

DQ11

BASIC LOGIC TEST PART 1 MD-11-DZDQA-C

EP-DZDQA-C-DL-B

APR 1977

COPYRIGHT © 1977

digital

FICHE 1 OF 1

MADE IN USA

Test ID	Component	Value 1	Value 2	Value 3	Value 4	Value 5	Value 6	Value 7	Value 8
101	AND	0	1	1	0	0	1	1	0
102	OR	1	0	0	1	1	0	0	1
103	XOR	0	1	1	0	1	0	0	1
104	NOT	1	0	0	1	0	1	1	0
105	AND	1	1	0	0	1	0	0	1
106	OR	0	0	1	1	0	1	1	0
107	XOR	1	0	0	1	0	1	0	1
108	NOT	0	1	1	0	1	0	0	1
109	AND	0	1	1	0	1	0	0	1
110	OR	1	0	0	1	1	0	0	1
111	XOR	1	1	0	0	1	0	0	1
112	NOT	0	0	1	1	0	1	1	0
113	AND	1	1	0	0	1	0	0	1
114	OR	0	0	1	1	0	1	1	0
115	XOR	1	0	0	1	0	1	0	1
116	NOT	0	1	1	0	1	0	0	1
117	AND	1	1	0	0	1	0	0	1
118	OR	0	0	1	1	0	1	1	0
119	XOR	1	0	0	1	0	1	0	1
120	NOT	0	1	1	0	1	0	0	1

101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120

B01

CO1

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 2
DZDQAC.P11 16-DEC-76 11:24

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDQA-C-D
PRODUCT NAME: BASIC LOGIC TEST PART 1
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 BASIC READ/WRITE TEST FIRST CHECKS THAT THE DQ11 WILL RESPOND TO ADDRESSING, THEN THE TEST VERIFIES ALL THE READ/WRITE BITS IN THE:
 RECEIVER CSR
 TRANSMITTER CSR
 ERROR REGISTER
 SYNC REGISTER

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. DZDOC [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)

2. REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 8K 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 8K OF MEMORY.
 LOCATION 1400 THRU 1600 ARE ESPECIALLY TO

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 4
 DZDQAC.P11 16-DEC-76 11:24

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 SMR TO ZERO FOR "AUTO SIZING" OR LEAVE
 LEAVE SMR BIT 7=1 TO USE EXISTING PARAMETERS SET UP
 BY DQ11 TRIAL PROGRAM OR A PREVIOUSLY RUN DQ11 DIAGNOSTIC
 THAT USED THE "AUTO SIZING".
 ****REFER TO SECTION 4.1 FOR SOFTWARE SWITCH REGISTER OPERATION
 AND OPTIONS.****

NOTE: THE SOFTWARE SWITCH REGISTER IS LOCATED AT LOC.176
 SOFTWARE DISPLAY REGISTER IS LOCATED AT LOC.174

C. THEN START

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

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

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

1400 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE USER IF AUTO SIZING IS DONE. FOR INFORMATION OF STATUS TABLE SEE SECTION 8.4 FOR HELP.

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

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

4.1 CONTROL SWITCH SETTINGS

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

CONTROL:

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

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

SW 15	SET: HALT ON ERROR
SW 14	SET: LOOP ON CURRENT TEST
SW 13	SET: INHIBIT ERROR PRINT OUT
SW 12	SET: INHIBIT TYPE OUT/BELL ON ERROR.
SW 11	SET: INHIBIT ITERATIONS
SW 10	SET: ESCAPE TO NEXT TEST

SW 09 SET: LOOP WITH CURRENT DATA
 SW 08 SET: CATCH ERROR AND LOOP ON IT
 SW 07 SET: USE PREVIOUS STATUS TABLE. CLR-DO AUTO SIZE.
 SW 06 SET:
 SW 05 SET:
 SW 04 SET:
 SW 03 SET:
 SW 02 SET: LOCK ON SELECTED TEST
 SW 01 SET: RESTART PROGRAM AT SELECTED TEST
 SW 00 SET: RESELECT DQ11'S DESIRED ACTIVE.

4.1.2 SWITCH REGISTER RESTRICTIONS

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

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

SW 01 IT IS STRONGLY SUGGESTED THAT
 AT LEAST ONE PASS HAS BEEN MADE
 BEFORE TRYING TO SELECT A TEST
 THAT IS NOT IN THE ORDER OF SEQUENCE
 THE REASON BEING IS THAT THE
 PROGRAM HAS TO CLEAR AREAS AND SET
 UP PARAMETERS. ALSO WHEN A TEST IS
 SELECTED ALWAYS START AT THE VERY
 BEGINNING OF THAT TEST.

SW 09 LOOP ON CURRENT DATA:
 THIS SWITCH WILL ONLY WORK IF
 CALL "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 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 DZDQA-C CSR: 160000 VEC: 300 PASSES: 000001 ERRORS: 000000

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

8.3 TST1 (MINI MONITOR)

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

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

8.4 KEY LOCATIONS

RETURN (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 EXISTES

8.5.6 JUMPER ON END OF CABLE?

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 11
DZDQAC.P11 16-DEC-76 11:24

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 APPRIQATE DQSTXX: AFTER AUTO SIZING

8.5.8 SET FOR ODD OR EVEN PARITY?

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

8.5.9 FINDING THE VECTOR.

THE PROGRAM SETS "PRIMARY DONE", "SECONDAY DONE", AND "INTERUPT ENABLE" AND LOOKS FOR AN INTERRUPT. IF IT INTERRUPTS IT IS PICKED UP AND STORED AWAY. IF NO INTERRUPT 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

MO1

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 12
 DZDQAC.P11 16-DEC-76 11:24

531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567

```
.ENABLE AMA
;MAINDEC-11-DZDQA-C/<377>/DQ11 STATIC LOGIC TEST-PART 1
;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-DZDQA-C/<377>/DQ11 STATIC LOGIC TEST-PART 1"
;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
SW15=100000      ;=1,HALT ON ERROR
SW14=40000       ;=1,LOOP ON CURRENT TEST
SW13=20000       ;=1,INHIBIT ERROR TYPEOUT
SW12=10000       ;=1,DELETE TYPEOUT/BELL ON ERROR.
SW11=4000        ;=1,INHIBIT ITERATIONS
SW10=2000        ;=1,ESCAPE TO NEXT TEST ON ERROR
SW09=1000        ;=1,LOOP WITH CURRENT DATA
SW08=400         ;=1,LOOP ON ERROR
SW06=100
SW05=40
SW04=20
SW03=10
SW02=4
SW01=2
SW00=1

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

```
100000
040000
020000
010000
004000
002000
001000
000400
000100
000040
000020
000010
000004
000002
000001
```

NO1

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 13
 DZDQAC.P11 16-DEC-76 11:24 GENERAL DEFINATIONS AND EQUIVALENCIES

```

568
569
570 ;REGISTER DEFINITIONS
571
572 000000 R0=%0 ;GENERAL REGISTER
573 000001 R1=%1 ;GENERAL REGISTER
574 000002 R2=%2 ;GENERAL REGISTER
575 000003 R3=%3 ;GENERAL REGISTER
576 000004 R4=%4 ;GENERAL REGISTER
577 000005 R5=%5 ;GENERAL REGISTER
578 000006 SP=%6 ;PROCESSOR STACK POINTER
579 000007 PC=%7 ;PROGRAM COUNTER
580
581 ;LOCATION EQUIVALENCIES
582
583 177570 DSWR= 177570 ;HARDWARE SWITCH REGISTER LOC.
584 177570 DLIGHTS=177570 ;HARDWARE DISPLAY REGISTER LOC.
585 177776 PS=177776 ;PROCESSOR STATUS WORD
586 001200 STACK=1200 ;START OF PROCESSOR STACK
587
588 ;INSTRUCTION DEFINITIONS
589
590 005746 PUSH1SP=5746 ;DECREMENT PROCESSOR STACK 1 WORD
591 005726 POP1SP=5726 ;INCREMENT PROCESSOR STACK 1 WORD
592 010046 PUSHRO=10046 ;SAVE R0 ON STACK
593 012600 POPRO=12600 ;RESTORE R0 FROM STACK
594 024646 PUSH2SP=24646 ;DECREMENT STACK TWICE
595 022626 POP2SP=22626 ;INCREMENT STACK TWICE
596 .EQUIV ENT,HLT ;BASIC DEFINITION OF ERROR CALL
597
598
599 100000 BIT15=100000
600 040000 BIT14=40000
601 020000 BIT13=20000
602 010000 BIT12=10000
603 004000 BIT11=4000
604 002000 BIT10=2000
605 001000 BIT9=1000
606 000400 BIT8=400
607 000200 BIT7=200
608 000100 BIT6=100
609 000040 BIT5=40
610 000020 BIT4=20
611 000010 BIT3=10
612 000004 BIT2=4
613 000002 BIT1=2
614 000001 BIT0=1
615
616 ;DQ11 OPTIONAL DEFINITIONS
617
618
619 002000 ABBIT=2000
620 004000 ACTBIT=4000
621 010000 BABIT=10000
622 020000 BBBIT=20000
623 040000 JUMBIT=40000
  
```

624 001000
 625 100000
 626
 627
 628
 629
 630 000000
 631 000001
 632 000002
 633 000003
 634 000004
 635 000005
 636 000006
 637 000007
 638
 639 000010
 640 000011
 641 000012
 642 000013
 643 000014
 644 000015
 645 000016
 646 000017
 647
 648

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.

```

        649          ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
        650          .=0
        651 000000 000002      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        652 000002 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        653 000004 000006      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        654 000006 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        655 000010 000012      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        656 000012 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        657 000014 000016      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        658 000016 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        659 000020 000022      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        660 000022 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        661 000024 000026      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        662 000026 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        663 000030 000032      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        664 000032 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        665 000034 000036      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        666 000036 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        667 000040 000042      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        668 000042 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        669 000044 000046      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        670 000046 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        671 000050 000052      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        672 000052 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        673 000054 000056      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        674 000056 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        675 000060 000062      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        676 000062 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        677 000064 000066      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        678 000066 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        679 000070 000072      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        680 000072 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        681 000074 000076      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        682 000076 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        683 000100 000102      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        684 000102 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        685 000104 000106      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        686 000106 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        687 000110 000112      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        688 000112 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        689 000114 000116      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        690 000116 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        691 000120 000122      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        692 000122 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        693 000124 000126      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        694 000126 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        695 000130 000132      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        696 000132 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        697 000134 000136      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        698 000136 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        699 000140 000142      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        700 000142 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        701 000144 000146      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        702 000146 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
        703 000150 000152      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
        704 000152 000000      HALT     ;EXAMINE STACK TO FIND CAUSE
    
