTIC GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

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

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

COPYRIGHT (C) 1974, 1977 BY DIGITAL EQUIPMENT CORPORATION



## 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 TRANSFERRED IN BLOCKS THE ACTUAL 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 SUCCESSFUL. 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 INTERRUPT 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 DISCUSSED LATER.  
1. DZDQO [REV] ONLINE TEST. (ITEP OVERLAY)

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

## 2.



DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 4  
 DZDQED.P11 16-DEC-76 13:29

## 2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 4K MEMORY)-WITH  
 OR WITHOUT A HARDWARE SWITCH REGISTER (LOC. 177570)  
 ASR 33 (OR EQUIVALENT)  
 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=''  
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 "SCOPI" IS IN THAT TEST.  
THE REASON BEING THAT MOST TESTS  
DEAL WITH BLOCKS OF DIFFERENT DATA  
TO BE SENT OR RECEIVED ALL AT ONCE  
THUS IN BLOCK DATA; ONE PATTERN 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 "SCOPI")
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 POSSIBLY 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 ALWAYS BE A TEST NUMBER AND PC TYPED OUT AT THE TIME OF AN ERROR (PROVIDING SW 13=0 AND SW 12=0). IN MOST CASES ADDITIONAL INFORMATION WILL BE SUPPLIED 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 1226) 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



DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 9  
 DZDQED.P11 16-DEC-76 13:29

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-D 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 (1214) CONTAINS THE ADDRESS WHERE PROGRAM WILL RETURN WHEN ITERATION COUNT IS REACHED OR IF LOOP ON TEST IS ASSERTED.  
 NEXT (1216) CONTAINS THE ADDRESS OF THE NEXT TEST TO BE PERFORMED.  
 TSTNO (1226) CONTAINS THE NUMBER OF THE TEST NOW BEING PERFORMED.  
 RUN (1304) THE BIT IN "RUN" ALWAYS POINTS ONE PAST THE DQ11 CURRENTLY BEING TESTED.  
 EXAMPLE:  
 (RUN) 1304/0000000001000000  
 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 SEQUENTIALLY. 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:	BB 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 APPRIORATE 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 EXISTS

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 APPRIORATE 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 APPRIORATE 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

544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580

.ENABLE AMA

;MAINDEC-11-DZDQE-D/<377>/TX AND RX MISC. AND BCC TESTS  
;COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

;REVISED 16-DEC-76 BY R. BLACK

... A)SUPPORTS SOFTWARE SWITCH REGISTER  
... B)SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER  
... BY <IG>.

;STARTING PROCEDURE  
;LOAD PROGRAM  
;LOAD ADDRESS 000200  
;PRESS START  
;PROGRAM WILL TYPE "MAINDEC-11-DZDQE-D/<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  
040000  
020000  
010000  
004000  
002000  
001000  
000400  
000100  
000040  
000020  
000010  
000004  
000002  
000001

SW15=100000  
SW14=40000  
SW13=20000  
SW12=10000  
SW11=4000  
SW10=2000  
SW09=1000  
SW08=400  
SW06=100  
SW05=40  
SW04=20  
SW03= 0  
SW02=1  
SW01=2  
SW00=1

=1,HALT ON ERROR  
=1,LOOP ON CURRENT TEST  
=1,INHIBIT ERROR TIMEOUT  
=1,DELETE TIMEOUT/BELL ON ERROR.  
=1,INHIBIT ITERATIONS  
=1,ESCAPE TO NEXT TEST ON ERROR  
=1,LOOP WITH CURRENT DATA  
=1,LOOP ON ERROR

;LOCK ON TEST SELECT  
;RESTART PROGRAM AT SELECTED TEST  
;RESELECT DQ11 DESIRED ACTIVE  
;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT



# NO1

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 13  
 DZDQED.P11 16-DEC-76 13:29 GENERAL DEFINATIONS AND EQUIVALENCIES

```

581
582
583           ;REGISTER DEFINITIONS
584
585           000000      R0=%0           ;GENERAL REGISTER
586           000001      R1=%1           ;GENERAL REGISTER
587           000002      R2=%2           ;GENERAL REGISTER
588           000003      R3=%3           ;GENERAL REGISTER
589           000004      R4=%4           ;GENERAL REGISTER
590           000005      R5=%5           ;GENERAL REGISTER
591           000006      SP=%6          ;PROCESSOR STACK POINTER
592           000007      PC=%7          ;PROGRAM COUNTER
593
594           ;LOCATION EQUIVALENCIES
595
596           177570      DSWR= 177570    ;HARDWARE SWITCH REGISTER LOC.
597           177570      DLIGHTS=177570 ;HARDWARE DISPLAY REGISTER LOC.
598           177776      PS=177776     ;PROCESSOR STATUS WORD
599           001200      STACK=1200     ;START OF PROCESSOR STACK
600
601           ;INSTRUCTION DEFINITIONS
602
603           005746      PUSH1SP=5746   ;DECREMENT PROCESSOR STACK 1 WORD
604           005726      POP1SP=5726    ;INCREMENT PROCESSOR STACK 1 WORD
605           010046      PUSHRO=10046   ;SAVE R0 ON STACK
606           012600      POPRO=12600    ;RESTORE R0 FROM STACK
607           024646      PUSH2SP=24646 ;DECREMENT STACK TWICE
608           022626      POP2SP=22626  ;INCREMENT STACK TWICE
609           .EQUIV EMT,HLT ;BASIC DEFINITION OF ERROR CALL
610
611
612           100000      BIT15=100000
613           040000      BIT14=40000
614           020000      BIT13=20000
615           010000      BIT12=10000
616           004000      BIT11=4000
617           002000      BIT10=2000
618           001000      BIT9=1000
619           000400      BIT8=400
620           000200      BIT7=200
621           000100      BIT6=100
622           000040      BIT5=40
623           000020      BIT4=20
624           000010      BIT3=10
625           000004      BIT2=4
626           000002      BIT1=2
627           000001      BIT0=1
628
629
630           ;DQ11 OPTIONAL DEFINITIONS
631
632           002000      ABBIT=2000
633           004000      ACTBIT=4000
634           010000      BABIT=10000
635           020000      BBBIT=20000
636           040000      JUMBIT=40000
  
```

637 001000  
 638 100000  
 639  
 640  
 641  
 642  
 643 000000  
 644 000001  
 645 000002  
 646 000003  
 647 000004  
 648 000005  
 649 000006  
 650 000007  
 651  
 652 000010  
 653 000011  
 654 000012  
 655 000013  
 656 000014  
 657 000015  
 658 000016  
 659 000017  
 660  
 661

ODDBIT=1000  
 SYNBIT=100000

;DQ11 SECONDARY REGISTER DEFINATIONS

RXBA.P=0	;RECEIVER BUS ADDRESS PRIMARY.
RXWC.P=1	;RECEIVER WORD COUNT PRIMARY.
TXBA.P=2	;TRANSMITTER BUS ADDRESS PRIMARY.
TXWC.P=3	;TRANSMITTER BUS ADDRESS PRIMARY.
RXBA.S=4	;RECEIVER BUS ADDRESS SECONDARY.
RXWC.S=5	;RECEIVER WORD COUNT SECONDARY.
TXBA.S=6	;TRANSMITTER BUS ADDRESS SECONDARY.
TXWC.S=7	;TRANSMITTER WORD COUNT SECONDARY.
CHARDT=10	;CHARACTER DETECT REGISTER.
SYNC.=11	;SYNC REGISTER.
MISC.=12	;MISCELLANEOUS REGISTER.
TX.MUX=13	;TRANSMITTER MUX REGISTER.
SEQ.=14	;SEQUENCE REGISTER.
RX.BCC=15	;RECEIVER BCC REGISTER.
TX.BCC=16	;TRANSMITTER BCC REGISTER.
POLY.=17	;POLYNOMIAL REGISTER.



```

        ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
662      .=0
663      000000 000000
664      000000 000002      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
665      000002 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
666      000004 000006      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
667      000006 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
668      000010 000012      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
669      000012 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
670      000014 000016      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
671      000016 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
672      000020 000022      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
673      000022 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
674      000024 000026      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
675      000026 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
676      000030 000032      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
677      000032 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
678      000034 000036      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
679      000036 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
680      000040 000042      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
681      000042 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
682      000044 000046      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
683      000046 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
684      000050 000052      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
685      000052 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
686      000054 000056      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
687      000056 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
688      000060 000062      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
689      000062 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
690      000064 000066      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
691      000066 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
692      000070 000072      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
693      000072 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
694      000074 000076      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
695      000076 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
696      000100 000102      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
697      000102 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
698      000104 000106      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
699      000106 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
700      000110 000112      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
701      000112 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
702      000114 000116      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
703      000116 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
704      000120 000122      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
705      000122 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
706      000124 000126      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
707      000126 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
708      000130 000132      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
709      000132 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
710      000134 000136      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
711      000136 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
712      000140 000142      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
713      000142 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
714      000144 000146      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
715      000146 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
716      000150 000152      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
717      000152 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
    
```

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 16  
 DZDQED.P11 16-DEC-76 13:29 TRAPCATCHER FOR UNEXPECTED INTERRUPTS

718	000154	000156	.+2	:UNEXPECTED TRAP TO THIS LOCATION
719	000156	000000	HALT	:EXAMINE STACK TO FIND CAUSE
720	000160	000162	.+2	:UNEXPECTED TRAP TO THIS LOCATION
721	000162	000000	HALT	:EXAMINE STACK TO FIND CAUSE
722	000164	000166	.+2	:UNEXPECTED TRAP TO THIS LOCATION
723	000166	000000	HALT	:EXAMINE STACK TO FIND CAUSE
724	000170	000172	.+2	:UNEXPECTED TRAP TO THIS LOCATION
725	000172	000000	HALT	:EXAMINE STACK TO FIND CAUSE
726	000174	000176	.+2	:UNEXPECTED TRAP TO THIS LOCATION
727	000176	000000	HALT	:EXAMINE STACK TO FIND CAUSE
728	000200	000202	.+2	:UNEXPECTED TRAP TO THIS LOCATION
729	000202	000000	HALT	:EXAMINE STACK TO FIND CAUSE
730	000204	000206	.+2	:UNEXPECTED TRAP TO THIS LOCATION
731	000206	000000	HALT	:EXAMINE STACK TO FIND CAUSE
732	000210	000212	.+2	:UNEXPECTED TRAP TO THIS LOCATION
733	000212	000000	HALT	:EXAMINE STACK TO FIND CAUSE
734	000214	000216	.+2	:UNEXPECTED TRAP TO THIS LOCATION
735	000216	000000	HALT	:EXAMINE STACK TO FIND CAUSE
736	000220	000222	.+2	:UNEXPECTED TRAP TO THIS LOCATION
737	000222	000000	HALT	:EXAMINE STACK TO FIND CAUSE
738	000224	000226	.+2	:UNEXPECTED TRAP TO THIS LOCATION
739	000226	000000	HALT	:EXAMINE STACK TO FIND CAUSE
740	000230	000232	.+2	:UNEXPECTED TRAP TO THIS LOCATION
741	000232	000000	HALT	:EXAMINE STACK TO FIND CAUSE
742	000234	000236	.+2	:UNEXPECTED TRAP TO THIS LOCATION
743	000236	000000	HALT	:EXAMINE STACK TO FIND CAUSE
744	000240	000242	.+2	:UNEXPECTED TRAP TO THIS LOCATION
745	000242	000000	HALT	:EXAMINE STACK TO FIND CAUSE
746	000244	000246	.+2	:UNEXPECTED TRAP TO THIS LOCATION
747	000246	000000	HALT	:EXAMINE STACK TO FIND CAUSE
748	000250	000252	.+2	:UNEXPECTED TRAP TO THIS LOCATION
749	000252	000000	HALT	:EXAMINE STACK TO FIND CAUSE
750	000254	000256	.+2	:UNEXPECTED TRAP TO THIS LOCATION
751	000256	000000	HALT	:EXAMINE STACK TO FIND CAUSE
752	000260	000262	.+2	:UNEXPECTED TRAP TO THIS LOCATION
753	000262	000000	HALT	:EXAMINE STACK TO FIND CAUSE
754	000264	000266	.+2	:UNEXPECTED TRAP TO THIS LOCATION
755	000266	000000	HALT	:EXAMINE STACK TO FIND CAUSE
756	000270	000272	.+2	:UNEXPECTED TRAP TO THIS LOCATION
757	000272	000000	HALT	:EXAMINE STACK TO FIND CAUSE
758	000274	000276	.+2	:UNEXPECTED TRAP TO THIS LOCATION
759	000276	000000	HALT	:EXAMINE STACK TO FIND CAUSE
760	000300	000302	.+2	:UNEXPECTED TRAP TO THIS LOCATION
761	000302	000000	HALT	:EXAMINE STACK TO FIND CAUSE
762	000304	000306	.+2	:UNEXPECTED TRAP TO THIS LOCATION
763	000306	000000	HALT	:EXAMINE STACK TO FIND CAUSE
764	000310	000312	.+2	:UNEXPECTED TRAP TO THIS LOCATION
765	000312	000000	HALT	:EXAMINE STACK TO FIND CAUSE
766	000314	000316	.+2	:UNEXPECTED TRAP TO THIS LOCATION
767	000316	000000	HALT	:EXAMINE STACK TO FIND CAUSE
768	000320	000322	.+2	:UNEXPECTED TRAP TO THIS LOCATION
769	000322	000000	HALT	:EXAMINE STACK TO FIND CAUSE
770	000324	000326	.+2	:UNEXPECTED TRAP TO THIS LOCATION
771	000326	000000	HALT	:EXAMINE STACK TO FIND CAUSE
772	000330	000332	.+2	:UNEXPECTED TRAP TO THIS LOCATION
773	000332	000000	HALT	:EXAMINE STACK TO FIND CAUSE