```


705	000154	000156	.+2	:UNEXPECTED TRAP TO THIS LOCATION
706	000156	000000	HALT	:EXAMINE STACK TO FIND CAUSE
707	000160	000162	.+2	:UNEXPECTED TRAP TO THIS LOCATION
708	000162	000000	HALT	:EXAMINE STACK TO FIND CAUSE
709	000164	000166	.+2	:UNEXPECTED TRAP TO THIS LOCATION
710	000166	000000	HALT	:EXAMINE STACK TO FIND CAUSE
711	000170	000172	.+2	:UNEXPECTED TRAP TO THIS LOCATION
712	000172	000000	HALT	:EXAMINE STACK TO FIND CAUSE
713	000174	000176	.+2	:UNEXPECTED TRAP TO THIS LOCATION
714	000176	000000	HALT	:EXAMINE STACK TO FIND CAUSE
715	000200	000202	.+2	:UNEXPECTED TRAP TO THIS LOCATION
716	000202	000000	HALT	:EXAMINE STACK TO FIND CAUSE
717	000204	000206	.+2	:UNEXPECTED TRAP TO THIS LOCATION
718	000206	000000	HALT	:EXAMINE STACK TO FIND CAUSE
719	000210	000212	.+2	:UNEXPECTED TRAP TO THIS LOCATION
720	000212	000000	HALT	:EXAMINE STACK TO FIND CAUSE
721	000214	000216	.+2	:UNEXPECTED TRAP TO THIS LOCATION
722	000216	000000	HALT	:EXAMINE STACK TO FIND CAUSE
723	000220	000222	.+2	:UNEXPECTED TRAP TO THIS LOCATION
724	000222	000000	HALT	:EXAMINE STACK TO FIND CAUSE
725	000224	000226	.+2	:UNEXPECTED TRAP TO THIS LOCATION
726	000226	000000	HALT	:EXAMINE STACK TO FIND CAUSE
727	000230	000232	.+2	:UNEXPECTED TRAP TO THIS LOCATION
728	000232	000000	HALT	:EXAMINE STACK TO FIND CAUSE
729	000234	000236	.+2	:UNEXPECTED TRAP TO THIS LOCATION
730	000236	000000	HALT	:EXAMINE STACK TO FIND CAUSE
731	000240	000242	.+2	:UNEXPECTED TRAP TO THIS LOCATION
732	000242	000000	HALT	:EXAMINE STACK TO FIND CAUSE
733	000244	000246	.+2	:UNEXPECTED TRAP TO THIS LOCATION
734	000246	000000	HALT	:EXAMINE STACK TO FIND CAUSE
735	000250	000252	.+2	:UNEXPECTED TRAP TO THIS LOCATION
736	000252	000000	HALT	:EXAMINE STACK TO FIND CAUSE
737	000254	000256	.+2	:UNEXPECTED TRAP TO THIS LOCATION
738	000256	000000	HALT	:EXAMINE STACK TO FIND CAUSE
739	000260	000262	.+2	:UNEXPECTED TRAP TO THIS LOCATION
740	000262	000000	HALT	:EXAMINE STACK TO FIND CAUSE
741	000264	000266	.+2	:UNEXPECTED TRAP TO THIS LOCATION
742	000266	000000	HALT	:EXAMINE STACK TO FIND CAUSE
743	000270	000272	.+2	:UNEXPECTED TRAP TO THIS LOCATION
744	000272	000000	HALT	:EXAMINE STACK TO FIND CAUSE
745	000274	000276	.+2	:UNEXPECTED TRAP TO THIS LOCATION
746	000276	000000	HALT	:EXAMINE STACK TO FIND CAUSE
747	000300	000302	.+2	:UNEXPECTED TRAP TO THIS LOCATION
748	000302	000000	HALT	:EXAMINE STACK TO FIND CAUSE
749	000304	000306	.+2	:UNEXPECTED TRAP TO THIS LOCATION
750	000306	000000	HALT	:EXAMINE STACK TO FIND CAUSE
751	000310	000312	.+2	:UNEXPECTED TRAP TO THIS LOCATION
752	000312	000000	HALT	:EXAMINE STACK TO FIND CAUSE
753	000314	000316	.+2	:UNEXPECTED TRAP TO THIS LOCATION
754	000316	000000	HALT	:EXAMINE STACK TO FIND CAUSE
755	000320	000322	.+2	:UNEXPECTED TRAP TO THIS LOCATION
756	000322	000000	HALT	:EXAMINE STACK TO FIND CAUSE
757	000324	000326	.+2	:UNEXPECTED TRAP TO THIS LOCATION
758	000326	000000	HALT	:EXAMINE STACK TO FIND CAUSE
759	000330	000332	.+2	:UNEXPECTED TRAP TO THIS LOCATION
760	000332	000000	HALT	:EXAMINE STACK TO FIND CAUSE

E02

DZD9A MACY11 27(1006) 22-DEC-76 10:57 PAGE 17
DZD9AC.P11 16-DEC-76 11:24 TRAPCATCHER FOR UNEXPECTED INTERRUPTS

761	000334	000336	.+2	:UNEXPECTED TRAP TO THIS LOCATION
762	000336	000000	HALT	:EXAMINE STACK TO FIND CAUSE
763	000340	000342	.+2	:UNEXPECTED TRAP TO THIS LOCATION
764	000342	000000	HALT	:EXAMINE STACK TO FIND CAUSE
765	000344	000346	.+2	:UNEXPECTED TRAP TO THIS LOCATION
766	000346	000000	HALT	:EXAMINE STACK TO FIND CAUSE
767	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
768	000352	000000	HALT	:EXAMINE STACK TO FIND CAUSE
769	000354	000356	.+2	:UNEXPECTED TRAP TO THIS LOCATION
770	000356	000000	HALT	:EXAMINE STACK TO FIND CAUSE
771	000360	000362	.+2	:UNEXPECTED TRAP TO THIS LOCATION
772	000362	000000	HALT	:EXAMINE STACK TO FIND CAUSE
773	000364	000366	.+2	:UNEXPECTED TRAP TO THIS LOCATION
774	000366	000000	HALT	:EXAMINE STACK TO FIND CAUSE
775	000370	000372	.+2	:UNEXPECTED TRAP TO THIS LOCATION
776	000372	000000	HALT	:EXAMINE STACK TO FIND CAUSE
777	000374	000376	.+2	:UNEXPECTED TRAP TO THIS LOCATION
778	000376	000000	HALT	:EXAMINE STACK TO FIND CAUSE
779	000400	000402	.+2	:UNEXPECTED TRAP TO THIS LOCATION
780	000402	000000	HALT	:EXAMINE STACK TO FIND CAUSE
781	000404	000406	.+2	:UNEXPECTED TRAP TO THIS LOCATION
782	000406	000000	HALT	:EXAMINE STACK TO FIND CAUSE
783	000410	000412	.+2	:UNEXPECTED TRAP TO THIS LOCATION
784	000412	000000	HALT	:EXAMINE STACK TO FIND CAUSE
785	000414	000416	.+2	:UNEXPECTED TRAP TO THIS LOCATION
786	000416	000000	HALT	:EXAMINE STACK TO FIND CAUSE
787	000420	000422	.+2	:UNEXPECTED TRAP TO THIS LOCATION
788	000422	000000	HALT	:EXAMINE STACK TO FIND CAUSE
789	000424	000426	.+2	:UNEXPECTED TRAP TO THIS LOCATION
790	000426	000000	HALT	:EXAMINE STACK TO FIND CAUSE
791	000430	000432	.+2	:UNEXPECTED TRAP TO THIS LOCATION
792	000432	000000	HALT	:EXAMINE STACK TO FIND CAUSE
793	000434	000436	.+2	:UNEXPECTED TRAP TO THIS LOCATION
794	000436	000000	HALT	:EXAMINE STACK TO FIND CAUSE
795	000440	000442	.+2	:UNEXPECTED TRAP TO THIS LOCATION
796	000442	000000	HALT	:EXAMINE STACK TO FIND CAUSE
797	000444	000446	.+2	:UNEXPECTED TRAP TO THIS LOCATION
798	000446	000000	HALT	:EXAMINE STACK TO FIND CAUSE
799	000450	000452	.+2	:UNEXPECTED TRAP TO THIS LOCATION
800	000452	000000	HALT	:EXAMINE STACK TO FIND CAUSE
801	000454	000456	.+2	:UNEXPECTED TRAP TO THIS LOCATION
802	000456	000000	HALT	:EXAMINE STACK TO FIND CAUSE
803	000460	000462	.+2	:UNEXPECTED TRAP TO THIS LOCATION
804	000462	000000	HALT	:EXAMINE STACK TO FIND CAUSE
805	000464	000466	.+2	:UNEXPECTED TRAP TO THIS LOCATION
806	000466	000000	HALT	:EXAMINE STACK TO FIND CAUSE
807	000470	000472	.+2	:UNEXPECTED TRAP TO THIS LOCATION
808	000472	000000	HALT	:EXAMINE STACK TO FIND CAUSE
809	000474	000476	.+2	:UNEXPECTED TRAP TO THIS LOCATION
810	000476	000000	HALT	:EXAMINE STACK TO FIND CAUSE
811	000500	000502	.+2	:UNEXPECTED TRAP TO THIS LOCATION
812	000502	000000	HALT	:EXAMINE STACK TO FIND CAUSE
813	000504	000506	.+2	:UNEXPECTED TRAP TO THIS LOCATION
814	000506	000000	HALT	:EXAMINE STACK TO FIND CAUSE
815	000510	000512	.+2	:UNEXPECTED TRAP TO THIS LOCATION
816	000512	000000	HALT	:EXAMINE STACK TO FIND CAUSE

817	000514	000516	.+2	:UNEXPECTED TRAP TO THIS LOCATION
818	000516	000000	HALT	:EXAMINE STACK TO FIND CAUSE
819	000520	000522	.+2	:UNEXPECTED TRAP TO THIS LOCATION
820	000522	000000	HALT	:EXAMINE STACK TO FIND CAUSE
821	000524	000526	.+2	:UNEXPECTED TRAP TO THIS LOCATION
822	000526	000000	HALT	:EXAMINE STACK TO FIND CAUSE
823	000530	000532	.+2	:UNEXPECTED TRAP TO THIS LOCATION
824	000532	000000	HALT	:EXAMINE STACK TO FIND CAUSE
825	000534	000536	.+2	:UNEXPECTED TRAP TO THIS LOCATION
826	000536	000000	HALT	:EXAMINE STACK TO FIND CAUSE
827	000540	000542	.+2	:UNEXPECTED TRAP TO THIS LOCATION
828	000542	000000	HALT	:EXAMINE STACK TO FIND CAUSE
829	000544	000546	.+2	:UNEXPECTED TRAP TO THIS LOCATION
830	000546	000000	HALT	:EXAMINE STACK TO FIND CAUSE
831	000550	000552	.+2	:UNEXPECTED TRAP TO THIS LOCATION
832	000552	000000	HALT	:EXAMINE STACK TO FIND CAUSE
833	000554	000556	.+2	:UNEXPECTED TRAP TO THIS LOCATION
834	000556	000000	HALT	:EXAMINE STACK TO FIND CAUSE
835	000560	000562	.+2	:UNEXPECTED TRAP TO THIS LOCATION
836	000562	000000	HALT	:EXAMINE STACK TO FIND CAUSE
837	000564	000566	.+2	:UNEXPECTED TRAP TO THIS LOCATION
838	000566	000000	HALT	:EXAMINE STACK TO FIND CAUSE
839	000570	000572	.+2	:UNEXPECTED TRAP TO THIS LOCATION
840	000572	000000	HALT	:EXAMINE STACK TO FIND CAUSE
841	000574	000576	.+2	:UNEXPECTED TRAP TO THIS LOCATION
842	000576	000000	HALT	:EXAMINE STACK TO FIND CAUSE
843	000600	000602	.+2	:UNEXPECTED TRAP TO THIS LOCATION
844	000602	000000	HALT	:EXAMINE STACK TO FIND CAUSE
845	000604	000606	.+2	:UNEXPECTED TRAP TO THIS LOCATION
846	000606	000000	HALT	:EXAMINE STACK TO FIND CAUSE
847	000610	000612	.+2	:UNEXPECTED TRAP TO THIS LOCATION
848	000612	000000	HALT	:EXAMINE STACK TO FIND CAUSE
849	000614	000616	.+2	:UNEXPECTED TRAP TO THIS LOCATION
850	000616	000000	HALT	:EXAMINE STACK TO FIND CAUSE
851	000620	000622	.+2	:UNEXPECTED TRAP TO THIS LOCATION
852	000622	000000	HALT	:EXAMINE STACK TO FIND CAUSE
853	000624	000626	.+2	:UNEXPECTED TRAP TO THIS LOCATION
854	000626	000000	HALT	:EXAMINE STACK TO FIND CAUSE
855	000630	000632	.+2	:UNEXPECTED TRAP TO THIS LOCATION
856	000632	000000	HALT	:EXAMINE STACK TO FIND CAUSE
857	000634	000636	.+2	:UNEXPECTED TRAP TO THIS LOCATION
858	000636	000000	HALT	:EXAMINE STACK TO FIND CAUSE
859	000640	000642	.+2	:UNEXPECTED TRAP TO THIS LOCATION
860	000642	000000	HALT	:EXAMINE STACK TO FIND CAUSE
861	000644	000646	.+2	:UNEXPECTED TRAP TO THIS LOCATION
862	000646	000000	HALT	:EXAMINE STACK TO FIND CAUSE
863	000650	000652	.+2	:UNEXPECTED TRAP TO THIS LOCATION
864	000652	000000	HALT	:EXAMINE STACK TO FIND CAUSE
865	000654	000656	.+2	:UNEXPECTED TRAP TO THIS LOCATION
866	000656	000000	HALT	:EXAMINE STACK TO FIND CAUSE
867	000660	000662	.+2	:UNEXPECTED TRAP TO THIS LOCATION
868	000662	000000	HALT	:EXAMINE STACK TO FIND CAUSE
869	000664	000666	.+2	:UNEXPECTED TRAP TO THIS LOCATION
870	000666	000000	HALT	:EXAMINE STACK TO FIND CAUSE
871	000670	000672	.+2	:UNEXPECTED TRAP TO THIS LOCATION
872	000672	000000	HALT	:EXAMINE STACK TO FIND CAUSE

DZDGA MACY11 27(1006) 22-DEC-76 10:57 PAGE 19
 DZDQAC.P11 16-DEC-76 11:24 TRAPCATCHER FOR UNEXPECTED INTERRUPTS

873	000674	000676	.+2	:UNEXPECTED TRAP TO THIS LOCATION
874	000676	000000	HALT	:EXAMINE STACK TO FIND CAUSE
875	000700	000702	.+2	:UNEXPECTED TRAP TO THIS LOCATION
876	000702	000000	HALT	:EXAMINE STACK TO FIND CAUSE
877	000704	000706	.+2	:UNEXPECTED TRAP TO THIS LOCATION
878	000706	000000	HALT	:EXAMINE STACK TO FIND CAUSE
879	000710	000712	.+2	:UNEXPECTED TRAP TO THIS LOCATION
880	000712	000000	HALT	:EXAMINE STACK TO FIND CAUSE
881	000714	000716	.+2	:UNEXPECTED TRAP TO THIS LOCATION
882	000716	000000	HALT	:EXAMINE STACK TO FIND CAUSE
883	000720	000722	.+2	:UNEXPECTED TRAP TO THIS LOCATION
884	000722	000000	HALT	:EXAMINE STACK TO FIND CAUSE
885	000724	000726	.+2	:UNEXPECTED TRAP TO THIS LOCATION
886	000726	000000	HALT	:EXAMINE STACK TO FIND CAUSE
887	000730	000732	.+2	:UNEXPECTED TRAP TO THIS LOCATION
888	000732	000000	HALT	:EXAMINE STACK TO FIND CAUSE
889	000734	000736	.+2	:UNEXPECTED TRAP TO THIS LOCATION
890	000736	000000	HALT	:EXAMINE STACK TO FIND CAUSE
891	000740	000742	.+2	:UNEXPECTED TRAP TO THIS LOCATION
892	000742	000000	HALT	:EXAMINE STACK TO FIND CAUSE
893	000744	000746	.+2	:UNEXPECTED TRAP TO THIS LOCATION
894	000746	000000	HALT	:EXAMINE STACK TO FIND CAUSE
895	000750	000752	.+2	:UNEXPECTED TRAP TO THIS LOCATION
896	000752	000000	HALT	:EXAMINE STACK TO FIND CAUSE
897	000754	000756	.+2	:UNEXPECTED TRAP TO THIS LOCATION
898	000756	000000	HALT	:EXAMINE STACK TO FIND CAUSE
899	000760	000762	.+2	:UNEXPECTED TRAP TO THIS LOCATION
900	000762	000000	HALT	:EXAMINE STACK TO FIND CAUSE
901	000764	000766	.+2	:UNEXPECTED TRAP TO THIS LOCATION
902	000766	000000	HALT	:EXAMINE STACK TO FIND CAUSE
903	000770	000772	.+2	:UNEXPECTED TRAP TO THIS LOCATION
904	000772	000000	HALT	:EXAMINE STACK TO FIND CAUSE
905	000774	000776	.+2	:UNEXPECTED TRAP TO THIS LOCATION
906	000776	000000	HALT	:EXAMINE STACK TO FIND CAUSE


```

963          000220          .=220
964 000220 012702 001400   CSRMAP: MOV    #1400,R2      ;CLEAR ALL STATUS TABLE
965 000224 005022          CLR    (R2)+      ;DO CLEAR
966 000226 022702 001512   CMP    #1512,R2   ;ALL TABLE DONE
967 000232 001374          BNE    -6         ;BR IF MORE TO GO
968 000234 005037 001504   CLR    DQNUM      ;SET NUMBER OF DQ11S TO 0
969 000240 012702 001400   MOV    #1400,R2   ;SET TABLE POINTER
970 000244 012701 160000   MOV    #160000,R1 ;GET FIRST FLOATING ADDRESS
971 000250 012737 000614 000004   MOV    #55,2#4    ;SET FOR TIME OUT TRAP--NO DEVICE--
972 000256 112761 000012 000005 1$:  MOVB   #12,5(R1)   ;TRY AND SEL MISC REGISTER
973 000264 005061 000006   CLR    6(R1)      ;TRY AND CLEAR MISC REG
974 000270 012711 010000   MOV    #10000,(R1);TRY AND SET RX ACTIVE
975 000274 022761 030000 000006   CMP    #30000,6(R1);LOOK FOR SYNC 1 AND SYNC 2
976 000302 001071          BNE    2$         ;THIS IS NOT A DQ11 IF I BRANCH
977 000304 010122          MOV    R1,(R2)+   ;NOW THIS IS A DQ11 --STORE CSR
978 000306 052712 100000   BIS    #SYNBIT,(R2);SET FOR TWO SYNC CHARS
979 000312 005011          CLR    (R1)       ;CLEAR DQ ACTIVE BIT
980 000314 112761 000010 000005   MOVB   #10,5(R1)  ;SEL CHAR DET REGISTER
981 000322 012761 177777 000006   MOV    #-1,6(R1) ;WRITE INTO CHAR DET REG
982 000330 005761 000006   TST    6(R1)      ;WAS THE REGISTER WRITTEN?
983 000334 001402          BEQ    +6         ;APPARENTLY NO BB OPTION.
984 000336 052712 020000   BIS    #BBBIT,(R2);SET FOR BB OPTION
985 000342 112761 000017 000005   MOVB   #17,5(R1)  ;SEL POLYNO. REGISTER
986 000350 012761 177777 000006   MOV    #-1,6(R1) ;WRITE POLYNO.REGISTER
987 000356 005761 000006   TST    6(R1)      ;WAS REG WRITTEN??
988 000362 001402          BEQ    +6         ;BR IF NO AB OPTION
989 000364 052712 002000   BIS    #ABBIT,(R2);SET FOR AB OPTION
990 000370 012761 001400 000002   MOV    #1400,2(R1);TRY TO SET .DTR. .RS.
991 000376 032761 001400 000002   BIT    #1400,2(R1);DID ANY OF THEM SET
992 000404 001402          BEQ    +6         ;BR IF NO BA OPTION
993 000406 052712 010000   BIS    #BABIT,(R2);SET FOR BA OPTION
994 000412 032761 030000 000002   BIT    #30000,2(R1);DID .CS. .CO. SET
995 000420 001402          BEQ    +6         ;BR IF NO JUMPER
996 000422 052712 040000   BIS    #JUMBIT,(R2);SET FOR JUMPER
997 000426 052712 004000   BIS    #ACTBIT,(R2);SET FOR ACTIVE ON FIRST NON-SYNC
998 000432 052712 001000   BIS    #ODDBIT,(R2);SET FOR ODD VRC.....
999 000436 005722          TST    (R2)+     ;POP POINTER
1000 000440 005011          CLR    (R1)       ;CLEAR RCSR
1001 000442 005061 000002   CLR    2(R1)      ;CLEAR TCSR
1002 000446 005061 000002   CLR    2(R1)      ;CLEAR AGAIN
1003 000452 005061 000004   CLR    4(R1)      ;CLEAR ERROR REG
1004 000456 005061 000006   CLR    6(R1)      ;CLEAR SEC REG
1005 000462 005237 001504          INC    DQNUM      ;UPDATE NUMBER OF DQ11S
1006 000466 062701 000010 3$:  ADD    #10,R1     ;UPDATE CSR POINTER BY 10 (8)
1007 000472 022701 164000   CMP    #164000,R1;HAVE ALL FLOATING ADDRESSES BEEN CHECKED??
1008 000476 001267          BNE    1$         ;BR IF NOT ALL DONE
1009 000500 005037 001500   CLR    DQACTV     ;ZERO ACTIVE DQ11S
1010 000504 005737 001504   TST    DQNUM      ;WERE ANY DQ11S FOUND
1011 000510 001434          BEQ    4$         ;HEY BUDDY. NO DQ11S FOUND IN SYSTEM
1012 000512 013701 001504   MOV    DQNUM,R1   ;SAVE NUMBER OF DQ11S
1013 000516 010137 001276   MOV    R1,SAVNUM  ;SAVE NUMBER FOR ACT11
1014 000522 000241          3$:  CLC             ;CLEAR CARRY
1015 000524 006137 001500   ROL    DQACTV     ;+++++ ACTIVE ADDRESS
1016 000530 005237 001500   INC    DQACTV     ;SET BIT 0
1017 000534 005301          DEC    R1         ;DEC NUMBER OF DQ11S
1018 000536 001371          BNE    3$         ;BR IF MORE TO GO

```

```

1019 000540 012737 000006 000004      MOV      #6, R4      ;RESET TIME OUT VECTOR
1020 000546 013737 001500 001502      MOV      DQACTV, SAVACT ;SAVE ACTIVE
1021 000554 012737 000340 000022      MOV      #340, R22   ;SET IOT TRAP PRIO: TO 7
1022 000562 012702 001400          MOV      #1400, R2   ;SET TABLE POINTER
1023 000566 012700 000300          MOV      #300, R0   ;SET VECTOR START
1024 000572 012701 000302          MOV      #302, R1   ;SET VECTOR+2 START
1025 000576 000137 000056          JMP      VECMAP     ;GO FIND THE VECTORS
1026 000602 104402          4$:      TYPE      ;TYPE MESSAGE
1027 000604 016525          MERR2    ;I DIDN'T FIND ANY DQ11S. DON'T USE AUTO SIZE.
1028 000606 005000          CLR      R0
1029 000610 000000          HALT
1030 000612 000776          BR
1031 000614 012716 000466          5$:      MOV      #-2, (SP) ;HOW CAN I TEST NO DQ11S
1032 000620 000002          RTI      ;DON'T LET OPR HIT CONT. SW
                                           ;ENTERED BY TIME OUT TRAP
                                           ;GO HOME.