774	000334	000336	.+2	:UNEXPECTED TRAP TO THIS LOCATION
775	000336	000000	HALT	:EXAMINE STACK TO FIND CAUSE
776	000340	000342	.+2	:UNEXPECTED TRAP TO THIS LOCATION
777	000342	000000	HALT	:EXAMINE STACK TO FIND CAUSE
778	000344	000346	.+2	:UNEXPECTED TRAP TO THIS LOCATION
779	000346	000000	HALT	:EXAMINE STACK TO FIND CAUSE
780	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
781	000352	000000	HALT	:EXAMINE STACK TO FIND CAUSE
782	000354	000356	.+2	:UNEXPECTED TRAP TO THIS LOCATION
783	000356	000000	HALT	:EXAMINE STACK TO FIND CAUSE
784	000360	000362	.+2	:UNEXPECTED TRAP TO THIS LOCATION
785	000362	000000	HALT	:EXAMINE STACK TO FIND CAUSE
786	000364	000366	.+2	:UNEXPECTED TRAP TO THIS LOCATION
787	000366	000000	HALT	:EXAMINE STACK TO FIND CAUSE
788	000370	000372	.+2	:UNEXPECTED TRAP TO THIS LOCATION
789	000372	000000	HALT	:EXAMINE STACK TO FIND CAUSE
790	000374	000376	.+2	:UNEXPECTED TRAP TO THIS LOCATION
791	000376	000000	HALT	:EXAMINE STACK TO FIND CAUSE
792	000400	000402	.+2	:UNEXPECTED TRAP TO THIS LOCATION
793	000402	000000	HALT	:EXAMINE STACK TO FIND CAUSE
794	000404	000406	.+2	:UNEXPECTED TRAP TO THIS LOCATION
795	000406	000000	HALT	:EXAMINE STACK TO FIND CAUSE
796	000410	000412	.+2	:UNEXPECTED TRAP TO THIS LOCATION
797	000412	000000	HALT	:EXAMINE STACK TO FIND CAUSE
798	000414	000416	.+2	:UNEXPECTED TRAP TO THIS LOCATION
799	000416	000000	HALT	:EXAMINE STACK TO FIND CAUSE
800	000420	000422	.+2	:UNEXPECTED TRAP TO THIS LOCATION
801	000422	000000	HALT	:EXAMINE STACK TO FIND CAUSE
802	000424	000426	.+2	:UNEXPECTED TRAP TO THIS LOCATION
803	000426	000000	HALT	:EXAMINE STACK TO FIND CAUSE
804	000430	000432	.+2	:UNEXPECTED TRAP TO THIS LOCATION
805	000432	000000	HALT	:EXAMINE STACK TO FIND CAUSE
806	000434	000436	.+2	:UNEXPECTED TRAP TO THIS LOCATION
807	000436	000000	HALT	:EXAMINE STACK TO FIND CAUSE
808	000440	000442	.+2	:UNEXPECTED TRAP TO THIS LOCATION
809	000442	000000	HALT	:EXAMINE STACK TO FIND CAUSE
810	000444	000446	.+2	:UNEXPECTED TRAP TO THIS LOCATION
811	000446	000000	HALT	:EXAMINE STACK TO FIND CAUSE
812	000450	000452	.+2	:UNEXPECTED TRAP TO THIS LOCATION
813	000452	000000	HALT	:EXAMINE STACK TO FIND CAUSE
814	000454	000456	.+2	:UNEXPECTED TRAP TO THIS LOCATION
815	000456	000000	HALT	:EXAMINE STACK TO FIND CAUSE
816	000460	000462	.+2	:UNEXPECTED TRAP TO THIS LOCATION
817	000462	000000	HALT	:EXAMINE STACK TO FIND CAUSE
818	000464	000466	.+2	:UNEXPECTED TRAP TO THIS LOCATION
819	000466	000000	HALT	:EXAMINE STACK TO FIND CAUSE
820	000470	000472	.+2	:UNEXPECTED TRAP TO THIS LOCATION
821	000472	000000	HALT	:EXAMINE STACK TO FIND CAUSE
822	000474	000476	.+2	:UNEXPECTED TRAP TO THIS LOCATION
823	000476	000000	HALT	:EXAMINE STACK TO FIND CAUSE
824	000500	000502	.+2	:UNEXPECTED TRAP TO THIS LOCATION
825	000502	000000	HALT	:EXAMINE STACK TO FIND CAUSE
826	000504	000506	.+2	:UNEXPECTED TRAP TO THIS LOCATION
827	000506	000000	HALT	:EXAMINE STACK TO FIND CAUSE
828	000510	000512	.+2	:UNEXPECTED TRAP TO THIS LOCATION
829	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



DZDGE MACY11 27(1006) 22-DEC-76 11:32 PAGE 19  
 DZDGED.P11 16-DEC-76 13:29 TRAPCATCHER FOR UNEXPECTED INTERUPTS

886	000674	000676	.+2	:UNEXPECTED TRAP TO THIS LOCATION
887	000676	000000	HALT	:EXAMINE STACK TO FIND CAUSE
888	000700	000702	.+2	:UNEXPECTED TRAP TO THIS LOCATION
889	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

```

920                                     ;STANDARD INTERRUPT VECTORS
921
922                                     .=24
923 000024 014256                       .PFAIL                       ;POWER FAIL HANDLER
924 000026 000340                       340                          ;SERVICE AT LEVEL 7
925 000030 013726                       .HLT                          ;ERROR HANDLER
926 000032 000340                       340                          ;SERVICE AT LEVEL 7
927 000034 013674                       .TRPSRV                       ;GENERAL HANDLER DISPATCH SERVICE
928 000036 000340                       340                          ;SERVICE AT LEVEL 7
929
930 000046 012454                       .=46                          LOGICAL                       ;ACT HOOKS
931                                     .=52
932 000052 000000                       .WORD 0
933                                     ; THIS ROUTINE TRIES TO FORCE THE RECEIVER TO INTERRUPT
934                                     ; TO ITS VECTOR WHERE IT WILL PICK UP THE STATUS LOCATION
935                                     ; FOR ITS NEW PC; AND PICK UP AN IOT INSTRUCTION FOR ITS
936                                     ; NEW PS. WHEN THE NEW PC IS FETCHED AN IOT INSTRUCTION IS
937                                     ; EXECUTED, TRAPPING TO LOCATION 20 WHERE A ROUTINE IS EXECUTED
938                                     ; TO TAKE THE PC FROM THE STACK AND USE IT AS THE VECTOR ADDRESS
939                                     .=56
940
941 000056
942 000056 010120 000004                 VECMAP:
943 000060 012721 000004                 1$:  MOV R1,(R0)+                ;START FILLING THE VECTOR AREA
944 000064 022021                        MOV #4,(R1)+                  ;WITH .+2; IOT (4)
945 000066 020127 001000                 CMP (R0)+(R1)+               ;UPDATE THE POINTERS
946 000072 101771                        CMP R1,#1000                 ;IS ALL FLOATING VECTOR AREA DONE
947 000074 012737 000146 000020         BLOS 1$                      ;BR IF NOT ALL DONE
948 000102 013737 001500 001244         MOV #4$,#20                 ;SET FOR IOT TRAP BY DQ11
949 000110 006037 001244                 MOV DQACTV,TEMP1            ;GET THE ACTIVE DQ11 S
950 000114 103023                        2$:  ROR TEMP1                 ;ARE YOU ACTIVE.. DQ11
951 000116 005037 177776                 BCC 5$                       ;IF CARRY CLEAR.. NO MORE DQ11S
952 000122 005722                        CLR PS                       ;CLEAR PS
953 000124 012772 000340 177776         TST (R2)+                   ;PUT POINTER TO STATUS TABLE
954 000132 105200                        MOV #340,#-2(R2)            ;TRY AND SET PRI/SEC DONE AND IE
955 000134 001376                        INCB RO                     ;DELAY.....
956 000136 112712 000300                 BNE -2                       ;.....DELAY
957 000142 005722                        3$:  MOVB #300,(R2)             ;NO INTERRUPT ASSUME 300 FIX IN TEST C
958 000144 000761                        TST (R2)+                   ;UPDATE POINTERS
959 000146 051612                        BR 2$                       ;GO DO IT AGAIN
960 000150 042712 000007                 4$:  BIS (SP),(R2)             ;ENTERD BY IOT TRAP BY DQ11
961 000154 022626                        BIC #7,(R2)                 ;CLEAR UNWANTED BITS
962 000156 012716 000142                 CMP (SP)+(SP)+              ;POP IOT JUNK OFF STACK
963 000162 000002                        MOV #3$, (SP)               ;SET RETURN PC ON STACK
964 000164 000207                        RTI                          ;GO HOME.
965                                     5$:  RTS PC                   ;ALL SIZING IS DONE
966
967                                     ;***SOFTWARE SWITCH REGISTER***
968 000174 000000                       .=174
969 000176 000000                       DISPREG: 0                   ;SOFTWARE DISPLAY REGISTER
970                                     SWREG: 0                     ;SOFTWARE SWITCH REGISTER
971
972                                     ;PROGRAM START
973                                     .=200
974 000200 000137 001512                 JMP .START                   ;GO TO START OF PROGRAM
975
    
```



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	#55,2#4			;SET FOR TIME OUT TRAP--NO DEVICE--		
985	000256	112761	000012	000005	1\$:	MOV#B	#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		MOV#B	#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			;APPARENTLY NO BB OPTION.		
997	000336	052712	020000			BIS	#BBBIT,(R2)			;SET FOR BB OPTION		
998	000342	112761	000017	000005		MOV#B	#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, R4      ; RESET TIME OUT VECTOR
1033 000546 013737 001500 001502      MOV      DQACTV, SAVACT ; SAVE ACTIVE
1034 000554 012737 000340 000022      MOV      #340, R22   ; 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          4S:     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
1043 000612 000776          BR      -2          ; HOW CAN I TEST NO DQ11S
1044 000614 012716 000466          5S:     MOV      #25, (SP) ; DON'T LET OPR HIT CONT. SW
1045 000620 000002          RTI        ; ENTERED BY TIME OUT TRAP
1046
1047
1048
1049 001000 005377 040515 047111      .=1000  MTITLE: .ASCIZ <377><12>/MAINDEC-11-DZDQE-D/<377>/TX AND RX MISC. AND BCC TESTS/<377>
1050 001006 042504 026503 030461
1051 001014 042055 042132 042521
1052 001022 042055 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 ITERATIONS 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
    
```



# K02

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 23  
DZDQED.P11 16-DEC-76 13:29

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

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		

```

1104
1105                ;PROGRAM CONTROL FLAGS
1106
1107 001310      000      INIFLG: .BYTE 0          ;PROGRAM INITIALIZATION FLAG
1108 001311      000      STFLG:  .BYTE 0          ;TEST START FLAG
1109 001312      000      ERRFLG: .BYTE 0          ;ERROR OCCURED FLAG
1110 001313      000      LOKFLG: .BYTE 0          ;LOCK ON CURRENT TEST FLAG
1111                SY=0
1112
1113                ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
1114                ;POINTERS TO SUBROUTINES CAN BE FOUND
1115                ;IN THE TABLE IMMEDIATLY FOLLOWING THE DEFINITIONS
1116
1117                ;*****
1118                ;*****
1119 001314      .TRPTAB:
1120                SCOPE=TRAP+0          ;CALL TO SCOPE LOOP AND ITERATION HANDLER
1121 001314      104400      .SCOPE
1122                SCOP1=TRAP+1          ;CALL TO LOOP ON CURRENT DATA HANDLER
1123 001316      104401      .SCOP1
1124                TYPE=TRAP+2          ;CALL TO TELETYPE OUTPUT ROUTINE
1125 001320      104402      .TYPE
1126                INSTR=TRAP+3          ;CALL TO ASCII STRING INPUT ROUTINE
1127 001322      104403      .INSTR
1128                INSTER=TRAP+4          ;CALL TO INPUT ERROR HANDLER
1129 001324      104404      .INSTER
1130                PARAM=TRAP+5          ;CALL TO NUMERICAL DATA INPUT ROUTINE
1131 001326      104405      .PARAM
1132                SAVOS=TRAP+6          ;CALL TO REGISTER SAVE ROUTINE
1133 001330      104406      .SAVOS
1134                RESOS=TRAP+7          ;CALL TO REGISTER RESTORE ROUTINE
1135 001332      104407      .RESOS
1136                CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
1137 001334      104410      .CONVRT
1138                CNVRT=TRAP+11          ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
1139 001336      104411      .CNVRT
1140                MSTCLR=TRAP+12          ;CALL TO ISSUE MASTER CLEAR
1141 001340      104412      .MSTCLR
1142                MEMCLR=TRAP+13          ;CALL TO CLEAR ALL SCRATCH PAD MEMORIES
1143 001342      104413      .MEMCLR
1144                CKSWR=TRAP+14          ;CALL TO ALLOW SWREG TO BE LOADED FROM TTY
1145 001344      104414      .CKSWR
1146                CNTLU=TRAP+15          ;CALL TO ALLOW LOADING OF SWREG FROM TTY
1147 001346      104415      .CNTLU
1148
1149                ;*****
1150                ;*****
1151
1152                ;DQ11 VECTOR AND REGISTER INDIRECT POINTERS
1153
1154 001350      000000      DQ11VEC: 0          ;POINTER TO DQ11 RECEIVER INTERRUPT VECTOR
1155 001352      000000      DQ11LVL: 0          ;POINTER TO DQ11 RECEIVER INTERRUPT SERVICE PS
1156 001354      000000      DQ11TVEC: 0          ;POINTER TO DQ11 TRANSMITTER INTERRUPT VECTOR
1157 001356      000000      DQ11TLVL: 0          ;POINTER TO DQ11 TRANSMITTER INTERRUPT SERVICE PS
1158 001360      000000      DQ11CSR: 0          ;POINTER TO DQ11 RECEIVER CONTROL REGISTER
1159 001362      000000      DQ11CSH: 0          ;POINTER TO HIGH BYTE OF DQ11 RECEIVER CONTROL REGISTER
    
```



# M02

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 25  
 DZDQED.P11 16-DEC-76 13:29

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

1160	001364	000000	DQTCR: 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  
1169

;DQ11 STATUS TABLE AND ADDRESS ASSIGNMENTS

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 OF DQ11S
1206	001506	000001	DQCSR: .BLKW 1	; CSR OF DQ11 UNDER TEST
1207	001510	000001	DQSTAT: .BLKW 1	; VECTOR AND CONFIGURATION STATUS OF DQ11 UNDER TEST

1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215

;PROGRAM INITIALIZATION  
 ;LOCK OUT INTERRUPTS  
 ;SET UP PROCESSOR STACK  
 ;SET UP POWER FAIL VECTOR  
 ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS  
 ;TYPE TITLE MESSAGE

# N02

DZDGE MACY11 27(1006) 22-DEC-76 11:32 PAGE 26  
 DZDGED.P11 16-DEC-76 13:29 PROGRAM INITIALIZATION AND START UP.

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	QANUM,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 TESTING STARTS
1229								;HAS INITIALIZATION BEEN PERFORMED
1230	001612	105737	001310			TSTB	INIFLG	
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	RO	
1269	002030	000000				HALT		
1270	002032	104414				CKSWR		
1271	002034	027737	177140	001502		CMP	#SWR,SAVACT	