1033
1034
1035          001000          .=1000
1036 001000 005377 040515 047111      MTITLE: .ASCIZ <377><12>/MAINDEC-11-DZDQA-C/<377>/DQ11 STATIC LOGIC TEST-PART 1/<377>
1037 001006 042504 026503 030461
1038 001014 042055 042132 040521
1039 001022 041455 042377 030521
1040 001030 020061 052123 052101
1041 001036 041511 046040 043517
1042 001044 041511 052040 051505
1043 001052 026524 040520 052122
1044 001060 030440 000377
1045
1046          001200          .=1200
1047          ;INDIRECT POINTERS
1048
1049 001200 177570      SWR:      177570      ;SWITCH REGISTER POINTER
1050 001202 177570      LIGHTS:   177570     ;DISPLAY REGISTER POINTER
1051 001204 177560      TKCSR:   177560     ;TELETYPE KEYBOARD CONTROL REGISTER
1052 001206 177562      TKDBR:   177562     ;TELETYPE KEYBOARD DATA BUFFER
1053 001210 177564      TPCSR:   177564     ;TELEPRINTER CONTROL REGISTER
1054 001212 177566      TPDBR:   177566     ;TELEPRINTER DATA BUFFER
1055
1056          ;PROGRAM CONTROL PARAMETERS
1057
1058 001214 000000      RETURN:  0          ;SCOPE ADDRESS FOR LOOP ON TEST
1059 001216 000000      NEXT:    0          ;ADDRESS OF NEXT TEST TO BE EXECUTED
1060 001220 000000      LOCK:    0          ;ADDRESS FOR LOCK ON CURRENT DATA
1061 001222 000003      ICOUNT:  3          ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
1062 001224 000000      LPCNT:   0          ;NUMBER OF ITERATIONS COMPLETED
1063 001226 000000      TSTNO:   0          ;NUMBER OF TEST IN PROGRESS
1064 001230 000000      PASCNT:  0          ;NUMBER OF PASSES COMPLETED
1065 001232 000000      ERRCNT:  0          ;TOTAL NUMBER OF ERRORS
1066 001234 000000      LSTERR:  0          ;PC OF LAST ERROR CALL
1067
1068          ;PROGRAM VARIABLES
1069
1070 001236 000000      CHAR1:   0
1071 001240 000000      CHAR2:   0
1072 001242 000000      CHAR3:   0
1073 001244 000000      TEMP1:   0          ;TEMPORARY STORAGE
1074 001246 000000      TEMP2:   0          ;TEMPORARY STORAGE

```

K02

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 23
DZDQAC.P11 16-DEC-76 11:24 PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

1075	001250	000000	TEMP3:	0	: TEMPORARY STORAGE
1076	001252	000000	TEMP4:	0	: TEMPORARY STORAGE
1077	001254	000000	TEMP5:	0	: TEMPORARY STORAGE
1078	001256	000000	SAVR0:	0	: R0 STORAGE
1079	001260	000000	SAVR1:	0	: R1 STORAGE
1080	001262	000000	SAVR2:	0	: R2 STORAGE
1081	001264	000000	SAVR3:	0	: R3 STORAGE
1082	001266	000000	SAVR4:	0	: R4 STORAGE
1083	001270	000000	SAVR5:	0	: R5 STORAGE
1084	001272	000000	SAVSP:	0	: STACK POINTER STORAGE
1085	001274	000000	SAVPC:	0	: PROGRAM COUNTER STORAGE
1086	001276	000000	SAVNUM:	0	
1087	001300	000001	CREAM:	.BLKW 1	
1088	001302	000000	RUNFLG:	0	
1089	001304	000000	RUN:	0	
1090	001306	000000	RUNCNT:	0	


```

1091
1092
1093
1094 001310 000
1095 001311 000
1096 001312 000
1097 001313 000
1098 000000
1099
1100
1101
1102
1103
1104
1105
1106 001314
1107 104400
1108 001314 014436
1109 104401
1110 001316 014550
1111 104402
1112 001320 014570
1113 104403
1114 001322 014676
1115 104404
1116 001324 015014
1117 104405
1118 001326 015046
1119 104406
1120 001330 015262
1121 104407
1122 001332 015322
1123 104410
1124 001334 015354
1125 104411
1126 001336 015360
1127 104412
1128 001340 017222
1129 104413
1130 001342 017220
1131 104414
1132 001344 016262
1133 104415
1134 001346 016336
1135
1136
1137
1138
1139
1140
1141 001350 000000
1142 001352 000000
1143 001354 000000
1144 001356 000000
1145 001360 000000
1146 001362 000000

;PROGRAM CONTROL FLAGS
INIFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG
STFLG: .BYTE 0 ;TEST START FLAG
ERRFLG: .BYTE 0 ;ERROR OCCURED FLAG
LOKFLG: .BYTE 0 ;LOCK ON CURRENT TEST FLAG
SY=0

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

;*****
;*****
TRPTAB:
SCOPE=TRAP+0 ;CALL TO SCOPE LOOP AND ITERATION HANDLER
;SCOPE
SCOPI=TRAP+1 ;CALL TO LOOP ON CURRENT DATA HANDLER
;SCOPI
TYPE=TRAP+2 ;CALL TO TELETYPE OUTPUT ROUTINE
;TYPE
INSTR=TRAP+3 ;CALL TO ASCII STRING INPUT ROUTINE
;INSTR
INSTER=TRAP+4 ;CALL TO INPUT ERROR HANDLER
;INSTER
PARAM=TRAP+5 ;CALL TO NUMERICAL DATA INPUT ROUTINE
;PARAM
SAVOS=TRAP+6 ;CALL TO REGISTER SAVE ROUTINE
;SAVOS
RESOS=TRAP+7 ;CALL TO REGISTER RESTORE ROUTINE
;RESOS
CONVRT=TRAP+10 ;CALL TO DATA OUTPUT ROUTINE
;CONVRT
CNVRT=TRAP+11 ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
;CNVRT
MSTCLR=TRAP+12 ;CALL TO ISSUE MASTER CLEAR
;MSTCLR
MEMCLR=TRAP+13 ;CALL TO CLEAR ALL SCRATCH PAD MEMORIES
;MEMCLR
CKSWR=TRAP+14 ;CALL TO ALLOW SWREG TO BE L/OADED FROM TTY
;CKSWR
CNTLU=TRAP+15 ;CALL TO ALLOW LOADING OF SAREG FROM TTY
;CNTLU

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

;DQ11 VECTOR AND REGISTER INDIRECT POINTERS
DQAVEC: 0 ;POINTER TO DQ11 RECEIVER INTERRUPT VECTOR
DQRLVL: 0 ;POINTER TO DQ11 RECEIVER INTERRUPT SERVICE PS
DQTEVC: 0 ;POINTER TO DQ11 TRANSMITTER INTERRUPT VECTOR
DQTLVL: 0 ;POINTER TO DQ11 TRANSMITTER INTERRUPT SERVICE PS
DQRC SR: 0 ;POINTER TO DQ11 RECEIVER CONTROL REGISTER
DQRC SH: 0 ;POINTER TO HIGH BYTE OF DQ11 RECEIVER CONTROL REGISTER
    
```

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 25
 DZDQAC.P11 16-DEC-76 11:24 PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

```

1147 001364 000000      DQTCR: 0          ; POINTER TO DQ11 TRANSMITTER CONTROL REGISTER
1148 001366 000000      DQERR: 0         ; POINTER TO DQ11 ERROR REGISTER
1149 001370 000000      DQREG: 0         ; POINTER TO HIGH BYTE OF ERROR REGISTER
1150 001372 000000      DQSEC: 0         ; POINTER TO DQ11 SECONDARY REGISTER
1151 001374 000000      DQSECH: 0        ; POINTER TO HIGH BYTE OF DQ11 SECONDARY REGISTER
1152
1153
1154
1155                      ; DQ11 STATUS TABLE AND ADDRESS ASSIGNMENTS
1156
1157                      . = 1400
1158 001400 000001      DQCR00: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 00
1159 001402 000001      DQST00: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 00
1160 001404 000001      DQCR01: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 01
1161 001406 000001      DQST01: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 01
1162 001410 000001      DQCR02: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 02
1163 001412 000001      DQST02: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 02
1164 001414 000001      DQCR03: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 03
1165 001416 000001      DQST03: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 03
1166 001420 000001      DQCR04: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 04
1167 001422 000001      DQST04: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 04
1168 001424 000001      DQCR05: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 05
1169 001426 000001      DQST05: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 05
1170 001430 000001      DQCR06: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 06
1171 001432 000001      DQST06: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 06
1172 001434 000001      DQCR07: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 07
1173 001436 000001      DQST07: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 07
1174 001440 000001      DQCR10: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 10
1175 001442 000001      DQST10: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 10
1176 001444 000001      DQCR11: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 11
1177 001446 000001      DQST11: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 11
1178 001450 000001      DQCR12: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 12
1179 001452 000001      DQST12: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 12
1180 001454 000001      DQCR13: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 13
1181 001456 000001      DQST13: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 13
1182 001460 000001      DQCR14: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 14
1183 001462 000001      DQST14: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 14
1184 001464 000001      DQCR15: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 15
1185 001466 000001      DQST15: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 15
1186 001470 000001      DQCR16: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 16
1187 001472 000001      DQST16: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 16
1188 001474 000001      DQCR17: .BLKW 1   ; CONTROL STATUS REGISTER FOR DEVICE NO: 17
1189 001476 000001      DQST17: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 17
1190 001500 000001      DQACTV: .BLKW 1   ; HOLD ACTIVE BITS FOR TESTING
1191 001502 000001      SAVACT: .BLKW 1   ; SAVE NUMBER OF ACTIVE DQ11S
1192 001504 000001      DQNUM: .BLKW 1    ; OCTAL NUMBER OF TOTAL NUMBER OF DQ11S
1193 001506 000001      DQCSR: .BLKW 1    ; CSR OF DQ11 UNDER TEST
1194 001510 000001      DQSTAT: .BLKW 1   ; VECTOR AND CONFIGURATION STATUS OF DQ11 UNDER TEST
1195
1196                      ; PROGRAM INITIALIZATION
1197                      ; LOCK OUT INTERRUPTS
1198                      ; SET UP PROCESSOR STACK
1199                      ; SET UP POWER FAIL VECTOR
1200                      ; CLEAR PROGRAM CONTROL FLAGS AND COUNTS
1201                      ; TYPE TITLE MESSAGE
1202

```

N02

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 26
 DZDQAC.P11 16-DEC-76 11:24 PROGRAM INITIALIZATION AND START UP.

1203	001512	012737	000340	177776	.START:	MOV	#340,PS	;LOCK OUT INTERRUPTS
1204	001520	012706	001200			MOV	#STACK,SP	;SET UP STACK
1205	001524	012737	016164	000024		MOV	#PFAIL,#24	;SET UP POWER FAIL VECTOR
1206	001532	013737	001504	001276		MOV	QNUM,SAVNUM	
1207	001540	105037	001311			CLRB	STFLG	;CLEAR START FLAG
1208	001544	005037	001230			CLR	PASCNT	;CLEAR PASS COUNT
1209	001550	105037	001312			CLRB	ERRFLG	;CLEAR ERROR FLAG
1210	001554	005037	001302			CLR	RUNFLG	
1211	001560	012737	001400	001300		MOV	#1400,CREAM	
1212	001566	005037	001232			CLR	ERRCNT	;CLEAR ERROR COUNT
1213	001572	005037	001234			CLR	LSTERR	;CLEAR LAST ERROR POINTER
1214	001576	012737	000001	001226		MOV	#1,TSTNO	;SET UP FOR TEST 1
1215	001604	012737	001512	001214		MOV	#.START,RETURN	;SET UP FOR POWER FAIL BEFORE TESTING STARTS
1216								;HAS INITIALIZATION BEEN PERFORMED
1217	001612	105737	001310			TSTB	INIFLG	
1218	001616	001075				BNE	12\$	
1219	001620	104402	001000			TYPE	.MTITLE	;TYPE TITLE MESSAGE
1220	001624	105137	001310			COMB	INIFLG	;IF NOT SET FLAG AND DO
1221								
1222	001630	012737	177570	001200		MOV	#DSWR,SWR	;MOV HARDWARE SWR TO SWR
1223	001636	012737	177570	001202		MOV	#DLIGHTS,LIGHTS	;MOV DISPLAY LIGHTS TO LIGHTS
1224	001644	013746	000006			MOV	#6,-(SP)	;SAVE VECTORS
1225	001650	013746	000004			MOV	#4,-(SP)	
1226	001654	012737	001674	000004		MOV	#64\$,#4	;SET UP FOR TIMEOUT
1227	001662	022777	177777	177310		CMP	#-1,#SWR	;REFERENCE HARDWARE SWITCH REGISTER
1228	001670	001402				BEQ	65\$	
1229	001672	000407				BR	66\$	
1230	001674	022626			64\$:	CMP	(SP)+,(SP)+	;ADJUST STACK
1231	001676	012737	000176	001200	65\$:	MOV	#SWREG,SWR	;POINT TO SOFTWARE SWITCH REG
1232	001704	012737	000174	001202		MOV	#DISPREG,LIGHTS	;POINT TO SOFT DISPLAY REG
1233	001712	012637	000004		66\$:	MOV	(SP)+,#4	;RESTORE VECTORS
1234	001716	012637	000006			MOV	(SP)+,#6	
1235	001722	005737	000042			TST	#42	;UNDER MONITOR
1236	001726	001005				BNE	67\$	
1237	001730	022737	000176	001200		CMP	#SWREG,SWR	;IS SWREG USED
1238	001736	001001				BNE	67\$	
1239	001740	104415				CNTLU		
1240	001742	105777	177232		67\$:	TSTB	#SWR	
1241	001746	100402				BMI	.+6	
1242	001750	004737	000220			JSR	PC,CSRMAP	
1243	001754	104402	017012			TYPE	XHEAD	
1244	001760	012737	001400	001244		MOV	#1400,TEMP1	
1245	001766	017737	177252	001246		MOV	#TEMP1,TEMP2	
1246	001774	001406				BEQ	.+16	
1247	001776	104410				CONVRT		
1248	002000	017040				XSTATQ		
1249	002002	062737	000002	001244		ADD	#2,TEMP1	
1250	002010	000766				BR	.-22	
1251	002012	032777	000001	177160	12\$:	BIT	#SW00,#SWR	
1252	002020	001424				BEQ	1\$	
1253	002022	104402				TYPE		
1254	002024	016733				MNEW		
1255	002026	005000				CLR	RO	
1256	002030	000000				HALT		
1257	002032	104414				CKSWR		
1258	002034	027737	177140	001502		CMP	#SWR,SAVACT	

```

1259 002042 101404 BLOS 11$
1260 002044 104402 TYPE
1261 002046 016574 MERR3
1262 002050 000000 HALT
1263 002052 000776 BR -2
1264 002054 017737 177120 001500 11$: MOV @SWR,DQACTV
1265 002052 013700 001500 MOV DQACTV,RO
1266 002056 000000 HALT
1267 002070 104414 CKSWR
1268 002072 012700 000300 1$: MOV #300,RO
1269 002076 012701 000302 MOV #302,R1
1270 002102 010120 2$: MOV R1,(R0)+
1271 002104 005021 CLR (R1)+
1272 002106 022021 CMP (R0)+,(R1)+
1273 002110 022700 001000 CMP #1000,RO
1274 002114 001372 BNE 2$
1275
1276 ;TEST START AND RESTART
1277
1278 002116 012737 000340 177776 .BEGIN: MOV #340,PS ;LOCK OUT INTERRUPTS
1279 002124 012706 001200 MOV #STACK,SP ;SET UP STACK
1280 002130 005737 000042 TST @#42 ;IS PROGRAM UNDER MONITOR CONTROL
1281 002134 001040 BNE 3$
1282 002136 104414 CKSWR ;CHECK FOR <IG>
1283 002140 032777 000004 177032 BIT #BIT2,@SWR ;CHECK FOR LOCK ON TEST
1284 002146 001411 BEQ 1$
1285 002150 104402 016632 TYPE MLOCK
1286 002154 012737 000240 014446 MOV #NOP,TTST
1287 002162 012737 000240 014450 MOV #NOP,TTST+2 ;SET UP TO LOCK
1288 002170 000406 BR 2$
1289 002172 013737 014544 014446 1$: MOV BRW,TTST
1290 002200 013737 014546 014450 MOV BRX,TTST+2 ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
1291 002206 032777 000002 176764 2$: BIT #SW01,@SWR ;IF SW01=1, GET STARTING PC
1292 002214 001410 BEQ 3$
1293 002216 104403 INSTR
1294 002220 016620 MTSTPC
1295 002222 104405 PARAM
1296 002224 002254 TST1
1297 002226 014160 TLAST
1298 002230 001214 #RETURN
1299 002232 001 .BYTE 1
1300 002233 001 .BYTE 1
1301 002234 000403 BR 4$
1302 002236 012737 002254 001214 3$: MOV #TST1,RETURN ;START AT TEST 1
1303 002244 104402 016522 4$: TYPE MR ;TYPE R
1304 002250 000177 176740 JMP @RETURN ;START TESTING
1305 ; TEST 1
1306 ;*****
1307 002254 012737 000001 001226 TST1: MOV #1,TSTNO
1308 002262 012737 002644 001214 MOV #TST2,RETURN
1309 002270 012737 002644 001216 MOV #TST2,NEXT
1310 002276 105737 001302 TSTB RUNFLG ;IS THIS MY FIRST TIME HERE?
1311 002302 001010 BNE 1$ ;BR IF FLAG IS SET
1312 002304 012737 000001 001304 MOV #BIT0,RUN ;SET RUN POINTER.
1313 002312 012737 000020 001306 MOV #16,RUNCNT ;SET FOR MAX OF 16 DQ11'S PER SYSTEM
1314 002320 105137 001302 COMB RUNFLG ;SET RUN FLAG

```

```

1315 002324 033737 001304 001500 1S: BIT RUN,DQACTV ;FIND AN ACTIVE DQ11 TO TEST.
1316 002332 001032 BNE 3S ;BR IF I FOUND ONE TO TEST.
1317 002334 005737 001500 TST DQACTV ;FIND OUT IF THERE ARE NO DQ11 ACTIVE.
1318 002340 001423 BEQ 2S ;BR TO FATAL ERROR. WHY AM I HERE IF NO ACTIVE DQ11'S???
1319 002342 000257 CCC ;CLEAR ALL THE CONDITION CODES OF CPU
1320 002344 006137 001304 ROL RUN ;UPDATE RUN POINTER
1321 002350 062737 000004 001300 ADD #4,CREAM ;UPDATE ADDRESS POINTER.
1322 002356 005337 001306 DEC RUNCNT ;DEC NUMBER OF TIMES I LOOKED AT ACTIVE.
1323 002362 001360 BNE 1S ;BR AND KEEP LOOKING.
1324 002364 012737 000020 001306 MOV #16,RUNCNT ;START RESTORING MY POINTERS.
1325 002372 012737 001400 001300 MOV #1400,CREAM ;RESTORE ADDRESS POINTER
1326 002400 012737 000001 001304 MOV #1,RUN ;RESTORE RUN POINTER.
1327 002406 000746 BR 1S ;KEEP ON TESTING.
1328 002410 104402 2S: TYPE ;ALERT OPERATOR OF FATAL ERROR
1329 002412 016525 MERR2 ;NO DQ11 ACTIVE. WHY AM I HERE???
1330 002414 000000 HALT ;YOU MUST RELOAD DQ11 DIAGNOSTIC!!
1331 002416 000776 BR .-2 ;STICK HERE ON CONT.
1332 002420 000257 3S: CCC ;CLEAR CPU COND. CODES
1333 002422 006137 001304 ROL RUN ;UPDATE RUN. ACTIVE DQ11 FOUND.
1334 002426 017737 176646 001506 MOV @CREAM,DQCSR ;PLACE ADDRESS OF DQ11 AT DQCSR
1335 002434 062737 000002 001300 ADD #2,CREAM ;UPDATE ADDRESS POINTER
1336 002442 017737 176632 001510 MOV @CREAM,DQSTAT ;PLACE STATUS OF DQ11 AT DQSTAT
1337 002450 062737 000002 001300 ADD #2,CREAM ;UPDATE ADDRESS POINTER
1338 002456 013737 001506 001360 MOV DQCSR,DQCSR
1339 002464 013737 001510 001350 MOV DQSTAT,DQVEC
1340 002472 042737 177007 001350 BIC #177007,DQVEC
1341 002500 013737 001350 001352 MOV DQVEC,DQRLVL ;GENERATE ADDRESS OF RECEIVER INTERRUPT SERVICE PS
1342 002506 062737 000002 001352 ADD #2,DQRLVL
1343 002514 013737 001352 001354 MOV DQRLVL,DQTVL ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT VECTOR
1344 002522 062737 000002 001354 ADD #2,DQTVL
1345 002530 013737 001354 001356 MOV DQTVL,DQTLVL ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT SERVICE PS
1346 002536 062737 000002 001356 ADD #2,DQTLVL
1347 002544 013737 001360 001362 MOV DQCSR,DQCSH
1348 002552 005237 001362 INC DQCSH ;GENERATE ADDRESS OF HIGH BYTE
1349 002556 013737 001360 001364 MOV DQCSR,DQCSR ;GENERATE ADDRESS OF TRANSMITTER CONTROL REGISTER
1350 002564 062737 000002 001364 ADD #2,DQCSR
1351 002572 013737 001364 001366 MOV DQCSR,DQERR ;GENERATE ADDRESS OF ERROR REGISTER
1352 002600 062737 000002 001366 ADD #2,DQERR
1353 002606 013737 001366 001370 MOV DQERR,DQREG ;GENERATE ADDRESS OF HIGH BYTE OF ERROR REGISTER
1354 002614 005237 001370 INC DQREG
1355 002620 013737 001370 001372 MOV DQREG,DQSEC ;GENERATE ADDRESS OF SECONDARY REGISTER
1356 002626 005237 001372 INC DQSEC
1357 002632 013737 001372 001374 MOV DQSEC,DQSECH ;GENERATE ADDRESS OF HIGH BYTE
1358 002640 005237 001374 INC DQSECH
1359
1360 ;ADDRESS SELECTOR TEST
1361 ;ADDRESS RECEIVER CONTROL REGISTER
1362 ;VERIFY THAT RECEIVER CONTROL REGISTER RESPONDS TO ADDRESSING
1363
1364 ; TEST 2
1365 ;*****
1366 002644 012737 000002 001226 TST2: MOV #2,TSTNO
1367 002652 012737 002730 001216 MOV #TST3,NEXT
1368 002660 012737 002706 000004 MOV #15,#4 ;SET UP TO RETURN FROM
1369 002666 012737 000340 000006 MOV #340,#6 ;BUS ERROR TRAP
1370 002674 013705 001360 MOV DQCSR,R5 ;GET ADDRESS OF RECEIVER CONTROL REGISTER

```