```

1272 002042 101404      BLOS      11$
1273 002044 104402      TYPE
1274 002046 014666      MERR3
1275 002050 000000      HALT
1276 002052 000776      BR
1277 002054 017737 177120 001500 11$: MOV      @SWR,DQACTV
1278 002062 013700 001500      MOV      DQACTV,RO
1279 002066 000000      HALT
1280 002070 104414      CKSWR
1281 002072 012700 000300 1$: MOV      #300,RO
1282 002076 012701 000302      MOV      #302,R1
1283 002102 010120 2$: MOV      R1,(R0)+
1284 002104 005021      CLR      (R1)+
1285 002106 022021      CMP      (RO)+,(R1)+
1286 002110 022700 001000      CMP      #1000,RO
1287 002114 001372      BNE      2$
1288
1289 ;TEST START AND RESTART
1290
1291 002116 012737 000340 177776 .BEGIN: MOV      #340,PS ;LOCK OUT INTERRUPTS
1292 002124 012706 001200      MOV      #STACK,SP ;SET UP STACK
1293 002130 005737 000042      TST      @#42 ;IS PROGRAM UNDER MONITOR CONTROL
1294 002134 001040      BNE      3$
1295 002136 104414      CKSWR ;CHECK FOR <IG>
1296 002140 032777 000004 177032 BIT      #BIT2,@SWR ;CHECK FOR LOCK ON TEST
1297 002146 001411      BEQ      1$
1298 002150 104402 014724      TYPE      MLOCK
1299 002154 012737 000240 012540 MOV      #NOP,TTST
1300 002162 012737 000240 012542 MOV      #NOP,TTST+2 ;SET UP TO LOCK
1301 002170 000406      BR      2$
1302 002172 013737 012636 012540 1$: MOV      BRW,TTST
1303 002200 013737 012640 012542 MOV      BRX,TTST+2 ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
1304 002206 032777 000002 176764 2$: BIT      #SW01,@SWR ;IF SW01=1, GET STARTING PC
1305 002214 001410      BEQ      3$
1306 002216 104403      INSTR
1307 002220 014712      MTSTPC
1308 002222 104405      PARAM
1309 002224 002254      TST1
1310 002226 007450      TLAST
1311 002230 001214      #RETURN
1312 002232 001      .BYTE 1
1313 002233 001      .BYTE 1
1314 002234 000403      BR      4$
1315 002236 012737 002254 001214 3$: MOV      #TST1,RETURN ;START AT TEST 1
1316 002244 104402 014614 4$: TYPE      MR ;TYPE R
1317 002250 000177 176740      JMP      @RETURN ;START TESTING
1318
1319 ; TEST 1
1320 002254 012737 000001 001226 *****
1321 002262 012737 002646 001214 TST1: MOV      #1,TSTNO
1322 002270 012737 002646 001216 MOV      #TST2,RETURN
1323 002276 105737 001302      MOV      #TST2,NEXT
1324 002302 001010      TSTB      RUNFLG ;IS THIS MY FIRST TIME HERE?
1325 002304 012737 000001 001304 BNE      1$ ;BR IF FLAG IS SET
1326 002312 012737 000020 001306 MOV      #BIT0,RUN ;SET RUN POINTER.
1327 002320 105137 001302      COMB      RUNFLG ;SET FOR MAX OF 16 DQ11'S PER SYSTEM
;SET RUN FLAG

```





1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433

: 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

: \*\*\*\*\*

```

1391 002646 012737 000002 001226 TST2: MOV #2,TSTNO
1392 002654 012737 003226 001216 MOV #TST3,NEXT
1394 002662 032737 100000 001510 CKSYN1: BIT #SYNBIT,DQSTAT ;ADJUST POINTER FOR NUMBER OF SYNC CHARS.
1395 002670 001003 BNE 1$ ;BR IF TWO SYNC CHARS SELECTED.
1396 002672 105037 015352 CLR SYNC ;SET FIRST SYNC TO 0 IF ONE SYNC SEL.
1397 002676 000403 BR CKDN ;BR TO CONT.
1398 002700 112737 000026 015352 1$: MOV #26,SYNC ;RESET FIRST SYNC TO 26
1399 002706 000240 CKDN: NOP ;CONTINUE TEST.
1401 002710 032737 040000 001510 BIT #JUMBIT,DQSTAT ;DOES THE TEST JUMPER EXIST.
1402 002716 001005 BNE .+14 ;BR IF YES
1403 002720 013737 001216 001214 MOV NEXT,RETURN ;PREPARE TO DO NEXT TEST.
1404 002726 000177 176262 JMP @RETURN ;GOTO NEXT TEST
1405 002732 005000 CLR R0 ;ZERO DATA POINTER.
1406 002734 012704 015354 MOV #TXBUFF,R4 ;SET BUFFER POINTER
1407 002740 110024 1$: MOV R0,(R4)+ ;FILL TX BUFFER WITH BINARY COUNT PATTERN
1408 002742 105200 INCB R0 ;UPDATE CHAR.
1409 002744 001375 BNE 1$ ;BR IF MORE TO DO.
1410 002746 104413 2$: MEMCLR ;CLEAR DQ11 MEMORIES.
1411 002750 005000 CLR R0 ;ZERO COUNTER POINTER
1412 002752 012704 015756 MOV #RXBUFF,R4 ;PREPARE TO ZERO ALL RX BUFFER.
1413 002756 105024 3$: CLRB (R4)+ ;START CLEARING.
1414 002760 105200 INCB R0 ;UPDATE
1415 002762 001375 BNE 3$ ;BR IF NOT ALL CLEARED.
1416 002764 105077 176400 CLRB @DQREG ;SELECT RX BA PRI.
1417 002770 012777 015756 176374 MOV #RXBUFF,@DQSEC ;LOAD IT.
1418 002776 105277 176366 INCB @DQREG ;SELECT RX WC PRI.
1419 003002 012777 177400 176362 MOV #-400,@DQSEC ;LOAD IT FOR 400(8) CHARS.
1420 003010 105277 176354 INCB @DQREG ;SEL TX BA PRI.
1421 003014 012777 015352 176350 MOV #SYNC,@DQSEC ;LOAD IT.
1422 003022 105277 176342 INCB @DQREG ;SEL TX WC PR.
1423 003026 012777 177376 176336 MOV #-402,@DQSEC ;SET 400(8) CHARS AND TWO SYNC.
1424 003034 112777 000011 176326 MOV #11,@DQREG ;SEL SYNC REGISTER
1425 003042 013777 015350 176322 MOV #SYNC,@DQSEC ;LOAD IT.
1426 003050 105277 176314 INCB @DQREG ;GET MISC REGISTER
1427 003054 012777 004000 176310 MOV #4000,@DQSEC ;SET FOR EIGHTBITS.
1428 003062 005037 001244 CLR TEMP1 ;SET DELAY.....
1429 003066 012737 000020 001246 MOV #20,TEMP2
1430 003074 005277 176260 INC @DQCSR ;SET RX GO!!
1431 003100 005277 176260 INC @DQTCR ;SET TX GO!!
1432 003104 005777 176256 4$: TST @DQERR ;ANY ERRORS
1433 003110 100007 BPL 7$ ;BR IF NO ERRORS

```

# E03

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 30  
 DZDQED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1434 003112 017700 176242      MOV      @DQRCR,RO
1435 003116 017701 176242      MOV      @DQTCR,R1
1436 003122 017702 176240      MOV      @DQERR,R2
1437 003126 104007              HLT      7
1438 003130 105777 176224      7S:    TSTB   @DQRCR
1439 003134 100407              BMI     5S
1440 003136 005237 001244      INC     TEMP1
1441 003142 001360              BNE     4S
1442 003144 005337 001246      DEC     TEMP2
1443 003150 001355              BNE     4S
1444 003152 104000              HLT
1445 003154 005000              5S:    CLR     RO
1446 003156 005037 001252      CLR     TEMP4
1447 003162 005037 001254      CLR     TEMP5
1448 003166 012704 015354      MOV     #TXBUFF,R4
1449 003172 012705 015756      MOV     #RXBUFF,R5
1450 003176 112437 001254      6S:    MOVB   (R4)+,TEMP5
1451 003202 112537 001252      MOVB   (R5)+,TEMP4
1452 003206 023737 001254 001252      CMP     TEMP5,TEMP4
1453 003214 001401              BEQ     .+4
1454 003216 104004              HLT     4
1455 003220 105200              INCB   RO
1456 003222 001365              BNE     6S
1457 003224 104400              SCOPE
  
```

```

;THE DQ11 ERROR FLAG IS SET.
;IS RX PRI DONE SET?
;BR IF YES
;DELAY.....
" " "
RX PRI. DONE FAILED TO SET.
ZERO COUNTER.
CLEAR STORAGE
SAME..
GET TX BUFFER AREA
GET RX BUFFER AREA
LOAD FOR ERROR CALL
DOES DATA CHECK OUT OK?
BR IF GOOD DATA.
DATA COMPARISON ERROR.
UPDATE COUNTER
BR IF MORE DATA TO CHECK.
SCOPE THE TEST.
  
```

```

1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
  
```

```

;RECEIVER STRIP SYNC TEST.
;TEST 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.
  
```

```

1475 003226 012737 000003 001226      ; TEST 3
1476 003234 012737 003520 001216      ;*****
1477 003242 005000              TST3:  MOV     #3,TSTNO
1478 003244 012704 015756              MOV     #TST4,NEXT
1479 003250 105024              CLR     RO
1480 003252 105200              MOV     #RXBUFF,R4
1481 003254 001375              1S:    CLRB   (R4)+
1482 003256 005000              INCB   RO
1483 003260 012704 015354              BNE     1S
1484 003264 105024              CLR     RO
1485 003266 105200              MOV     #TXBUFF,R4
1486 003270 001375              8S:    CLRB   (R4)+
1487
1488 003272 005000              INCB   RO
1489 003274 012704 015354              BNE     8S
  
```

```

;CLEAR POINTER
;SET THE RX BUFFER
;BEGIN TO CLEAR THE RX BUFFER
;ALL DONE?
;BR IF NO
;RESET RO TO ZERO
;GET TX BUFFER
;BEGIN TO CLEAR THE TX BUFFER
;DONE YET?
;BR IF NO
;RESET RO
;GET TX BUFFER
  
```



# F03

DZDGE MACY11 27(1006) 22-DEC-76 11:32 PAGE 31  
 DZDGED.P11 16-DEC-76 13:29

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1490	003300	110024			7\$:	MOVB	RO, (R4)+	; START FILLING WITH CHARS.
1491	003302	105200				INCB	RO	; UPDATE POINTER
1492	003304	022700	000026			CMP	#26, RO	; 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	RO, (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	2DQREG	; SEL RX BA PRI.
1501	003332	012777	015756	176032		MOV	#RXBUFF, 2DQSEC	; SET BUFFER.
1502	003340	105277	176024			INCB	2DQREG	; RX MC PRI.
1503	003344	012777	177400	176020		MOV	#-400, 2DQSEC	; 256. CHARS
1504	003352	105277	176012			INCB	2DQREG	; TX BA PRI.
1505	003356	012777	015352	176006		MOV	#SYNC, 2DQSEC	; SET TO XMIT SYNC CHARS.
1506	003364	105277	176000			INCB	2DQREG	; TX MC PRI.
1507	003370	012777	177400	175774		MOV	#-400, 2DQSEC	; 256. CHARS
1508	003376	112777	000011	175764		MOVB	#11, 2DQREG	; SYNC REGISTER
1509	003404	013777	015350	175760		MOV	. SYNC, 2DQSEC	; LOAD SYNC REGISTER
1510	003412	105277	175752			INCB	2DQREG	; MISC. REGISTER.
1511	003416	012777	004010	175746		MOV	#4010, 2DQSEC	; EIGHT BITS AND TEST LOOP
1512	003424	012777	000003	175726		MOV	#00003, 2DQRCR	; GO AND STRIP SYNC.
1513	003432	005277	175726			INC	2DQTCR	; SET TX GO.
1514	003436	005000				CLR	RO	; PREPARE TO DELAY.
1515	003440	005001				CLR	R1	; SAME
1516	003442	105777	175712		2\$:	TSTB	2DQRCR	; RX DONE?
1517	003446	100406				BMI	9\$	; BR IF YES
1518	003450	062700	000001			ADD	#1, RO	; 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	2DQRCR	; DISABLE RX
1525	003470	005077	175670			CLR	2DQTCR	; DISABLE TX
1526	003474	005000				CLR	RO	; 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	RO	; 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								
1540								
1541								
1542								
1543								
1544								
1545								

; DQ11 MEMORY TRANSFER TESTS  
 ; IF THE PROCESSOR HAS AT LEAST BK OF MEMORY  
 ; THIS TEST WILL BE EXECUTED.  
 ; THE FIRST PART EXERCISES THE TRANSMITTER ALONE  
 ; TRANSMITTING FROM ADD.20000 TO LAST MEMORY ADD.

```

1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558 003520 012737 000004 001226
1559 003526 012737 004456 001216
1560 003534 104413
1561 003536 013705 000004
1562 003542 012737 004432 000004
1563 003550 005737 020000
1564 003554 000240
1565 003556 012737 003606 000004
1566 003564 012700 020000
1567 003570 062700 020000
1568 003574 005710
1569 003576 000240
1570 003600 022700 160000
1571 003604 001371
1572 003606 012706 001200
1573 003612 012737 003706 001220
1574 003620 010537 000004
1575 003624 162700 005000
1576 003630 010037 004454
1577 003634 005002
1578 003636 012700 020000
1579 003642 110220
1580 003644 005202
1581 003646 122702 000026
1582 003652 001001
1583 003654 005202
1584 003656 020037 004454
1585 003662 001367
1586 003664 112777 000002 175476
1587 003672 012777 020000 175472
1588 003700 012700 020000
1589 003704 000406
1590 003706 112777 000002 175454
1591 003714 162777 000002 175450
1592 003722 112777 000002 175440
1593 003730 017737 175436 001252
1594 003736 104412
1595 003740 112777 000002 175422
1596 003746 013777 001252 175416
1597 003754 105277 175410
1598 003760 012777 177776 175404
1599 003766 112777 000012 175374
1600 003774 012777 004012 175370
1601 004002 005277 175356

```

```

: UNDER 28K MINUS 400 (SAVE ABL). THE DATA IS
: VERIFIED ONE CHAR AT A TIME. TWO EIGHT BIT
: CHARS ARE TRANSFERED AT ONE TIME.
:
: AFTER THE TX ALONE THEN THE TRANSMITTER AND RECEIVER
: ARE EXERCISED TOGETHER IN THE SAME
: MANNER AS DESCRIBED ABOVE.
:

```

```

; TEST 4
: *****
TST4: MOV #4, TSTNO
      MOV #TST5, NEXT
      MEMCLR
      MOV #4, R5
      MOV #10$, #4
      TST #20000
      NOP
      MOV #2$, #4
      MOV #20000, R0
1$:  ADD #20000, R0
      TST (R0)
      NOP
      CMP #160000, R0
      BNE 1$
      MOV #STACK, SP
2$:  MOV #4$, LOCK
      MOV R5, #4
      SUB #5000, R0
      MOV R0, LIMIT.HI
      CLR R2
      MOV #20000, R0
3$:  MOVB R2, (R0)+
      INC R2
      CMPB #26, R2
      BNE .+4
      INC R2
      CMP R0, LIMIT.HI
      BNE 3$
      MOVB #2, #DQREG
      MOV #20000, #DQSEC
      MOV #20000, R0
      BR 5$
4$:  MOVB #2, #DQREG
      SUB #2, #DQSEC
5$:  MOVB #2, #DQREG
      MOV #DQSEC, TEMP4
      MSTCLR
      MOVB #2, #DQREG
      MOV TEMP4, #DQSEC
      INCB #DQREG
      MOV #-2, #DQSEC
      MOVB #MISC, #DQREG
      MOV #4012, #DQSEC
      INC #DQTC5R

```

```

: SAVE THE TIME OUT VECTOR
: LOAD TRAP VECTOR
: CHECK FOR BK OF MEMORY.
:
: CPU DOES HAVE AT LEAST BK.
: PREPARE TO SIZE MEMORY TO 28K
: CHECK MEMORY.
: EXIST?
:
: 28K HIT YET.
: BR IF NO
: ADJUST STACK
: SET FOR LOCK (SW09=1)
: RESET TRAP VECTOR.
: ALLOW ROOM FOR DDP2 MONITOR.
: SAVE LAST MEMORY ADDRESS
: ZERO DATA CHAR POINTER
: PREPARE TO FILL MEMORY WITH BINARY COUNT
: START FILL
: UPDATE CHAR.
: DOES IT EQUAL THE SYNC CHAR?
: BR IF NO
: BUMP ONE HIGHER
: IS ALL OF MEMORY FULL?
: BR IF NO.
: SEL TX BA PRI.
: SET TX BA TO FIRST ADD IN BK
: SET SOFTWARE POINTER.
: CONTINUE TEST
: SEL TX BA PRI.
: GO BACKWARDS FOR SCOPE ROUTINE
: SEL TX BA PRI.
: SAVE THE TX BA PRI. ADDRESS
: DO INIT DQ11
: SEL TX BA PRI.
: RELOAD TX BA PRI.
: SEL TX WC PRI.
: SET FOR A TWO EIGHT BIT XFER
: SEL MISC REGISTER
: SET 8 BITS TEST LOOP AND AUTO/STEP
: SET TX GO.