```

1371 002700 005777 176454          TST  2DQRCR          ; ADDRESS RECEIVER CONTROL REGISTER
1372 002704 000401                    BR   25              ; NO TRAP, REGISTER RESPONDED
1373 002706 104000                    15:  HLT  0          ; RECEIVER CONTROL REGISTER DID NOT
1374                                     ; RESPOND TO ADDRESSING
1375 002710 012706 001200                    25:  MOV  #STACK,SP  ; RESTORE STACK
1376 002714 012737 000006 000004          MOV  #6,2#4         ; RESTORE TRAPCATCHER
1377 002722 005037 000006                    CLR  2#6
1378 002726 104400                    35:  SCOPE          ; CHECK FOR ITERATIONS, LOOP
1379
1380                                     ; ADDRESS SELECTOR TEST
1381                                     ; ADDRESS TRANSMITTER CONTROL REGISTER
1382                                     ; VERIFY THAT TRANSMITTER CONTROL REGISTER RESPONDS TO ADDRESSING
1383
1384 ; TEST 3
1385 ;*****
1386 002730 012737 000003 001226  TST3:  MOV  #3,TSTNO
1387 002736 012737 003014 001216          MOV  #TST4,NEXT
1388 002744 012737 002772 000004          MOV  #15,2#4       ; SET UP TO RETURN FROM
1389 002752 012737 000340 000006          MOV  #340,2#6      ; BUS ERROR TRAP
1390 002760 013705 001364          MOV  DQTCR,R5      ; GET ADDRESS OF TRANSMITTER CONTROL REGISTER
1391 002764 005777 176374          TST  2DQTCR        ; ADDRESS TRANSMITTER CONTROL REGISTER
1392 002770 000401                    BR   25              ; NO TRAP, REGISTER RESPONDED
1393 002772 104000                    15:  HLT  0          ; TRANSMITTER CONTROL REGISTER DID NOT
1394                                     ; RESPOND TO ADDRESSING
1395 002774 012706 001200                    25:  MOV  #STACK,SP  ; RESTORE STACK
1396 003000 012737 000006 000004          MOV  #6,2#4         ; RESTORE TRAPCATCHER
1397 003006 005037 000006                    CLR  2#6
1398 003012 104400                    35:  SCOPE          ; CHECK FOR ITERATIONS, LOOP
1399
1400                                     ; ADDRESS SELECTOR TEST
1401                                     ; ADDRESS ERROR REGISTER
1402                                     ; VERIFY THAT ERROR REGISTER RESPONDS TO ADDRESSING
1403
1404 ; TEST 4
1405 ;*****
1406 003014 012737 000004 001226  TST4:  MOV  #4,TSTNO
1407 003022 012737 003100 001216          MOV  #TST5,NEXT
1408 003030 012737 003056 000004          MOV  #15,2#4       ; SET UP TO RETURN FROM
1409 003036 012737 000340 000006          MOV  #340,2#6      ; BUS ERROR TRAP
1410 003044 013705 001366          MOV  DQERR,R5      ; GET ADDRESS OF ERROR REGISTER
1411 003050 005777 176312          TST  2DQERR        ; ADDRESS ERROR REGISTER
1412 003054 000401                    BR   25              ; NO TRAP, REGISTER RESPONDED
1413 003056 104000                    15:  HLT  0          ; ERROR REGISTER DID NOT
1414                                     ; RESPOND TO ADDRESSING
1415 003060 012706 001200                    25:  MOV  #STACK,SP  ; RESTORE STACK
1416 003064 012737 000006 000004          MOV  #6,2#4         ; RESTORE TRAPCATCHER
1417 003072 005037 000006                    CLR  2#6
1418 003076 104400                    35:  SCOPE          ; CHECK FOR ITERATIONS, LOOP
1419
1420                                     ; ADDRESS SELECTOR TEST
1421                                     ; ADDRESS SECONDARY REGISTER
1422                                     ; VERIFY THAT SECONDARY REGISTER RESPONDS TO ADDRESSING
1423
1424 ; TEST 5
1425 ;*****
1426 003100 012737 000005 001226  TST5:  MOV  #5,TSTNO

```

```

1427 003106 012737 003164 001216      MOV      #TST6,NEXT
1428 003114 012737 003142 000004      MOV      #15,284      ;SET UP TO RETURN FROM
1429 003122 012737 000340 000006      MOV      #340,286     ;BUS ERROR TRAP
1430 003130 013705 001372      MOV      DQSEC,R5     ;GET ADDRESS OF SECONDARY REGISTER
1431 003134 005777 176232      TST      2DQSEC       ;ADDRESS SECONDARY REGISTER
1432 003140 000401      BR       2$           ;NO TRAP, REGISTER RESPONDED
1433 003142 104000      1$:      HLT      0       ;SECONDARY REGISTER DID NOT
1434                                ;RESPOND TO ADDRESSING
1435 003144 012706 001200 000004 2$:      MOV      #STACK,SP   ;RESTORE STACK
1436 003150 012737 000006      MOV      #6,284      ;RESTORE TRAPCATCHER
1437 003156 005037 000006      CLR      286
1438 003162 104400      3$:      SCOPE                ;CHECK FOR ITERATIONS, LOOP
1439
1440                                ;PRIMARY REGISTER ADDRESSING TEST
1441                                ;LOAD EACH PRIMARY REGISTER WITH A DIFFERENT
1442                                ;NUMBER AND VERIFY THAT THE CORRECT REGISTER
1443                                ;WAS ADDRESSED
1444
1445                                ; TEST 6
1446                                ;*****
1447 003164 012737 000006 001226  TST6:  MOV      #6,TSTNO
1448 003172 012737 003364 001216      MOV      #TST7,NEXT
1449 003200 012777 000040 176152      MOV      #40,2DQRCR   ;LOAD RECEIVER CONTROL REGISTER
1450                                ;WITH BITS
1451 003206 012777 000100 176150      MOV      #100,2DQTCR  ;LOAD TRANSMITTER CONTROL
1452                                ;REGISTER WITH BIT6
1453 003214 012777 000200 176144      MOV      #200,2DQERR  ;LOAD ERROR REGISTER
1454                                ;WITH BIT7
1455 003222 012777 000400 176142      MOV      #400,2DQSEC  ;LOAD SECONDARY REGISTER
1456                                ;WITH BITS
1457 003230 012705 000040      MOV      #40,R5       ;FIRST EXPECTED DATA
1458 003234 017704 176120      MOV      2DQRCR,R4    ;READ RECEIVER CONTROL REGISTER
1459 003240 013703 001360      MOV      DQRCR,R3     ;SET UP ADDRESS OF RECEIVER CONTROL REGISTER
1460 003244 020504      CMP      R5,R4        ;WAS RECEIVER CONTROL REGISTER ADDRESSED
1461 003246 001401      BEQ     1$           ;BR IF GOOD
1462 003250 104001      HLT     1             ;REGISTER ADDRESSING ERROR
1463 003252 006305      1$:      ASL     R5           ;NEXT EXPECTED DATA
1464 003254 017704 176104      MOV      2DQTCR,R4    ;READ TRANSMITTER CONTROL REGISTER
1465 003260 042704 077400      BIC     #77400,R4     ;CLEAR UNWANTED BITS
1466 003264 062703 000002      ADD     #2,R3         ;UPDATE ADDRESS OF EXPECTED REGISTER
1467 003270 020504      CMP     R5,R4        ;WAS TRANSMITTER CONTROL REGISTER ADDRESSED
1468 003272 001401      BEQ     2$           ;BR IF GOOD
1469 003274 104001      HLT     1             ;REGISTER ADDRESSING ERROR
1470 003276 006305      2$:      ASL     R5           ;NEXT EXPECTED DATA
1471 003300 017704 176062      MOV      2DQERR,R4    ;READ ERROR REGISTER
1472 003304 042704 170000      BIC     #170000,R4    ;CLEAR UNWANTED BITS
1473 003310 062703 000002      ADD     #2,R3         ;UPDATE EXPECTED REGISTER ADDRESS
1474 003314 020504      CMP     R5,R4        ;WAS ERROR REGISTER ADDRESSED
1475 003316 001401      BEQ     3$           ;BR IF GOOD
1476 003320 104001      HLT     1             ;REGISTER ADDRESSING ERROR
1477 003322 006305      3$:      ASL     R5           ;NEXT EXPECTED DATA
1478 003324 017704 176042      MOV      2DQSEC,R4    ;READ SECONDARY REGISTER
1479 003330 062703 000002      ADD     #2,R3         ;UPDATE EXPECTED REGISTER ADDRESS
1480 003334 020504      CMP     R5,R4        ;WAS SECONDARY REGISTER ADDRESSED
1481 003336 001401      BEQ     4$           ;BR IF GOOD
1482 003340 104001      HLT     1             ;REGISTER ADDRESSING ERROR

```

F03

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 31
DZDQAC.P11 16-DEC-76 11:24 BASIC DQ11 ADDRESSING TESTS.

1483 003342 005077 176012
1484 003346 005077 176012
1485 003352 005077 176010
1486 003356 005077 176010
1487 003362 104400

4S: CLR @DQRCR
CLR @DQTCSR
CLR @DQERR
CLR @DQSEC
5S: SCOPE

:CLEAR SEL 0
:CLEAR SEL 2
:CLEAR SEL 4
:CLEAR SEL 6
:CHECK FOR ITERATIONS, LOOP


```

1488
1489
1490
1491
1492
1493
1494
1495 003364 012737 000007 001226
1496 003372 012737 003440 001216
1497 003400 013703 001360
1498
1499 003404 012705 000002
1500
1501 003410 010513
1502 003412 011304
1503
1504 003414 020504
1505 003416 001401
1506 003420 104002
1507 003422 040513
1508
1509 003424 011304
1510 003426 005005
1511
1512 003430 020504
1513 003432 001401
1514 003434 104002
1515 003436 104400
1516
1517
1518
1519
1520
1521
1522
1523 003440 012737 000010 001226
1524 003446 012737 003514 001216
1525 003454 013703 001360
1526
1527 003460 012705 000010
1528
1529 003464 010513
1530 003466 011304
1531
1532 003470 020504
1533 003472 001401
1534 003474 104002
1535 003476 040513
1536
1537 003500 011304
1538 003502 005005
1539
1540 003504 020504
1541 003506 001401
1542 003510 104002
1543 003512 104400

```

```

;RECEIVER CONTROL REGISTER READ/WRITE TEST
;SET BIT1, VERIFY BIT1 WAS SET
;CLEAR BIT1, VERIFY BIT1 WAS CLEARED

; TEST 7
;*****
TST7: MOV #7,TSTNO
      MOV #TST10,NEXT
      MOV DQRCSR,R3
      ;LOAD R3 WITH ADDRESS
      ;OF RECEIVER CONTROL REGISTER
      ;RECEIVER CONTROL REGISTER WILL
      ;BE SET TO BIT1
      MOV #BIT1,R5
      ;LOAD RECEIVER CONTROL
      MOV R5,(R3)
      MOV (R3),R4
      ;(R4)=ACTUAL DATA
      ;IN RECEIVER CONTROL REGISTER
      ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
      CMP R5,R4
      BEQ 1$
      HLT 2
1$: BIC R5,(R3)
      ;RECEIVER CONTROL REGISTER DATA ERROR
      ;CLEAR BITS SET
      ;IN RECEIVER CONTROL REGISTER
      MOV (R3),R4
      CLR R5
      ;READ RECEIVER CONTROL REGISTER
      ;(R5)=EXPECTED CONTENTS
      ;OF RECEIVER CONTROL REGISTER, 0
      ;WAS RECEIVER CONTROL CLEARED
      CMP R5,R4
      BEQ 2$
      HLT 2
2$: SCOPE
      ;RECEIVER CONTROL REGISTER DATA ERROR
      ;CHECK FOR ITERATIONS, LOOP

;RECEIVER CONTROL REGISTER READ/WRITE TEST
;SET BIT3, VERIFY BIT3 WAS SET
;CLEAR BIT3, VERIFY BIT3 WAS CLEARED

; TEST 10
;*****
TST10: MOV #10,TSTNO
      MOV #TST11,NEXT
      MOV DQRCSR,R3
      ;LOAD R3 WITH ADDRESS
      ;OF RECEIVER CONTROL REGISTER
      ;RECEIVER CONTROL REGISTER WILL
      ;BE SET TO BIT3
      MOV #BIT3,R5
      ;LOAD RECEIVER CONTROL
      MOV R5,(R3)
      MOV (R3),R4
      ;(R4)=ACTUAL DATA
      ;IN RECEIVER CONTROL REGISTER
      ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
      CMP R5,R4
      BEQ 1$
      HLT 2
1$: BIC R5,(R3)
      ;RECEIVER CONTROL REGISTER DATA ERROR
      ;CLEAR BITS SET
      ;IN RECEIVER CONTROL REGISTER
      MOV (R3),R4
      CLR R5
      ;READ RECEIVER CONTROL REGISTER
      ;(R5)=EXPECTED CONTENTS
      ;OF RECEIVER CONTROL REGISTER, 0
      ;WAS RECEIVER CONTROL CLEARED
      CMP R5,R4
      BEQ 2$
      HLT 2
2$: SCOPE
      ;RECEIVER CONTROL REGISTER DATA ERROR
      ;CHECK FOR ITERATIONS, LOOP

```

H03

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 33
 DZDQAC.P11 16-DEC-76 11:24 BASIC RECEIVER READ/WRITE TESTS.

```

1544
1545
1546
1547
1548
1549
1550
1551 003514 012737 000011 001226
1552 003522 012737 003570 001216
1553 003530 013703 001360
1554
1555 003534 012705 000020
1556
1557 003540 010513
1558 003542 011304
1559
1560 003544 020504
1561 003546 001401
1562 003550 104002
1563 003552 040513
1564
1565 003554 011304
1566 003556 005005
1567
1568 003560 020504
1569 003562 001401
1570 003564 104002
1571 003566 104400
1572
1573
1574
1575
1576
1577
1578
1579 003570 012737 000012 001226
1580 003576 012737 003644 001216
1581 003604 013703 001360
1582
1583 003610 012705 000040
1584
1585 003614 010513
1586 003616 011304
1587
1588 003620 020504
1589 003622 001401
1590 003624 104002
1591 003626 040513
1592
1593 003630 011304
1594 003632 005005
1595
1596 003634 020504
1597 003636 001401
1598 003640 104002
1599 003642 104400

;RECEIVER CONTROL REGISTER READ/WRITE TEST
;SET BIT4, VERIFY BIT4 WAS SET
;CLEAR BIT4, VERIFY BIT4 WAS CLEARED

; TEST 11
;*****
TST11: MOV #11,TSTNO
MOV #TST12,NEXT
MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
;OF RECEIVER CONTROL REGISTER
;RECEIVER CONTROL REGISTER WILL
;BE SET TO BIT4
MOV #BIT4,R5 ;LOAD RECEIVER CONTROL
;(R4)=ACTUAL DATA
;(R5)=EXPECTED CONTENTS
;IN RECEIVER CONTROL REGISTER
;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
MOV R5,(R3)
MOV (R3),R4
CMP R5,R4
BEQ 1$
HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1$: BIC R5,(R3) ;CLEAR BITS SET
;IN RECEIVER CONTROL REGISTER
;READ RECEIVER CONTROL REGISTER
;R5=EXPECTED CONTENTS
;OF RECEIVER CONTROL REGISTER, 0
;WAS RECEIVER CONTROL CLEARED
MOV (R3),R4
CLR R5
CMP R5,R4
BEQ 2$
HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
2$: SCOPE ;CHECK FOR ITERATIONS, LOOP

;RECEIVER CONTROL REGISTER READ/WRITE TEST
;SET BITS, VERIFY BITS WAS SET
;CLEAR BITS, VERIFY BITS WAS CLEARED

; TEST 12
;*****
TST12: MOV #12,TSTNO
MOV #TST13,NEXT
MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
;OF RECEIVER CONTROL REGISTER
;RECEIVER CONTROL REGISTER WILL
;BE SET TO BITS
MOV #BITS,R5 ;LOAD RECEIVER CONTROL
;(R4)=ACTUAL DATA
;(R5)=EXPECTED CONTENTS
;IN RECEIVER CONTROL REGISTER
;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
MOV R5,(R3)
MOV (R3),R4
CMP R5,R4
BEQ 1$
HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1$: BIC R5,(R3) ;CLEAR BITS SET
;IN RECEIVER CONTROL REGISTER
;READ RECEIVER CONTROL REGISTER
;R5=EXPECTED CONTENTS
;OF RECEIVER CONTROL REGISTER, 0
;WAS RECEIVER CONTROL CLEARED
MOV (R3),R4
CLR R5
CMP R5,R4
BEQ 2$
HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
2$: SCOPE ;CHECK FOR ITERATIONS, LOOP

```

```

1600
1601                                     ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1602                                     ;SET BIT6, VERIFY BIT6 WAS SET
1603                                     ;CLEAR BIT6, VERIFY BIT6 WAS CLEARED
1604
1605                                     ; TEST 13
1606                                     ;*****
1607 003644 012737 000013 001226 †ST13: MOV #13,TSTNO
1608 003652 012737 003720 001216 MOV #TST14,NEXT
1609 003660 013703 001360 MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
1610                                     ;OF RECEIVER CONTROL REGISTER
1611 003664 012705 000100 MOV #BIT6,R5 ;RECEIVER CONTROL REGISTER WILL
1612                                     ;BE SET TO BIT6
1613 003670 010513 MOV R5,(R3) ;LOAD RECEIVER CONTROL
1614 003672 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
1615                                     ;IN RECEIVER CONTROL REGISTER
1616 003674 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1617 003676 001401 BEQ 1$
1618 003700 104002 HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1619 003702 040513 1$: BIC R5,(R3) ;CLEAR BITS SET
1620                                     ;IN RECEIVER CONTROL REGISTER
1621 003704 011304 MOV (R3),R4 ;READ RECEIVER CONTROL REGISTER
1622 003706 005005 CLR R5 ;(R5)=EXPECTED CONTENTS
1623                                     ;OF RECEIVER CONTROL REGISTER, 0
1624 003710 020504 CMP R5,R4 ;WAS RECEIVER CONTROL CLEARED
1625 003712 001401 BEQ 2$
1626 003714 104002 HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1627 003716 104400 2$: SCOPE ;CHECK FOR ITERATIONS, LOOP
1628
1629                                     ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1630                                     ;SET BIT7, VERIFY BIT7 WAS SET
1631                                     ;CLEAR BIT7, VERIFY BIT7 WAS CLEARED
1632
1633                                     ; TEST 14
1634                                     ;*****
1635 003720 012737 000014 001226 †ST14: MOV #14,TSTNO
1636 003726 012737 003774 001216 MOV #CKBBD,NEXT
1637 003734 013703 001360 MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
1638                                     ;OF RECEIVER CONTROL REGISTER
1639 003740 012705 000200 MOV #BIT7,R5 ;RECEIVER CONTROL REGISTER WILL
1640                                     ;BE SET TO BIT7
1641 003744 010513 MOV R5,(R3) ;LOAD RECEIVER CONTROL
1642 003746 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
1643                                     ;IN RECEIVER CONTROL REGISTER
1644 003750 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1645 003752 001401 BEQ 1$
1646 003754 104002 HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1647 003756 040513 1$: BIC R5,(R3) ;CLEAR BITS SET
1648                                     ;IN RECEIVER CONTROL REGISTER
1649 003760 011304 MOV (R3),R4 ;READ RECEIVER CONTROL REGISTER
1650 003762 005005 CLR R5 ;(R5)=EXPECTED CONTENTS
1651                                     ;OF RECEIVER CONTROL REGISTER, 0
1652 003764 020504 CMP R5,R4 ;WAS RECEIVER CONTROL CLEARED
1653 003766 001401 BEQ 2$
1654 003770 104002 HLT 2 ;RECEIVER CONTROL REGISTER DATA ERROR
1655 003772 104400 2$: SCOPE ;CHECK FOR ITERATIONS, LOOP

```