```



# H03

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 33  
 DZDGED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1602	004006	005777	175354		TST	ADQERR	: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,ADQREG	:SEL TX MUX REG
1606	004024	017737	175342	001252	MOV	ADQSEC,TEMP4	:READ TX MUX
1607	004032	011037	001254		MOV	(RO),TEMP5	:READ SOFTWARE POINTER
1608	004036	023710	001252		CMP	TEMP4,(RO)	:IS THE DATA CORRECT
1609	004042	001401			BEG	.+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	(RO)+	:UPDATE SOFTWARE POINTER.
1614	004052	020037	004454		CMP	RO,LIMIT.HI	:ALL DONE?
1615	004056	001321			BNE	55	:BR IF NO
1616							
1617							
1618							
1619							
1620							
1621							
1622							
1623							
1624							
1625							
1626							
1627							
1628	004060	012737	004106	001220	MOV	#65,LOCK	:SET FOR LOCK ON CHAR(SW09=1)
1629	004066	012700	020000		MOV	#20000,RO	:SET FIRT ADD IN BK
1630	004072	112777	000006	175270	MOVB	#6,ADQREG	:SEL TX BA SEC
1631	004100	010077	175266		MOV	RO,ADQSEC	:LOAD TX BA SEC
1632	004104	000413			BR	75	:CONT TEST.
1633	004106	112777	000006	175254	MOVB	#6,ADQREG	:SEL TX BA SEC.
1634	004114	162777	000002	175250	SUB	#2,ADQSEC	:KILL LAST XFER
1635	004122	105077	175242		CLRB	ADQREG	:SEL RX BA PRI.
1636	004126	162777	000002	175236	SUB	#2,ADQSEC	:KILL LAST XFER
1637	004134	112777	000006	175226	MOVB	#6,ADQREG	:SEL TX BA SEC
1638	004142	017737	175224	001252	MOV	ADQSEC,TEMP4	:SAVE IT
1639	004150	104412			MSTCLR		:INIT DQ11
1640	004152	105077	175212		CLRB	ADQREG	:SEL RX BA PRI
1641	004156	013777	001252	175206	MOV	TEMP4,ADQSEC	:LOAD IT
1642	004164	062777	000400	175200	ADD	#400,ADQSEC	:UPDATE IT
1643	004172	105277	175172		INCB	ADQREG	:SEL RX WC PRI
1644	004176	012777	177776	175166	MOV	#-2,ADQSEC	:SET FOR TWO CHARS.
1645	004204	105277	175160		INCB	ADQREG	:SEL TX BA PRI
1646	004210	012777	015352	175154	MOV	#SYNC,ADQSEC	:TX SYNC ON PRI.
1647	004216	105277	175146		INCB	ADQREG	:SEL TX WC PRI.
1648	004222	012777	177776	175142	MOV	#-2,ADQSEC	:SET FOR TWO SYNC.
1649	004230	112777	000006	175132	MOVB	#6,ADQREG	:LOAD TX BA SEC.
1650	004236	013777	001252	175126	MOV	TEMP4,ADQSEC	:LOAD IT
1651	004244	105277	175120		INCB	ADQREG	:SEL TX WC SEC
1652	004250	012777	177776	175114	MOV	#-2,ADQSEC	:SET FOR TWO CHAR XFER
1653	004256	112777	000011	175104	MOVB	#11,ADQREG	:SEL THE SYNC REG
1654	004264	013777	015350	175100	MOV	.SYNC,ADQSEC	:LOAD IT
1655	004272	105277	175072		INCB	ADQREG	:SEL MISC REG.
1656	004276	012777	004010	175066	MOV	#4010,ADQSEC	:EIGHT BITS TEST LOOP
1657	004304	005277	175050		INC	ADQRCSR	:SET RX GO..

:THE ABOVE WAS FOR THE TX ONLY

:THE BELOW ROUTINE EXECISES BOTH THE  
 :TX AND RX TOGETHER.

:NOTE THAT THE RX CA SHOULD BE  
 :400 (8) LOCATIONS HIGHER THAN THE TX CA.

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 34  
 DZDQED.P11 16-DEC-76 13:29

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1658	004310	005277	175050		INC	3DQTCR	:SET TX GO.
1659	004314	005037	001244		CLR	TEMP1	:DELAY COUNTER
1660	004320	012737	000005	001246	MOV	#5,TEMP2	:"" " "
1661	004326	105777	175026		8\$: TSTB	3DQRCSR	: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	3DQERR	: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	3DQREG	:SELECT RX BA PRI.
1673	004372	017701	174774		MOV	3DQSEC,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	3RETURN	:" " "
1689	004452	104400			11\$: SCOPE		:SCOPE THIS TEST
1690							
1691	004454				LIMIT.HI:		
1692	004454	000000			0		
1693							



1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749

004456 012737 000005 001226  
004464 012737 005000 001216  
004472 032737 002000 001510  
004500 001005  
004502 012737 012342 001214  
004510 000177 174500  
004514 104412  
004516 104412  
004520 012737 000351 004776  
004526 112777 000017 174634  
004534 012777 000200 174630  
004542 012737 000011 015330  
004550 112777 000002 174612  
004556 012777 004776 174606  
004564 112777 000123 174576  
004572 012777 177777 174572  
004600 112777 000067 174562  
004606 005077 174560  
004612 112777 000012 174550  
004620 012777 004012 174544  
004626 005277 174532  
004632 027777 174526 174524  
004640 027777 174520 174516  
004646 027777 174512 174510  
004654 005277 174512  
004660 005377 174506  
004664 005277 174502  
004670 005377 174476  
004674 005337 015330  
004700 001371  
004702 112777 000016 174460  
004710 017705 174456  
004714 022705 000351  
004720 001401  
004722 104000  
004724 112777 000012 174436  
004732 012737 000010 015330  
004740 005277 174426  
004744 005377 174422  
004750 005337 015330  
004754 001371  
004756 112777 000016 174404

: TEST OF "ENTER T" AND "EXIT T"  
: TRANSMITTER TRIGGERED  
: TEST TO TRANSMIT ONE CHARACTER ENTERING T  
: CHECKING THE BCC THEN OVERFLOWING  
: CAUSING AN EXIT T THEN MAKING SURE  
: THAT THE BCC WENT TO ZERO.  
: NOTE: IF THE BCC DOES NOT EXIST THESE TESTS WILL NOT BE DONE  
: TEST 5  
: \*\*\*\*\*  
TST5: MOV #5, TSTNO  
MOV #TST6, NEXT  
BIT #ABBIT, DQSTAT  
BNE .+14  
MOV #.EOP, RETURN  
JMP @RETURN  
MSTCLR  
MSTCLR  
MOV #351, WORD  
MOVB #17, @DQREG  
MOV #200, @DQSEC  
MOV #11, COUNT  
MOVB #2, @DQREG  
MOV #WORD, @DQSEC  
MOVB #123, @DQREG  
MOV #-1, @DQSEC  
MOVB #67, @DQREG  
CLR @DQSEC  
MOVB #MISC., @DQREG  
MOV #4012, @DQSEC  
INC @DQTCR  
CMP @DQTCR, @DQTCR  
CMP @DQTCR, @DQTCR  
CMP @DQTCR, @DQTCR  
INC @DQSEC  
DEC @DQSEC  
15: INC @DQSEC  
DEC @DQSEC  
DEC COUNT  
BNE 15  
MOVB #16, @DQREG  
MOV @DQSEC, R5  
CMP #351, R5  
BEQ .+4  
HLT  
1743: MOVB #MISC., @DQREG  
MOV #10, COUNT  
25: INC @DQSEC  
DEC @DQSEC  
DEC COUNT  
BNE 25  
MOVB #16, @DQREG  
: ISSUE A MASTER CLEAR  
: SET CHAR  
: SELECT POLY REGISTER  
: SET POLY FOR LRC 8  
: SET COUNT TO 11  
: SET TX BA PRI  
: LOAD TX BA  
: SEL TX CC-WRITE EN ENTER T  
: SEL TX CC TO -1  
: SEL TX CC SEC-WRITE EN EXIT T  
: SET TX CC TO ZERO  
: SEL MISC REG  
: EIGHT BITS TEST LOOP AUTO SET  
: SET TX GO  
: WAIT TIME.  
: WAIT TIME  
: WAIT TIME  
: CLOCK UP  
: CLOCK DOWN  
: START THE CHAR UP  
DOWN  
: DONE YET?  
: BR IF NO  
: SET TX BCC REG  
: STORE IT  
: DID CHAR GET INTO BCC  
: BR IF YES  
: TX BCC FAILED  
: SEL MISC REG  
: SET COUNT TO 10  
: START CLOCKING BCC OUT UP  
DOWN  
: DONE YET  
: BR IF NO  
: SEL TX BCC

K03

1750 004764 005777 174402  
1751 004770 001401  
1752 004772 104000  
1753 004774 104400  
1754 004776 000000

TST @DQSEC ; DID BCC GET SHIFTED OUT?  
BEQ .+4 ; BR IF YES  
HLT ; TX BCC NOT ZERO  
SCOPE ; SCOPE TEST  
WORD: 0

1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771

; TEST TO FORCE  
; RECEIVER BCC ERROR  
; THE TRANSMITTER CHARACTER COUNT  
; WILL BE SET TO -300  
; AND THE RECEIVER CHARACTER COUNT  
; WILL BE SET TO -400  
; THUS THE RECEIVER WILL RECEIVE  
; MORE CHARACTERS THAN THE TRANSMITTER  
; TRANSMITTED. \*BCC ERROR\*

1772  
1773 005000 012737 000006 001226  
1774 005006 012737 005040 001214  
1775 005014 012737 005106 001216  
1776 005022 104413  
1777 005024 005000  
1778 005026 012704 015354  
1779 005032 110024  
1780 005034 105200  
1781 005036 001375  
1782 005040 104412  
1783 005042 012737 120001 015326  
1784 005050 004537 010546  
1785 005054 177400  
1786 005056 177500  
1787 005060 017705 174302  
1788 005064 005705  
1789 005066 100401  
1790 005070 104001  
1791 005072 032777 000100 174266  
1792 005100 001001  
1793 005102 104001  
1794 005104 104400

; TEST 6  
; \*\*\*\*\*  
TST6: MOV #6, TSTNO  
MOV #2, RETURN  
MOV #TST7, NEXT  
MEMCLR ; CLEAR THE DEVICE  
CLR R0 ; SET R0 TO ZERO  
MOV #TXBUFF, R4 ; SET POINTER FOR BUFFER  
1S: MOV R0, (R4) ; START FILLING THE BUFFER  
INCB R0 ; UPDATE THE DATA  
BNE 1S ; HAS THE BUFFER BEEN FILLED  
2S: MSTCLR ; SELECT CRC 16 FOR POLYNOMIAL  
MOV #120001, XPOLY ; GO PRIM THE DQ11  
JSR RS, SYNBC ; THIS IS THE CHARACTER COUNT FOR THE RECEIVER  
-400 ; THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER  
-300 ; SAVE THE DQ ERROR REGISTER  
MOV @DQERR, RS ; DID AN ERROR OCCUR??  
TST RS ; BR IF THE ERROR DID OCCUR  
BMI .+4 ; HALT THE DQ ERROR BIT IS NOT SET  
HLT 1 ; MAKE SURE IT WAS A RX BCC THAT CAUSED THE ERROR  
BIT #BIT6, @DQERR ; BR IF THE RX BCC BIT IS SET  
BNE .+4 ; RX BCC ERROR BIT NOT SET  
HLT 1 ; SCOPE THIS TEST  
SCOPE

1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805

; TEST OF TRANSMITTER BCC  
; WITH POLYNOMIAL EQUAL TO 177777  
; A FOUR HUNDRED BINARY COUNT  
; DATA PATTERN IS RUN THROUGH  
; THE BCC WITH A SHIFT BY SHIFT  
; CHECK OF THE HARDWARE BY THE SOFTWARE.  
; AT THE END THE TRANSMITTER IS ALSO





# M03

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 38  
 DZDQED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1862	005434	000241			CLC				: CLEAR THE CARRY BIT OF PSW
1863	005436	006037	005364		ROR	65			: 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	45			: BR IF CHAR NOT DONE
1867	005456	105237	015332		INCB	DATA			: GET NEW CHAR
1868	005462	001325			BNE	35			: HAVE ALL CHARACTERS BEEN DONE
1869					.				
1870	005464	005037	015330		CLR	COUNT			: INIT COUNT
1871	005470	112777	000012	173672	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	85			: 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	104000			HLT				: TX BCC NOT ZERO
1891	005612	104400			SCOPE				: SCOPE TEST

1892  
 1893  
 1894  
 1895  
 1896  
 1897  
 1898  
 1899  
 1900  
 1901  
 1902  
 1903  
 1904  
 1905  
 1906  
 1907  
 1908  
 1909  
 1910

```

: TEST OF RECEIVER BCC
: WITH POLYNOMIAL EQUAL TO 177777
:
: A FOUR HUNDRED BINARY COUNT
: DATA PATTERN IS RUN THROUGH
: THE BCC WITH A SHIFT BY SHIFT
: CHECK OF THE HARDWARE COMPARE
: WITH THE SOFTWARE.
: NOTE THERE IS ONE CHARACTER TIME DELAY
: FOR THE BCC TO START.
: ALSO THE IS ONE PAD CHAR
: NEEDED AT THE END OF THE DATA.

```

1911  
 1912  
 1913  
 1914  
 1915  
 1916  
 1917

```

: TEST I0
: *****
: TST10: MOV #10, TSTNO
: MOV #25, RETURN
: MOV #TST11, NEXT
15: MEMCLR ; CLEAR THE DEVICE
: MOV #177777, XPOLY ; SELXPOLY TO 177777
: MOV #15, TEMPS ; SET TYPE OUT REG TO 15
25: 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, 20QREG		;	WRITE EN, RX BA PRI
1922	005676	012777	015756	173466	MOV	#RXBUFF, 20QSEC		;	LOAD THE RX BA
1923	005704	112777	000121	173456	MOVB	#121, 20QREG		;	ENTER "T" WRITE EN, RX CC PRI
1924	005712	012777	177400	173452	MOV	#-400, 20QSEC		;	SET FOR FOUR HUNDRED CHARS
1925	005720	112777	000024	173442	MOVB	#24, 20QREG		;	SEL THE RX BA SEC
1926	005726	012777	015756	173436	MOV	#RXBUFF, 20QSEC		;	SET FOR THE PAD CHAR.
1927	005734	112777	000065	173426	MOVB	#65, 20QREG		;	EXIT "T" WRITE EN, RX CC SEC
1928	005742	012777	177777	173422	MOV	#-1, 20QSEC		;	SEL RX CC SEC FOR ONE PAD CHAR.
1929	005750	112777	000017	173412	MOVB	#17, 20QREG		;	SEL THE POLYNO REGISTER
1930	005756	013777	015326	173406	MOV	XPOLY, 20QSEC		;	LOAD IT WITH THE POLY
1931	005764	112777	000012	173376	MOVB	#MISC., 20QREG		;	SEL THE MISC REGISTER
1932	005772	012777	004012	173372	MOV	#4012, 20QSEC		;	EIGHT BITS TEST LOOP AND AUTO STEP
1933	006000	012777	010001	173352	MOV	#10001, 20QRCR		;	SET RX ACTIVE AND GO!!
1934	006006	012737	000010	015330	MOV	#8., COUNT	3\$:	;	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		CLR	TEMP3	4\$:	;	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, 20QSEC		;	CLEAR THE BIT WINDOW
1941	006050	053777	001250	173314	BIS	TEMP3, 20QSEC		;	PLACE DATA ON BIT WINDOW
1942	006056	005277	173310		INC	20QSEC		;	CLOCK UP---
1943	006062	005377	173304		DEC	20QSEC		;	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		CLR	COUNT	5\$:	;	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	MOV	CALBCC, 9\$	6\$:	;	MOV CALCULATED BCC TO SUB ROUTINE USE
1953	006136	004537	012042		JSR	R5, SIMBCC		;	GO AND CALCULATE BCC (SOFTWARE)
1954	006142	000001			1		7\$:	;	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., 20QREG		;	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, 20QSEC		;	CLEAR THE BIT WINDOW
1962	006200	053777	001250	173164	BIS	TEMP3, 20QSEC		;	LOAD THE DATA
1963	006206	005277	173160		INC	20QSEC		;	CLOCK UP---
1964	006212	005377	173154		DEC	20QSEC		;	CLOCK DN---
1965	006216	112777	000015	173144	MOVB	#15, 20QREG		;	SEL RX BCC REGISTER
1966	006224	017737	173142	001252	MOV	20QSEC, 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??

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 40  
 DZDQED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

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	RS,SIMBCC		: GO FINISH THE BCC
1981	006322	000010				B.			: SHIFTS REQUIRED
1982	006324	000377				377			: DATA CHARACTER
1983	006326	000001			21\$:	.BLKW	1		: PREVIOUS 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		MOV	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\$:	MOV	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	015330		12\$:	CLR	COUNT		: INIT COUNT (THIS IS FOR NUMBER OF BITS PER CHAR)
1998	006434	113737	015332	006456		MOV	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	RS,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		MOV	#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		MOV	#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	017705	172536			MOV	@DQERR,RS		: SAVE THE ERROR REGISTER
2028	006630	005705				TST	RS		: DID AN ERROR OCCUR??
2029	006632	100001				BPL	.+4		: BR IF NO ERROR



2030 006634 104001  
2031 006636 104400

HLT 1 ;DQ11 ERROR FLAG SET  
SCOPE ;SCOPE THIS TEST

2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039

; TEST OF TRANSMITTER AND RECEIVER  
; BCC WITH A POLYNOMIAL OF  
; CRC 16:  $X^{16}+X^{15}+X^2+1$   
; NOTE: IN THIS TEST IT IS UP TO  
; THE HARDWARE TO DISCOVER  
; AN ERROR IF ONE OCCURS.

2040  
2041  
2042  
2043 006640 012737 000011 001226  
2044 006646 012737 006704 001216  
2045 006654 104412  
2046 006656 012737 120001 015326  
2047 006664 004737 010160  
2048 006670 017705 172472  
2049 006674 005705  
2050 006676 100001  
2051 006700 104001  
2052 006702 104400

; TEST 11  
; \*\*\*\*\*  
TST11: MOV #11,TSTNO  
MOV #TST12,NEXT  
MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR  
MOV #120001,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.  
JSR PC,STBCC ;TRANSFER CHARACTERS.  
MOV @DQERR,R5 ;SAVE THE ERROR REGISTER  
TST R5 ;DID AN ERROR OCCUR??  
BPL .+4 ;BR IF NO ERROR  
HLT 1 ;AN ERROR OCCURED  
SCOPE

2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064

; TEST OF TRANSMITTER AND RECEIVER  
; BCC WITH A POLYNOMIAL OF  
; CRC 12:  $X^{12}+X^{11}+X^3+X^2+X+1$   
; NOTE: IN THIS TEST IT IS UP TO  
; THE HARDWARE TO DISCOVER  
; AN ERROR IF ONE OCCURS

2065  
2066  
2067 006704 012737 000012 001226  
2068 006712 012737 006750 001216  
2069 006720 104412  
2070 006722 012737 007401 015326  
2071 006730 004737 010160  
2072 006734 017705 172426  
2073 006740 005705  
2074 006742 100001  
2075 006744 104001  
2076 006746 104400

; TEST 12  
; \*\*\*\*\*  
TST12: MOV #12,TSTNO  
MOV #TST13,NEXT  
MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR  
MOV #7401,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.  
JSR PC,STBCC ;TRANSFER CHARACTERS.  
MOV @DQERR,R5 ;SAVE THE ERROR REGISTER  
TST R5 ;DID AN ERROR OCCUR??  
BPL .+4 ;BR IF NO ERROR  
HLT 1 ;AN ERROR OCCURED  
SCOPE

2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085

; TEST OF TRANSMITTER AND RECEIVER  
; BCC WITH A POLYNOMIAL OF  
; CRC/CCITT:  $X^{16}+X^{12}+X^2+X+1$   
; NOTE: IN THIS TEST IT IS UP TO

; THE HARDWARE TO DISCOVER  
; AN ERROR IF ONE OCCURES

2086  
2087  
2088  
2089  
2090  
2091 006750 012737 000013 001226  
2092 006756 012737 007014 001216  
2093 006764 104412  
2094 006766 012737 102010 015326  
2095 006774 004737 010160  
2096 007000 017705 172362  
2097 007004 005705  
2098 007006 100001  
2099 007010 104001  
2100 007012 104400

```
; TEST 13
;*****
†ST13: MOV #13,TSTNO
      MOV #TST14,NEXT
      MSTCLR
      MOV #102010,XPOLY ;CLEAR DQ11 WITH A MASTER CLEAR
      JSR PC,STBCC ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
      MOV DDERR,RS ;TRANSFER CHARACTERS.
      TST RS ;SAVE THE ERROR REGISTER
      BPL +4 ;DID AN ERROR OCCUR??
      HLT 1 ;BR IF NO ERROR
      SCOPE ;AN ERROR OCCURED
```

; 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.

2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115 007014 012737 000014 001226  
2116 007022 012737 007060 001216  
2117 007030 104412  
2118 007032 012737 000200 015326  
2119 007040 004737 010160  
2120 007044 017705 172316  
2121 007050 005705  
2122 007052 100001  
2123 007054 104001  
2124 007056 104400

```
; TEST 14
;*****
†ST14: MOV #14,TSTNO
      MOV #TST15,NEXT
      MSTCLR
      MOV #200,XPOLY ;CLEAR DQ11 WITH A MASTER CLEAR
      JSR PC,STBCC ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
      MOV DDERR,RS ;TRANSFER CHARACTERS.
      TST RS ;SAVE THE ERROR REGISTER
      BPL +4 ;DID AN ERROR OCCUR??
      HLT 1 ;BR IF NO ERROR
      SCOPE ;AN ERROR OCCURED
```

; 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

2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139 007060 012737 000015 001226  
2140 007066 012737 007124 001216  
2141 007074 104412

```
; TEST 15
;*****
†ST15: MOV #15,TSTNO
      MOV #TST16,NEXT
      MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR
```



E04

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 43  
DZDQED.P11 16-DEC-76 13:29

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

2142 007076 012737 100000 015326  
2143 007104 004737 010160  
2144 007110 017705 172252  
2145 007114 005705  
2146 007116 100001  
2147 007120 104001  
2148 007122 104400

MOV #100000,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.  
JSR PC,STBCC ;TRANSFER CHARACTERS.  
MOV @DQERR,R5 ;SAVE THE ERROR REGISTER  
TST R5 ;DID AN ERROR OCCUR??  
BPL :+4 ;BR IF NO ERROR  
HLT i ;AN ERROR OCCURED  
SCOPE

2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164

; TEST OF RECEIVER AND TRANSMITTER  
; BCC USING CRC 16 FOR POLYNOMIAL.  
; THIS TEST USES IDLE MODE TO  
; GET INTO TRANSPARENCY  
; AND IF AN ERROR SHOULD OCCUR  
; IT MUST BE REPORTED BY THE HARDWARE.

2165 007124 012737 000016 001226  
2166 007132 012737 007164 001214  
2167 007140 012737 007220 001216  
2168 007146 104413  
2169 007150 005000  
2170 007152 012704 015354  
2171 007156 110024  
2172 007160 105200  
2173 007162 001375  
2174 007164 104412  
2175 007166 012737 120001 015326  
2176 007174 004537 010546  
2177 007200 177400  
2178 007202 177400  
2179 007204 017705 172156  
2180 007210 005705  
2181 007212 100001  
2182 007214 104001  
2183 007216 104400

; TEST 16  
; \*\*\*\*\*  
TST16: MOV #16,TSTNO  
MOV #25,RETURN  
MOV #TST17,NEXT  
MEMCLR ;CLEAR THE DEVICE  
CLR R0 ;SET R0 TO ZERO  
MOV #TXBUFF,R4 ;SET POINTER FOR BUFFER  
15: MOVB R0,(R4)+ ;START FILLING THE BUFFER  
INCB R0 ;UPDATE THE DATA  
BNE 15 ;HAS THE BUFFER BEEN FILLED  
25: MSTCLR  
MOV #120001,XPOLY ;SELECT CRC 16 FOR POLYNOMIAL  
JSR R5,SYNBCC ;GO PRIM THE DQ11  
-400 ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER  
-400 ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER  
MOV @DQERR,R5 ;SAVE THE DQ ERROR REGISTER  
TST R5 ;DID AN ERROR OCCUR??  
BPL :+4 ;BR IF THE ERROR DID NOT OCCUR  
HLT i ;HALT THE DQ ERROR BIT IS SET  
SCOPE ;SCOPE THIS TEST

2184  
2185  
2186  
2187  
2188  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197

; TEST OF TRANSMITTER AND RECEIVER  
; BCC WITH A POLYNOMIAL OF 177777  
; THIS TEST USES IDLE MODE TO  
; GET INTO TRANSPARENCY. IF AN  
; ERROR SHOULD HAPPEN, THE HARDWARE  
; MUST FLAG IT.

; TEST 17  
; \*\*\*\*\*

# F04

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 44  
 DZDQED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2198 007220 012737 000017 001226 TST17: MOV #17,TSTNO
2199 007226 012737 007260 001214 MOV #25,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 1S: MOVB RO,(R4)+ ;START FILLING THE BUFFER
2205 007254 105200 INCB RO ;UPDATE THE DATA
2206 007256 001375 BNE 1S ;HAS THE BUFFER BEEN FILLED
2207 007260 104412 2S: MSTCLR ;ISSUE MASTER CLEAR
2208 007262 012737 177777 015326 MOV #177777,XPOLY ;SELECT 177777 FOR POLYNOMIAL
2209 007270 004537 010546 JSR RS,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 ;
2220 ;TEST OF TRANSMITTER AND RECEIVER
2221 ;BCC WITH ALL POLYNOMIALS
2222 ;BETWEEN 000000 AND 177777
2223 ;SENDING ONE CHARACTER (351)
2224 ;AND CALCULATING WHAT THE BCC
2225 ;SHOULD BE AND COMPARING IT
2226 ;WITH THE TRANSMITTER BCC AND RECEIVER BCC
2227 ;
2228 ;NOTE: SW 09=1 WILL FREEZE THE POLYNOMIAL
2229 ;WHEN SW 09 IS ASSERTED.
2230
2231 ; TEST 20
2232 ;*****
2232 007314 012737 000020 001226 TST20: MOV #20,TSTNO
2233 007322 012737 000003 001222 MOV #3,COUNT
2234 007330 012737 007450 001216 MOV #TST21,NEXT
2235 007336 012737 007350 001220 MOV #15,LOCK
2236 007344 005037 015326 CLR XPOLY ;INIT SET XPOLY TO ZERO
2237 007350 104412 1S: MSTCLR
2238 007352 004737 011066 JSR PC,TYBCC ;GO AND PRIM THE DQ11
2239 007356 004537 012042 JSR RS,SIMBCC ;GO AND CALCULATE WHAT THE BCC SHOULD BE
2240 007362 000010 B. ;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

```



```

2254 007436 104401          SCOPI          ; DOES THE USER HAVE SW09=1??
2255 007440 005237 015326  INC          XPOLY        ; UPDATE THE POLYNO
2256 007444 001341          BNE          1$          ; BE IF NOT ALL POLYNO HAVE BEEN DONE
2257 007446 104400          SCOPE          ; SCOPE THIS TEST
    
```

```

2258
2259
2260
2261          ; TEST OF BIT 06 OF MISC REGISTER
2262          ; POLYNOMIAL 16-24
2263
2264          ; TEST WILL SEND ONE CHARACTER AT
2265          ; A TIME CHECKING THAT THE BCC
2266          ; OF RECEIVER AND TRANSMITTER
2267          ; ARE CORRECT.
2268          ; CHARACTER SENT: 000-377
2269
2270
    
```

```

2271          ; TEST 21
2272          ; *****
2273 007450 012737 000021 001226  TST21: MOV      #21,TSTNO
2274 007456 012737 012342 001216  MOV      #.EOP,NEXT
2275          ; PART 1 READ/WRITE TEST OF POLY 16-24
2276          ; TEST OF ALL READ WRITE BITS IN POLY 16-24
2277          ; BY RUNNING A BINARY COUNT PATTERN TROUGH
2278          ; THE REGISTER.
2279
    
```

```

2281 007464 012737 007476 001220  MOV      #1$,LOCK        ; SET FOR LOCK ON TEST(SW09=1)
2282 007472 005037 001254          CLR      TEMPS          ; ZERO POINTER
2283 007476 104412          MSTCLR        ; INIT DQ11
2284 007500 112777 000012 171662  1$:  MOVB    #MISC.,DQREG    ; SEL MISC REG
2285 007506 012777 000100 171656  MOV      #BIT6,DQSEC     ; SEL POLY 16-24
2286 007514 112777 000017 171646  MOVB    #17,DQREG       ; SEL POLY REGISTER
2287 007522 053777 001254 171642  BIS     TEMPS,DQSEC     ; LOAD WITH CHAR.
2288 007530 017737 171636 001252  MOV     DQSEC,TEMP4     ; READ CHAR BACK.
2289 007536 023737 001254 001252  CMP     TEMPS,TEMP4     ; IS IT CORRECT?
2290 007544 001401          BEQ     .+4             ; BR IF YES.
2291 007546 104006          HLT     6               ; POLY READ/WRITE ERROR.
2292 007550 104401          SCOPI          ; LOCK ON CHAR (SW09=1)
2293 007552 105237 001254          INCB    TEMPS          ; UPDATE CHAR.
2294 007556 001347          BNE     1$             ; BR IF MORE TO GO
    
```

```

2295
2296          ; PART 2 RX AND TX BCC TESTS
2297
2298 007560 005037 015322          CLR      CALBCC        ; ZERO EXPECTED BCC
2299 007564 012737 007572 001220  2$:  MOV      #2$,LOCK     ; SET FOR SW09=1
2300 007572 104412          MSTCLR        ; INIT DQ11
2301 007574 012737 000200 015326  MOV     #200,XPOLY     ; SEL "LRC 24"
2302 007602 004737 011434          JSR     PC,TXBCC      ; GOTO SUBROUTINE
2303 007606 012705 000015          MOV     #1$,R5        ; SEL BCC REG
2304 007612 112777 000015 171550  MOVB    #15,DQREG     ; " "
2305 007620 017701 171546          MOV     DQSEC,R1      ; READ BCC REG.
2306 007624 023701 015322          CMP     CALBCC,R1     ; IS BCC CORRECT?
2307 007630 001401          BEQ     .+4           ; BR IF GOOD.
2308 007632 104002          HLT     2             ; BCC ERROR.
2309 007634 012705 000016          MOV     #16,R5        ; SEL BCC REG.
    
```

# H04

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 46  
 DZDQED.P11 16-DEC-76 13:29 MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

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.,ADQREG	;SEL MISC REGISTER
2317	007672	042777	000100	171472	BIC	#BIT6,ADQSEC	;SEL POLY 0-15
2318	007700	012705	000015		MOV	#15,R5	;SET FOR ERROR
2319	007704	110577	171460		MOVB	R5,ADQREG	;SEL BCC REG
2320	007710	017701	171456		MOV	ADQSEC,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.,ADQREG	
2331	007752	005277	171414		INC	ADQSEC	
2332	007756	005377	171410		DEC	ADQSEC	
2333	007762	012705	000016		MOV	#16,R5	
2334	007766	110577	171376		MOVB	R5,ADQREG	
2335	007772	017701	171374		MOV	ADQSEC,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.,ADQREG	
2342	010024	005277	171342		INC	ADQSEC	
2343	010030	005377	171336		DEC	ADQSEC	
2344	010034	005337	015330		DEC	COUNT	
2345	010040	001371			BNE	3\$	
2346	010042	012705	000015		MOV	#15,R5	
2347	010046	110577	171316		MOVB	R5,ADQREG	
2348	010052	017701	171314		MOV	ADQSEC,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	104400			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





MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

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 1S: MOVB R0,(R4)+
2393 010176 105200 INCB R0
2394 010200 001375 BNE 1S
2395 010202 105077 171162 CLR# 20QREG ;SELECT THE RX BA PRI.
2396 010206 012777 015756 171156 MOV #RXBUFF,20QSEC ;LOAD THE RX BA
2397 010214 112777 000121 171146 MOVB #121,20QREG ;ENTER T,WRITE ENABLE,RX CC,PRI.
2398 010222 012777 177400 171142 MOV #-400,20QSEC ;SET RX CC FOR A TRANSFER OF 400 CHARS.
2399 010230 112777 000022 171132 MOVB #22,20QREG ;WRITE ENABLE, TX BA PRI.
2400 010236 012777 015352 171126 MOV #SYNC,20QSEC ;LOAD THE TX BA PRI.
2401 010244 112777 000023 171116 MOVB #23,20QREG ;ENTER T,WRITE ENABLE, TX CC PRI.
2402 010252 012777 177776 171112 MOV #-2,20QSEC ;SET TX CC FOR A TRANSFER OF 2 CHARS.
2403 010260 112777 000024 171102 MOVB #24,20QREG ;WRITE ENABLE, RX BA SEC.
2404 010266 005077 171100 CLR 20QSEC ;CLEAR THE RX BA SEC
2405 010272 112777 000065 171070 MOVB #65,20QREG ;EXIT T,WRITE ENABLE, RX CC SEC.
2406 010300 012777 177777 171064 MOV#-1,20QSEC ;SET THE RX CC SEC FOR ONE PAD CHAR.
2407 010306 112777 000026 171054 MOVB #26,20QREG ;WRITE ENABLE, TX BA SEC.
2408 010314 012777 015354 171050 MOV #TXBUFF,20QSEC
2409 010322 112777 000127 171040 MOVB #127,20QREG ;EXIT T,WRITE ENABLE, TX CC SEC.
2410 010330 012777 177400 171034 MOV #-400,20QSEC
2411 010336 112777 000011 171024 MOVB #11,20QREG
2412 010344 013777 015350 171020 MOV .SYNC,20QSEC
2413 010352 112777 000017 171010 MOVB #17,20QREG
2414 010360 013777 015326 171004 MOV XPOLY,20QSEC
2415 010366 112777 000012 170774 MOVB #MISC,20QREG
2416 010374 012777 004010 170770 MOV #4010,20QSEC
2417 010402 005037 001244 CLR TEMP1
2418 010406 005037 001246 CLR TEMP2
2419 010412 012777 010462 170734 MOV #TXISR1,20QTEVC
2420 010420 005077 170732 CLR 20QTLVL
2421 010424 012777 000001 170726 MOV #1,20QRCR
2422 010432 012777 000041 170724 MOV #41,20QTCR
2423 010440 005037 177776 CLR PS
2424 010444 105237 001244 2S: INCB TEMP1
2425 010450 001375 BNE 2S
2426 010452 105237 001246 INCB TEMP2
2427 010456 001372 BNE 2S
2428 010460 104000 HLT
2429 010462 022626 4.7 TXISR1: CMP (SP)+,(SP)+
2430 010464 042777 000040 170672 8.2 BIC #BITS,20QTCR
2431 010472 112777 000063 170670 7.6 MOVB #63,20QREG
2432 010500 005077 170666 6.1 CLR 20QSEC
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 1S: BIT #BIT6,20QRCR
2437 010524 001007 2.6 BNE ENDBCC
2438 010526 105237 001244 3.7 INCB TEMP1
2439 010532 001371 2.6 BNE 1S
2440 010534 105237 001246 3.7 INCB TEMP2
2441 010540 001366 2.6 BNE 1S
2442 010542 104000 9.3 HLT ;RX SECONDARY DONE NOT SET.
2443 010544 000207 3.5 ENDBCC: RTS PC

```



```

2475 010546 010537 015336 SYNBC: MOV R5,SAVEPC ;SAVE PC OF ENTERING ROUTINE
2476 010552 104412 MSTCLR ;CLEAR THE DQ11
2477 010554 105077 170610 CLR R5 ;SELECT THE RX BA PRI.
2478 010560 012777 015756 170604 MOV #RXBUFF,2DQSEC ;LOAD THE RX BA
2479 010566 112777 000121 170574 MOVB #121,2DQREG ;ENTER T,WRITE ENABLE,RX CC,PRI.
2480 010574 012577 170572 MOV (R5)+,2DQSEC ;SET RX CC
2481 010600 112777 000022 170562 MOVB #22,2DQREG ;WRITE ENABLE, TX BA PRI.
2482 010606 012777 015354 170556 MOV #TXBUFF,2DQSEC ;LOAD THE TX BA PRI.
2483 010614 112777 000123 170546 MOVB #123,2DQREG ;ENTER T,WRITE ENABLE, TX CC PRI.
2484 010622 012577 170544 MOV (R5)+,2DQSEC ;SET TX CC
2485 010626 112777 000024 170534 MOVB #24,2DQREG ;WRITE ENABLE, RX BA SEC.
2486 010634 012777 015756 170530 MOV #RXBUFF,2DQSEC ;LOAD THE RX BA SEC.
2487 010642 112777 000065 170520 MOVB #65,2DQREG ;EXIT T,WRITE ENABLE, RX CC SEC.
2488 010650 012777 177777 170514 MOV #-1,2DQSEC ;SET FOR ONE PAD CHARACTER.
2489 010656 112777 000026 170504 MOVB #26,2DQREG ;WRITE ENABLE, TX BA SEC.
2490 010664 005077 170502 CLR 2DQSEC ;CLEAR THE TX CC SEC
2491 010670 112777 000067 170472 MOVB #67,2DQREG ;EXIT T,WRITE ENABLE, TX CC SEC.
2492 010676 005077 170470 CLR 2DQSEC ;CLEAR THE TX CC SEC
2493 010702 112777 000011 170460 MOVB #11,2DQREG
2494 010710 013777 015350 170454 MOV .SYNC,2DQSEC
2495 010716 112777 000017 170444 MOVB #17,2DQREG
2496 010724 013777 015326 170440 MOV XPOLY,2DQSEC
2497 010732 112777 000012 170430 MOVB #MISC.,2DQREG
2498 010740 012777 004010 170424 MOV #4010,2DQSEC
2499 010746 005037 001244 CLR TEMP1
2500 010752 005037 001246 CLR TEMP2
2501 010756 052777 000003 170374 BIS #3,2DQRCSR
2502 010764 052777 000002 170372 BIS #2,2DQTCSR
2503 010772 032777 020000 170372 25: BIT #BIT13,2DQSEC
2504 011000 001006 BNE 35
2505 011002 005237 001244 INC TEMP1
2506 011006 001371 BNE 25
2507 011010 005237 001246 INC TEMP2
2508 011014 001366 BNE 25
2509 011016 012777 000001 170340 35: MOV #BIT0,2DQTCSR
2510 011024 005037 001244 CLR TEMP1
2511 011030 012737 000005 001246 MOV #5,TEMP2
2512 011036 032777 000100 170314 45: BIT #BIT6,2DQRCSR
2513 011044 001007 BNE ENDSYN
2514 011046 005237 001244 INC TEMP1
2515 011052 001371 BNE 45
2516 011054 005337 001246 DEC TEMP2
2517 011060 001366 BNE 45
2518 011062 104000 HLT ;RECEIVER DONE SECONDARY NOT SET.
2519 011064 000205 ENDSYN: RTS R5

2520 011066 011637 015336 TYBCC: MOV (SP),SAVEPC ;SAVE PC OF ENTERING ROUTINE
2521 011072 104412 MSTCLR ;CLEAR THE DQ11
2522 011074 000240 NOP
2523 011076 104412 MSTCLR
2524 011100 012737 000351 015354 MOV #351, TXBUFF

```

```

2500 011106 012737 000011 015330      MOV      #9, COUNT
2501 011114 105077 170250      1S:     CLR      @DQREG      ;SELECT THE RX BA PRI.
2502 011120 012777 015756 170244      MOV      @RXBUFF, @DQSEC ;LOAD THE RX BA
2503 011126 112777 000121 170234      MOV      #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      MOV      #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      MOV      #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      MOV      #24, @DQREG     ;WRITE ENABLE, RX BA SEC.
2510 011200 005077 170166      CLR      @DQSEC         ;CLEAR THE RX BA SEC
2511 011204 112777 000065 170156      MOV      #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      MOV      #26, @DQREG     ;WRITE ENABLE, TX BA SEC.
2514 011224 005077 170142      CLR      @DQSEC         ;CLEAR THE TX CC SEC
2515 011230 112777 000067 170132      MOV      #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      MOV      #17, @DQREG
2518 011250 013777 015326 170114      MOV      XPOLY, @DQSEC
2519 011256 112777 000012 170104      MOV      #MISC., @DQREG
2520 011264 012777 004012 170100      MOV      #4012, @DQSEC
2521 011272 052777 000001 170064      BIS      #BIT0, @DQTCR   ;SET TRANSMITTER GO
2522 011300 027777 170060 170056      CMP      @DQTCR, @DQTCR ;WAIST TIME.
2523 011306 027777 170052 170050      CMP      @DQTCR, @DQTCR ;WAIST TIME
2524 011314 027777 170044 170042      CMP      @DQTCR, @DQTCR ;WAIST TIME
2525 011322 005277 170044      INC      @DQSEC         ;PRIM THE
2526 011326 005377 170040      DEC      @DQSEC         ;
2527 011332 042777 000200 170032      BIC      #BIT7, @DQSEC   ;CLEAR THE BIT WINDOW.
2528 011340 052777 010001 170012      BIS      #10001, @DQRCR
2529 011346 005277 170020      2S:     INC      @DQSEC
2530 011352 005377 170014      DEC      @DQSEC
2531 011356 005337 015330      DEC      COUNT
2532 011362 001371      BNE      2S
2533 011364 112777 000016 167776      MOV      #16, @DQREG
2534 011372 017737 167774 015340      MOV      @DQSEC, SEC16
2535 011400 112777 000012 167762      MOV      #MISC., @DQREG
2536 011406 012737 000007 015330      MOV      #7, COUNT
2537 011414 005277 167752      3S:     INC      @DQSEC
2538 011420 005377 167746      DEC      @DQSEC
2539 011424 005337 015330      DEC      COUNT
2540 011430 001371      BNE      3S
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      1S:     CLR      @DQREG      ;SELECT THE RX BA PRI.
2552 011462 012777 015756 167702      MOV      @RXBUFF, @DQSEC ;LOAD THE RX BA
2553 011470 112777 000121 167672      MOV      #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      MOV      #22, @DQREG     ;WRITE ENABLE, TX BA PRI.
    
```



2556	011512	012777	015354	167652	MOV	#TXBUFF, 20QSEC	; LOAD THE TX BA PRI.
2557	011520	112777	000123	167642	MOVB	#123, 20QREG	; ENTER T WRITE ENABLE, TX CC PRI.
2558	011526	012777	177777	167636	MOV	#-1, 20QSEC	; SET TX CC FOR A TRANSFER OF 1 CHARACTER.
2559	011534	112777	000024	167626	MOVB	#24, 20QREG	; WRITE ENABLE, RX BA SEC.
2560	011542	005077	167624		CLR	20QSEC	; CLEAR THE RX BA SEC
2561	011546	112777	000065	167614	MOVB	#65, 20QREG	; EXIT T WRITE ENABLE, RX CC SEC.
2562	011554	005077	167612		CLR	20QSEC	; CLEAR THE RX CC SEC.
2563	011560	112777	000026	167602	MOVB	#26, 20QREG	; WRITE ENABLE, TX BA SEC.
2564	011566	005077	167600		CLR	20QSEC	; CLEAR THE TX CC SEC
2565	011572	112777	000067	167570	MOVB	#67, 20QREG	; EXIT T WRITE ENABLE, TX CC SEC.
2566	011600	005077	167566		CLR	20QSEC	; CLEAR THE TX CC SEC
2567	011604	112777	000012	167556	MOVB	#MISC., 20QREG	
2568	011612	012777	004112	167552	MOV	#4112, 20QSEC	
2569	011620	112777	000017	167542	MOVB	#17, 20QREG	
2570	011626	013777	015326	167536	MOV	XPOLY, 20QSEC	
2571	011634	112777	000012	167526	MOVB	#MISC., 20QREG	
2572	011642	052777	000001	167514	BIS	#BIT0, 20QTCR	; SET TRANSMITTER GO
2573	011650	027777	167510	167506	CMP	20QTCR, 20QTCR	; WAIT TIME.
2574	011656	027777	167502	167500	CMP	20QTCR, 20QTCR	; WAIT TIME
2575	011664	027777	167474	167472	CMP	20QTCR, 20QTCR	; WAIT TIME
2576	011672	005277	167474		INC	20QSEC	; PRIM THE
2577	011676	005377	167470		DEC	20QSEC	; TRANSMITTER.
2578	011702	042777	000200	167462	BIC	#BIT7, 20QSEC	; CLEAR THE BIT WINDOW.
2579	011710	052777	010001	167442	BIS	#10001, 20QRCR	
2580	011716	005277	167450		INC	20QSEC	
2581	011722	005377	167444		DEC	20QSEC	
2582	011726	005337	015330		DEC	COUNT	
2583	011732	001371			BNE	25	
2584	011734	042777	000100	167430	BIC	#BIT6, 20QSEC	
2585	011742	112777	000016	167420	MOVB	#16, 20QREG	
2586	011750	017737	167416	015342	MOV	20QSEC, SEC16X	
2587	011756	112777	000012	167404	MOVB	#MISC., 20QREG	
2588	011764	052777	000100	167400	BIS	#BIT6, 20QSEC	
2589	011772	112777	000016	167370	MOVB	#16, 20QREG	
2590	012000	017737	167366	015340	MOV	20QSEC, SEC16	
2591	012006	112777	000012	167354	MOVB	#MISC., 20QREG	
2592	012014	012737	000007	015330	MOV	#7, COUNT	
2593	012022	005277	167344		INC	20QSEC	
2594	012026	005377	167340		DEC	20QSEC	
2595	012032	005337	015330		DEC	COUNT	
2596	012036	001371			BNE	35	
2597	012040	000207			RTS	PC	

25:

35:

ENDTX:

```

2598
2599
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 1S: CLR BCCFBK
2605 012062 013700 001250 MOV TEMP3,RO
2606 012066 006037 001246 ROR TEMP2
2607 012072 005500 ADC RO
2608 012074 032700 000001 BIT #BIT0,RO
2609 012100 001402 BEQ 2S
2610 012102 005137 015324 COM BCCFBK
2611 012106 013700 015326 2S: MOV XPOLY,RO
2612 012112 005100 COM RO
2613 012114 040037 015324 BIC RO,BCCFBK
2614 012120 000241 CLC
2615 012122 006037 001250 ROR TEMP3
2616 012126 013700 015324 MOV BCCFBK,RO
2617 012132 013701 001250 MOV TEMP3,R1
2618 012136 010102 MOV R1,R2
2619 012140 040100 BIC R1,RO
2620 012142 043702 015324 BIC BCCFBK,R2
2621 012146 050200 BIS R2,RO
2622 012150 043737 015326 001250 BIC XPOLY,TEMP3
2623 012156 050037 001250 BIS RO,TEMP3
2624 012162 005337 001244 DEC TEMP1
2625 012166 001333 BNE 1S
2626 012170 013737 001250 015322 MOV TEMP3,CALBCC
2627 012176 000205 RTS R5
2628
2629 012200 .MEMCLR:
2630 012200 005077 167154 CLR @DQCSR
2631 012204 005077 167154 CLR @DQTCR
2632 012210 005077 167152 CLR @DQERR
2633 012214 012705 000020 MOV #16,R5
2634 012220 152777 000020 167142 1S: BISR #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 1S
2640 012250 105077 167114 CLRB @DQREG
2641 012254 105077 167102 CLRB @DQRCRSH
2642 012260 012705 000020 MOV #16,R5
2643 012264 112777 000010 167076 2S: 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 @DQRCRSH
2648 012314 005305 DEC R5
2649 012316 001362 BNE 2S
2650 012320 105077 167036 CLRB @DQRCRSH
2651 012324 .MSTCLR:
2652 012324 112777 000012 167036 MOVB #MISC,@DQREG
2653 012332 012777 000040 167032 MOV #BITS,@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 @QNUM, 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 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 DGRVEC
2707 012514 000001 XPASS: 1
2708 012516 006 002 .BYTE 6,2
2709 012520 001230 PASCNT
    
```

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 54  
 DZDQED.P11 16-DEC-76 13:29 END OF PASS ROUTINE

```

2710 012522 000001          XERR: 1
2711 012524 006          .BYTE 6,2
2712 012526 001232      ERRCNT
2713
2714                                ;SCOPE LOOP AND INTERATION HANDLER
2715
2716 012530 104414      .SCOPE: CKSWR
2717 012532 032777 040000 166440 BIT      #BIT14, @SWR
2718 012540 001407      TTST: BEQ      1$
2719 012542 000432      BR      3$
2720 012544 105777 166434 TSTB    @TKCSR
2721 012550 100027      BPL     3$
2722 012552 017700 166430 MOV     @TKDBR, R0
2723 012556 000412      BR      2$
2724 012560 032777 004000 166412 1$: BIT    #SW11, @SWR
2725 012566 001006      BNE     2$
2726 012570 005237 001224 INC     LPCNT
2727 012574 023737 001224 001222 CMP     LPCNT, ICOUNT
2728 012602 001012      BNE     3$
2729 012604 105037 001312 2$: CLRB  ERRFLG
2730 012610 005037 001224 CLR     LPCNT
2731 012614 012737 000011 001222 MOV     #9, ICOUNT
2732 012622 013737 001216 001214 MOV     NEXT, RETURN
2733 012630 013716 001214 3$: MOV     RETURN, (SP)
2734 012634 000002      RTI
2735 012636 001407      BRW: 1407
2736 012640 000432      BRX: 432
2737
2738                                ;CHECK FOR FREEZE ON CURRENT DATA
2739
2740 012642 104414      .SCOPE1: CKSWR
2741 012644 032777 001000 166326 BIT     #SW09, @SWR
2742 012652 001402      BEQ     1$
2743 012654 013716 001220 MOV     LOCK, (SP)
2744 012660 000002      1$: RTI
2745
2746                                ;TELETYPE OUTPUT ROUTINE
2747
2748 012662 010546      .TYPE: MOV     R5, -(SP)
2749 012664 017605 000002 MOV     @2(SP), R5
2750 012670 062766 000002 000002 ADD     #2, 2(SP)
2751 012676 005737 014352 1$: TST    @#RDSW
2752 012702 001004      BNE     300$
2753 012704 032777 010000 166266 BIT     #SW12, @SWR
2754 012712 001024      BNE     3$
2755 012714 105715 300$: TSTB  (R5)
2756 012716 100014      BPL     2$
2757 012720 105777 166264 TSTB  @TPCSR
2758 012724 100375      BPL     #-4
2759 012726 012777 000015 166256 MOV     #15, @TPDBR
2760 012734 105777 166250 TSTB  @TPCSR
2761 012740 100375      BPL     #-4
2762 012742 012777 000012 166242 MOV     #12, @TPDBR
2763 012750 105777 166234 2$: TSTB  @TPCSR
2764 012754 100375      BPL     2$
2765 012756 112577 166230 MOVB  (R5)+, @TPDBR

```



```

2766 012762 001345
2767 012764 012605
2768 012766 000002
2769
2770
2771
2772 012770 010346
2773 012772 010446
2774 012774 017637 000004 013012
2775 013002 062766 000002 000004
2776 013010 104402
2777 013012 000000
2778 013014 012704 015144
2779 013020 012703 000007
2780 013024 105777 166154
2781 013030 100375
2782 013032 117714 166150
2783 013036 142714 000200
2784 013042 121427 000025
2785 013046 001003
2786 013050 104402 014532
2787 013054 000755
2788 013056 122427 000015
2789 013062 001423
2790 013064 117777 166116 166120
2791 013072 105777 166112
2792 013076 100375
2793 013100 005303
2794 013102 001350
2795 013104 000402
2796 013106 010346
2797 013110 010446
2798 013112 104402
2799 013114 014526
2800 013116 005737 014352
2801 013122 001402
2802 013124 104402 014532
2803 013130 000727
2804 013132 012604
2805 013134 012603
2806 013136 000002
2807
2808
2809
2810 013140 010546
2811 013142 010446
2812 013144 016605 000004
2813 013150 012537 013344
2814 013154 012537 013346
2815 013160 012537 013350
2816 013164 112537 013352
2817 013170 112537 013353
2818 013174 010566 000004
2819 013200 005005
2820 013202 012704 015144
2821 013206 122714 000015

;ASCII STRING INPUT ROUTINE
.INSTR: MOV R3,-(SP)
MOV R4,-(SP)
MOV 4(SP),MSG
ADD #2,4(SP)
.INST1: TYPE
.MSG: 0
MOV #INBUF,R4
MOV #7,R3
1$: TSTB @TKCSR
BPL 1$
MOV B @TKDBR,(R4)
BICB #200,(R4)
CMPB (R4),#25
BNE 200$
TYPE,MCRLF
BR .INST1
200$: CMPB (R4)+,#15
BEQ INSTR2
MOV B @TKDBR,@TPDBR
2$: TSTB @TPCSR
BPL 2$
DEC R3
BNE 1$
BR .INSTG
.INSTE: MOV R3,-(SP)
MOV R4,-(SP)
.INSTG: TYPE
MQM
TST @RDSW
BEQ 400$
TYPE,MCRLF
400$: BR .INST1
INSTR2: MOV (SP)+,R4
MOV (SP)+,R3
RTI

;IS IT (<G)

;CONVERT ASCII STRING TO OCTAL
.PARAM: MOV R5,-(SP)
MOV R4,-(SP)
MOV 4(SP),R5
MOV (R5)+,LOLIM
MOV (R5)+,HILIM
MOV (R5)+,DEVADR
MOVB (R5)+,LOBITS
MOVB (R5)+,ADRCNT
MOV R5,4(SP)
PARAM1: CLR R5
MOV #INBUF,R4
CMPB #15,(R4)
    
```

# E05

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 56  
 DZDQED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

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	2#RDSW	; IS CKSWR ROUTINE BEING USED
2838	013266	001023			BNE	PARTI	
2839	013270	104404		120\$:	INSTR		
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	.SAVOS:	MOV	4(SP), SAVPC
2870							
2871							; SAVE R0-R5
2872							
2873	013362	010537	001270	SVOS:	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	



# F05

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 57  
 DZDQED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

2878	013406	010037	001256		MOV RO, SAVRO
2879	013412	000002			RTI
2880					
2881					;RESTORE RO-R5
2882					
2883	013414	013700	001256	.RES05:	MOV SAVRO, RO
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 RO, -(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		MOV TEMP, TEMP3
2902	013476	062766	000002	001250	ADD #2, 12(SP)
2903	013504	012137	013666	000012	MOV (R1)+, WRDCNT
2904	013510	112137	013670	1\$:	MOVB (R1)+, CHRCNT
2905	013514	112137	013671		MOVB (R1)+, SPACNT
2906	013520	013137	013672		MOV @2(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

# G05

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 58  
 DZDQED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

2934	013636	005337	013666		DEC	WRDCNT	
2935	013642	001322			BNE	IS	
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							
2948							
2949							
2950							
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	2(SP),(SP)	;GET TRP
2955	013706	006316			TRPOK: ASL	(SP)	;MULTIPLY TRAP ARG BY 2
2956	013710	042716	177001		BIC	#177001,(SP)	;CLEAR UNWANTED BITS
2957	013714	062716	001314		ADD	#.TRPTAB,(SP)	;POINTER TO SUBROUTINE ADDRESS
2958	013720	017616	000000		MOV	2(SP),(SP)	;SUBROUTINE ADDRESS
2959	013724	000136			JMP	2(SP)+	;GO TO SUBROUTINE
2960							
2961							
2962							
2963	013726	104414					
2964	013730	032777	010000	165242	.HLT: CKSWR		
2965	013736	001406			BIT	#SW12,2SWR	
2966	013740	105777	165244		BEQ	XBX	
2967	013744	100003			TSTB	2TPCSR	
2968	013746	112777	000207	165236	BPL	XBX	
2969	013754	032777	020000	165216	MOVB	#207,2TPDBR	
2970	013762	001074			XBX: BIT	#SW13,2SWR	
2971	013764	021637	001234		BNE	HALTS	
2972	013770	001404			CMP	(SP),LSTERR	
2973	013772	011637	001234		BEQ	IS	
2974	013776	105037	001312		MOV	(SP),LSTERR	
2975	014002	104406			CLRB	ERRFLG	
2976	014004	011605			IS: SAVOS		
2977	014006	162705	000002		MOV	(SP),R5	
2978	014012	011504			SUB	#2,R5	
2979	014014	006304			MOV	(R5),R4	
2980	014016	061504			ASL	R4	
2981	014020	006304			ADD	(R5),R4	
2982	014022	042704	177001		ASL	R4	
2983	014026	062704	016360		BIC	#177001,R4	
2984	014032	012437	014124		ADD	#.ERRTAB,R4	
2985	014036	012437	014136		MOV	(R4)+,ERRMSG	
2986	014042	011437	014150		MOV	(R4)+,DATAHD	
2987	014046	105737	001312		MOV	(R4),DATABP	
2988	014052	001403			TSTB	ERRFLG	
2989	014054	005737	014150		BEQ	TYPMSG	
					TST	DATABP	



# H05

DZDGE MACY11 27(1006) 22-DEC-76 11:32 PAGE 59  
 DZDGED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

```

2990 014060 001027
2991 014062 104402
2992 014064 015011
2993 014066 104411
2994 014070 014250
2995 014072 104402
2996 014074 015077
2997 014076 104411
2998 014100 014242
2999 014102 104402
3000 014104 014532
3001 014106 112737 177777 001312
3002 014114 005737 014124
3003 014120 001402
3004 014122 104402
3005 014124 000000
3006 014126
3007 014126 005737 014136
3008 014132 001402
3009 014134 104402
3010 014136 000000
3011 014140 005737 014150
3012 014144 001402
3013 014146 104410
3014 014150 000000
3015 014152 104407
3016 014154 005777 165020
3017 014160 100005
3018 014162 010046
3019 014164 016600 000002
3020 014170 000000
3021 014172 012600
3022 014174 104414
3023 014176 005237 001232
3024 014202 032777 000400 164770
3025 014210 001007
3026 014212 032777 002000 164760
3027 014220 001407
3028 014222 013737 001216 001214
3029 014230 012706 001200
3030 014234 000177 164754
3031 014240 000002
3032 014242 000001
3033 014244 006 002
3034 014246 001274
3035 014250 000001
3036 014252 003 002
3037 014254 001226
3038
3039
3040
3041 014256
3042 014256 012737 014270 000024
3043 014264 000000
3044 014266 000777
3045

TYPMSG: BNE TYPDAT
         TYPE
         MTSTN
         CNVRT
         XTSTN
         TYPE
         MERRPC
         CNVRT
         ERTABO
         TYPE
         MCRLF
         MOVB #-1,ERRFLG
         TST  ERRMSG
         BEQ  WRKO.FM
         TYPE
ERRMSG: 0
WRKO.FM:
         TST  DATAHD
         BEQ  TYPDAT
         TYPE
DATAHD: 0
TYPDAT: TST  DATABP
         BEQ  RESREG
         CONVRT
DATABP: 0
RESREG: RES05
HALTS:  TST  @SWR
        BPL  EXITER
        PUSHRO
        MOV  2(SP),R0
        HALT
        POPRO
EXITER: CKSWR
        INC  ERRCNT
        BIT  #SW08,@SWR
        BNE  1$
        BIT  #SW10,@SWR
        BEQ  2$
        MOV  NEXT,RETURN
1$:     MOV  #STACK,SP
        JMP  @RETURN
2$:
ERTABO: 1
        .BYTE 6,2
XTSTN:  1
        .BYTE 3,2
        TSTNO
        ;ENTER HERE ON POWER FAILURE

.PFAIL:
        MOV  #RESTART,24
        HALT
        BR   .
;SET UP FOR POWER UP TRAP
;HALT ON POWER DOWN NORMAL

```

```

3046                                     ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
3047
3048 014270                                RESTAR:
3049 014270 012737 014256 000024        MOV     #PFAIL, 24           ;SET UP FOR POWER FAILURE
3050 014276 012706 001200                MOV     #STACK, SP
3051 014302 005037 015206                ^' R     TEMP
3052 014306 005237 015206                INC     TEMP
3053 014312 001375                        BNE     .-4
3054 014314 104402                        TYPE
3055 014316 014534                        MPFAIL
3056 014320 104411                        CNVRT
3057 014322 014344                        PFTAB
3058 014324 005037 001312                CLR     ERRFLG
3059 014330 005037 001234                CLR     LSTERR
3060 014334 104412                        MSTCLR
3061 014336 104413                        MEMCLR
3062 014340 000177 164650                JMP     @RETURN
3063 014344 000001                        PFTAB: 1
3064 014346 003 002                      .BYTE 3, 2
3065 014350 001226                        TSTNO
3066
3067
3068                                     ;CHECK SWITCH REGISTER ROUTINE. CHECKS FOR †G TO ALLOW CHANGING
3069                                     ;OF LOC. 176.
3070                                     ;LOCATIONS USED:
3071 014352 000000                        RDSW: .WORD 0
3072
3073
3074 014354 005737 000042                .CKSWR: TST     @#42
3075 014360 001042                        BNE     OUT
3076 014362 022737 000176 001200        CMP     #SWREG, SWR           ;SOFTWARE SWITCH REGISTER PRESENT
3077 014370 001036                        BNE     OUT                   ;NO, GET OUT
3078 014372 105777 164606                TSTB   @TKCSR                ;YES, WAIT FOR
3079 014376 100033                        BPL     OUT                   ;READY, GET CHARACTER
3080 014400 017737 164602 013012        MOV     @TKDBR, .MSG         ;AND STRIP OFF
3081 014406 042737 177600 013012        BIC     #177600, .MSG        ;THE GARBAGE
3082 014414 122737 000007 013012        CMPB   #7, .MSG              ;IS IT A †G?
3083 014422 001021                        BNE     OUT
3084 014424 104402 014502                TYPE, %CNTG
3085 014430 005137 014352                .CNTLU: COM     @#RDSW
3086 014434 104402 014506                TYPE, %MSWR
3087 014440 104411 014474                CNVRT, %SWREGC
3088 014444 104403 014515                INSTR, %MNEW
3089 014450 104405                        PARAM
3090 014452 000000                        0
3091 014454 177777                        177777
3092 014456 000176                        SWREG
3093 014460 000 001                      .BYTE 0, 1
3094 014462 104402 014532                TYPE, %MCRLF
3095 014466 005037 014352                OUT:   CLR     @#RDSW
3096 014472 000002                        RTI
3097 014474 000001                        SWREGC: 1
3098 014476 006 002                      .BYTE 6, 2
3099 014500 000176                        SWREG
3100 014502 057377 000107                %CNTG: .ASCIZ <377>/†G/
3101 014506 051777 051127 020075        %MSWR: .ASCIZ <377>/SWR= /
    
```



# J05

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 61  
DZDQED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

3102	014514	000				
3103	014515	040	047040	053505	SMNEW: .ASCIZ / NEW= /	
3104	014522	020075	000			
3105		014526			.EVEN	
3106	014526	020040	000077		MOM: .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	126	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			

# K05

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 62  
 DZDQED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

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			
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



3214	016422	017404				DT3	
3215	016424	016634				EM1	
3216	016426	017224				DH5	;HALT 6
3217	016430	017434				DT5	
3218	016432	016715				EM3	
3219	016434	017266				DH6	;HALT 7
3220	016436	017456				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>/DQRCR DQTCR DQERR /
					.EVEN		
3222	017316	000002			DT0:	2	
	017320	006	007		.BYTE	6,7	
3223	017322	015326			.XPOLY		
3224	017324	006	001		.BYTE	6,1	
3225	017326	001270			.SAVRS		
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			.SAVRS		
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			.TEMPS		
3248	017404	000003			DT3:	3	
3249	017406	006	003		.BYTE	6,3	
3250	017410	001256			.SAVRO		
3251	017412	006	004		.BYTE	6,4	
3252	017414	001254			.TEMPS		
3253	017416	006	002		.BYTE	6,2	
3254	017420	001252			.TEMP4		

# M05

DZDGE MACY11 27(1006) 22-DEC-76 11:32 PAGE 64  
 DZDGED.P11 16-DEC-76 13:29 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

3255	017422	000002		DT4:	2	
3256	017424	006	004	.BYTE	6,4	
3257	017426	001254			TEMPS	
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		SAVRO		
3263	017442	006	002	.BYTE	6,2	
3264	017444	001260		SAVR1		
3265	017446	006	004	.BYTE	6,4	
3266	017450	001254		TEMPS		
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		SAVRO		
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		











LOCK	001220	1073#	1573#	1628#	2235#	2281#	2299#	2743										
LOC1	015346	1977#	2022#	2023	2025	3190#												
LOGICA	012454	930	2693#															
LOKFLG	001313	1110#																
LOLIM	013344	2813#	2846	2861#														
LPCNT	001224	1075#	2726#	2727	2730#													
LSTERR	001234	1079#	1226#	2665#	2971	2973#	3059#											
MCRLF	014532	2786	2802	2894	3000	3094	3107#											
MCSRX	014753	2671	3134#															
MDATA	015250	2922	2933	3172#														
MEMCLR=	104413	1142#	1410	1499	1560	1776	1817	1914	2168	2201	2356	2357	2389	3061				
MEPASS	014572	2669	3113#															
MERRPC	015077	2996	3153#															
MERRX	015000	2683	3140#															
MERR2	014617	1040	1342	3117#														
MERR3	014666	1274	3124#															
MISC. =	000012	654#	1599	1726	1743	1837	1854	1871	1884	1931	1957	2004	2284	2316				
		2330	2341	2415	2469	2519	2535	2567	2571	2587	2591	2652						
MLOCK	014724	1298	3130#															
MNEW	015025	1267	3145#															
MPASSX	014767	2679	3138#															
MPFAIL	014534	3055	3108#															
MPOLY	016440	2375	3221#															
MOM	014526	2799	3106#															
MR	014614	1316	3116#															
MSTCLR=	104412	1140#	1594	1639	1714	1715	1782	1824	1825	1917	1918	2045	2069	2093				
		2117	2141	2174	2207	2237	2283	2300	2448	2496	2498	2548	3060					
MTITLE	001000	1049#	1232															
MTSTN	015011	2992	3142#															
MTSTPC	014712	1307	3128#															
MVECX	014761	2675	3136#															
NEXT	001216	1072#	1322#	1392#	1403	1476#	1559#	1687	1709#	1775#	1816#	1913#	2044#	2068#				
		2092#	2116#	2140#	2167#	2200#	2234#	2274#	2372#	2732	3028							
ODDBIT=	001000	637#	1011															
OUT	014466	3075	3077	3079	3083	3095#												
PARAM =	104405	1130#	1308	3089														
PARAM1	013200	2819#	2840															
PARERR	013254	2822	2824	2826	2835#	2845	2847	2849										
PART1	013336	2838	2858#															
PASCNT	001230	1077#	1221#	2667#	2686	2709												
PFTAB	014344	3057	3063#															
POLY. =	000017	659#																
POPPO =	012600	606#	3021															
POP1SP=	005726	604#																
POP2SP=	022626	608#	1685															
PS =	177776	598#	951#	1216#	1291#	2423#												
PUSHRO=	010046	605#	3018															
PUSH1S=	005746	603#																
PUSH2S=	024646	607#																
RDSM	014352	2751	2800	2837	3071#	3085#	3095#											
RESREG	014152	3012	3015#															
RESTAR	014270	3042	3048#															
RESTRT	012464	2688	2691	2698#														
RESOS =	104407	1134#	3015															
RETURN	001214	1071#	1228#	1311	1315#	1317	1321#	1403#	1404	1687#	1688	1712#	1713	1774#				
		1815#	1912#	2166#	2199#	2373#	2699#	2732#	2733	3028#	3030	3062						











H06

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 73  
DZDQED.P11 16-DEC-76 13:29 CROSS REFERENCE TABLE -- USER SYMBOLS

.HLT	013726	925	2963#				
.INSTE	013106	1129	2796#				
.INSTG	013112	2795	2798#				
.INSTR	012770	1127	2772#				
.INST1	013010	2776#	2787	2803			
.MEMCL	012200	1143	2629#				
.MSG	013012	2774*	2777#	3080*	3081*	3082	
.MSTCL	012324	1141	2651#				
.PARAM	013140	1131	2810#				
.PFAIL	014256	923	1218	3041#	3049		
.RESOS	013414	1135	2883#				
.SAVOS	013354	1133	2869#				
.SCOPE	012530	1121	2716#				
.SCOPI	012642	1123	2740#				
.START	001512	974	1216#	1228			
.SYNC	015350	1425	1509	1654	2412	2466	3191#
.TRPSR	013674	927	2952#				
.TRPTA	001314	1119#	2957				
.TYPE	012662	1125	2748#				



DEND	18	2658													
DGFRNT	18	545													
HLT	6098	1437	1444	1454	1523	1530	1604	1610	1667	1670	1678	1742	1752	1790	1793
	1861	1880	1890	1969	2016	2030	2051	2075	2099	2123	2147	2182	2215	2248	2253
	2291	2308	2313	2323	2328	2339	2352	2383	2428	2442	2490				
HOMO	13748	2041	2065	2089	2113	2137									
IDENT	18														
ORANGE	18	1318													
TESTA1	18														
TESTB1	18														
TESTC1	18														
TESTD1	18														
TESTE1	18	50													
TESTF1	18														
TESTH1	18														
TESTH2	18														
SBEGIN	18	1288													
SBUFE	18	3165													
SCATCH	18	662													
SCLRVE	18	1253													
SCONVR	18	2890													
SEOP	18	2658													
SGETFL	18														
SGETPA	18	1306													
SHEADE	18	545													
SILT	18	2960													
SINSTR	18	2769													
SINTNP	18														
SMINT	18														
SMSG	18	3106													
SPARAM	18	2807													
SPFAIL	18	3038													
SRED	13778	2533	2589												
SREG	18	2866													
SSCOPE	18	2713													
SSCOPI	18	2737													
SSETFL	18														
SSETVE	18	920													
SSTART	18	1208													
SSYMO	18	562													
STRAPS	18	1111													
STRPOE	18	1120	1122	1124	1126	1128	1130	1132	1134	1136	1138	1140	1142	1144	1146
STRPSR	18	2947													
STSTN	18	1318	1389	1473	1556	1706	1771	1812	1909	2041	2065	2089	2113	2137	2163
	2196	2230	2271												
STYPE	18	2745													
SVARIA	18	1047													

. ABS. 017474 000

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

MULE:DZDQED.BIN,MULE:DZDQED.SEG/SOL/CRF=DSKZ:UNIV.P11,DSKZ:DZDQED.P11

J06

DZDQE MACY11 27(1006) 22-DEC-76 11:32 PAGE 76  
DZDQED.P11 16-DEC-76 13:29 CROSS REFERENCE TABLE -- MACRO NAMES

RUN-TIME: 19 30 3 SECONDS  
RUN-TIME RATIO: 152/53=2.8  
CORE USED: 19K (37 PAGES)