```

1656 003774 032737 020000 001510 CKBBO: BIT #BBBIT,DQSTAT
1657 004002 001530 BEQ CONT.0
1658
1659 ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1660 ;SET BIT8, VERIFY BIT8 WAS SET
1661 ;CLEAR BIT8, VERIFY BIT8 WAS CLEARED
1662
1663 ; TEST 15
1664 ;*****
1665 004004 012737 000015 001226 TST15: MOV #15,TSTNO
1666 004012 012737 004060 001216 MOV #TST16,NEXT
1667 004020 013703 001360 MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
1668 ;OF RECEIVER CONTROL REGISTER
1669 004024 012705 000400 MOV #BIT8,R5 ;RECEIVER CONTROL REGISTER WILL
1670 ;BE SET TO BIT8
1671 004030 010513 MOV R5,(R3) ;LOAD RECEIVER CONTROL
1672 004032 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
1673 ;IN RECEIVER CONTROL REGISTER
1674 004034 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1675 004036 001401 BEQ 1$
1676 004040 104002 HLT 2
1677 004042 040513 1$: BIC R5,(R3) ;RECEIVER CONTROL REGISTER DATA ERROR
1678 ;CLEAR BITS SET
1679 004044 011304 MOV (R3),R4 ;IN RECEIVER CONTROL REGISTER
1680 004046 005005 CLR R5 ;READ RECEIVER CONTROL REGISTER
1681 ;(R5)=EXPECTED CONTENTS
1682 004050 020504 CMP R5,R4 ;OF RECEIVER CONTROL REGISTER, 0
1683 004052 001401 BEQ 2$ ;WAS RECEIVER CONTROL CLEARED
1684 004054 104002 HLT 2
1685 004056 104400 2$: SCOPE ;RECEIVER CONTROL REGISTER DATA ERROR
1686 ;CHECK FOR ITERATIONS, LOOP
1687
1688 ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1689 ;SET BIT9, VERIFY BIT9 WAS SET
1690 ;CLEAR BIT9, VERIFY BIT9 WAS CLEARED
1691
1692 ; TEST 16
1693 004060 012737 000016 001226 TST16: MOV #16,TSTNO
1694 004066 012737 004134 001216 MOV #TST17,NEXT
1695 004074 013703 001360 MOV DQRCSR,R3 ;LOAD R3 WITH ADDRESS
1696 ;OF RECEIVER CONTROL REGISTER
1697 004100 012705 001000 MOV #BIT9,R5 ;RECEIVER CONTROL REGISTER WILL
1698 ;BE SET TO BIT9
1699 004104 010513 MOV R5,(R3) ;LOAD RECEIVER CONTROL
1700 004106 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
1701 ;IN RECEIVER CONTROL REGISTER
1702 004110 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1703 004112 001401 BEQ 1$
1704 004114 104002 HLT 2
1705 004116 040513 1$: BIC R5,(R3) ;RECEIVER CONTROL REGISTER DATA ERROR
1706 ;CLEAR BITS SET
1707 004120 011304 MOV (R3),R4 ;IN RECEIVER CONTROL REGISTER
1708 004122 005005 CLR R5 ;READ RECEIVER CONTROL REGISTER
1709 ;(R5)=EXPECTED CONTENTS
1710 004124 020504 CMP R5,R4 ;OF RECEIVER CONTROL REGISTER, 0
1711 004126 001401 BEQ 2$ ;WAS RECEIVER CONTROL CLEARED
    
```

K03

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 36
 DZDQAC.P11 16-DEC-76 11:24 BASIC RECEIVER READ/WRITE TESTS.

```

1712 004130 104002          HLT      2          ;RECEIVER CONTROL REGISTER DATA ERROR
1713 004132 104400          2$:    SCOPE          ;CHECK FOR ITERATIONS, LOOP
1714
1715          ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1716          ;SET BIT10, VERIFY BIT10 WAS SET
1717          ;CLEAR BIT10, VERIFY BIT10 WAS CLEARED
1718
1719          ; TEST 17
1720          ;*****
1721 004134 012737 000017 001226 1$T17:  MOV    #17,TSTNO
1722 004142 012737 004210 001216      MOV    #T$T20,NEXT
1723 004150 013703 001360          MOV    DQRC$R,R3          ;LOAD R3 WITH ADDRESS
1724          ;OF RECEIVER CONTROL REGISTER
1725 004154 012705 002000          MOV    #BIT10,R5        ;RECEIVER CONTROL REGISTER WILL
1726          ;BE SET TO BIT10
1727 004160 010513          MOV    R5,(R3)         ;LOAD RECEIVER CONTROL
1728 004162 011304          MOV    (R3),R4        ;(R4)=ACTUAL DATA
1729          ;IN RECEIVER CONTROL REGISTER
1730 004164 020504          CMP    R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1731 004166 001401          BEQ    1$
1732 004170 104002          HLT    2
1733 004172 040513          1$:    BIC    R5,(R3)    ;RECEIVER CONTROL REGISTER DATA ERROR
1734          ;CLEAR BITS SET
1735 004174 011304          MOV    (R3),R4        ;IN RECEIVER CONTROL REGISTER
1736 004176 005005          CLR    R5             ;READ RECEIVER CONTROL REGISTER
1737          ;(R5)=EXPECTED CONTENTS
1738 004200 020504          CMP    R5,R4          ;OF RECEIVER CONTROL REGISTER, 0
1739 004202 001401          BEQ    2$             ;WAS RECEIVER CONTROL CLEARED
1740 004204 104002          HLT    2
1741 004206 104400          2$:    SCOPE          ;RECEIVER CONTROL REGISTER DATA ERROR
1742          ;CHECK FOR ITERATIONS, LOOP
1743          ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1744          ;SET BIT11, VERIFY BIT11 WAS SET
1745          ;CLEAR BIT11, VERIFY BIT11 WAS CLEARED
1746
1747          ; TEST 20
1748          ;*****
1749 004210 012737 000020 001226 1$T20:  MOV    #20,TSTNO
1750 004216 012737 004264 001216      MOV    #T$T21,NEXT
1751 004224 013703 001360          MOV    DQRC$R,R3          ;LOAD R3 WITH ADDRESS
1752          ;OF RECEIVER CONTROL REGISTER
1753 004230 012705 004000          MOV    #BIT11,R5        ;RECEIVER CONTROL REGISTER WILL
1754          ;BE SET TO BIT11
1755 004234 010513          MOV    R5,(R3)         ;LOAD RECEIVER CONTROL
1756 004236 011304          MOV    (R3),R4        ;(R4)=ACTUAL DATA
1757          ;IN RECEIVER CONTROL REGISTER
1758 004240 020504          CMP    R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1759 004242 001401          BEQ    1$
1760 004244 104002          HLT    2
1761 004246 040513          1$:    BIC    R5,(R3)    ;RECEIVER CONTROL REGISTER DATA ERROR
1762          ;CLEAR BITS SET
1763 004250 011304          MOV    (R3),R4        ;IN RECEIVER CONTROL REGISTER
1764 004252 005005          CLR    R5             ;READ RECEIVER CONTROL REGISTER
1765          ;(R5)=EXPECTED CONTENTS
1766 004254 020504          CMP    R5,R4          ;OF RECEIVER CONTROL REGISTER, 0
1767 004256 001401          BEQ    2$             ;WAS RECEIVER CONTROL CLEARED
  
```

```

1768 004260 104002          HLT      2          ;RECEIVER CONTROL REGISTER DATA ERROR
1769 004262 104400          25:    SCOPE          ;CHECK FOR ITERATIONS, LOOP
1770 004264          CONT.O:
1771
1772          ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1773          ;SET BIT12, VERIFY BIT12 WAS SET
1774          ;CLEAR BIT12, VERIFY BIT12 WAS CLEARED
1775
1776          ; TEST 21
1777          ;*****
1778 004264 012737 000021 001226 TST21: MOV     #21,TSTNO
1779 004272 012737 004340 001216      MOV     #CHKBA1,NEXT
1780 004300 013703 001360          MOV     DQRCSR,R3          ;LOAD R3 WITH ADDRESS
1781                                     ;OF RECEIVER CONTROL REGISTER
1782 004304 012705 010000          MOV     #BIT12,R5        ;RECEIVER CONTROL REGISTER WILL
1783                                     ;BE SET TO BIT12
1784 004310 010513          MOV     R5,(R3)         ;LOAD RECEIVER CONTROL
1785 004312 011304          MOV     (R3),R4        ;(R4)=ACTUAL DATA
1786                                     ;IN RECEIVER CONTROL REGISTER
1787 004314 020504          CMP     R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1788 004316 001401          BEQ     15
1789 004320 104002          HLT     2
1790 004322 040513          15:    BIC     R5,(R3)   ;RECEIVER CONTROL REGISTER DATA ERROR
1791                                     ;CLEAR BITS SET
1792 004324 011304          MOV     (R3),R4        ;IN RECEIVER CONTROL REGISTER
1793 004326 005005          CLR     R5            ;READ RECEIVER CONTROL REGISTER
1794                                     ;(R5)=EXPECTED CONTENTS
1795 004330 020504          CMP     R5,R4          ;OF RECEIVER CONTROL REGISTER, 0
1796 004332 001401          BEQ     25            ;WAS RECEIVER CONTROL CLEARED
1797 004334 104002          HLT     2
1798 004336 104400          25:    SCOPE          ;RECEIVER CONTROL REGISTER DATA ERROR
1799                                     ;CHECK FOR ITERATIONS, LOOP
1800          ; IF DATASET CONTROL OPTION IS INSTALLED,
1801          ; TEST 22 AND TEST 23 WILL BE EXECUTED
1802
1803 004340 032737 010000 001510 CHKBA1: BIT     #BABIT,DQSTAT
1804 004346 001454          BEQ     TST24
1805
1806          ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1807          ;SET BIT13, VERIFY BIT13 WAS SET
1808          ;CLEAR BIT13, VERIFY BIT13 WAS CLEARED
1809
1810          ; TEST 22
1811          ;*****
1812 004350 012737 000022 001226 TST22: MOV     #22,TSTNO
1813 004356 012737 004424 001216      MOV     #TST23,NEXT
1814 004364 013703 001360          MOV     DQRCSR,R3          ;LOAD R3 WITH ADDRESS
1815                                     ;OF RECEIVER CONTROL REGISTER
1816 004370 012705 020000          MOV     #BIT13,R5        ;RECEIVER CONTROL REGISTER WILL
1817                                     ;BE SET TO BIT13
1818 004374 010513          MOV     R5,(R3)         ;LOAD RECEIVER CONTROL
1819 004376 011304          MOV     (R3),R4        ;(R4)=ACTUAL DATA
1820                                     ;IN RECEIVER CONTROL REGISTER
1821 004400 020504          CMP     R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1822 004402 001401          BEQ     15
1823 004404 104002          HLT     2
1824                                     ;RECEIVER CONTROL REGISTER DATA ERROR
    
```

```

1824 004406 040513      1S:  BIC      R5,(R3)          ;CLEAR BITS SET
1825                                     ;IN RECEIVER CONTROL REGISTER
1826 004410 011304      MOV      (R3),R4          ;READ RECEIVER CONTROL REGISTER
1827 004412 005005      CLR      R5              ;(R5)=EXPECTED CONTENTS
1828                                     ;OF RECEIVER CONTROL REGISTER, 0
1829 004414 020504      CMP      R5,R4          ;WAS RECEIVER CONTROL CLEARED
1830 004416 001401      BEQ     2S              ;RECEIVER CONTROL REGISTER DATA ERROR
1831 004420 104002      HLT     2               ;CHECK FOR ITERATIONS, LOOP
1832 004422 104400      2S:  SCOPE
1833                                     ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1834                                     ;SET BIT14, VERIFY BIT14 WAS SET
1835                                     ;CLEAR BIT14, VERIFY BIT14 WAS CLEARED
1836
1837
1838 ; TEST 23
1839 ;*****
1840 004424 012737 000023 001226 TST23: MOV      #23,TSTNO
1841 004432 012737 004500 001216 MOV      #TST24,NEXT
1842 004440 013703 001360          MOV      DQRCSR,R3          ;LOAD R3 WITH ADDRESS
1843                                     ;OF RECEIVER CONTROL REGISTER
1844 004444 012705 040000          MOV      #BIT14,R5         ;RECEIVER CONTROL REGISTER WILL
1845                                     ;BE SET TO BIT14
1846 004450 010513          MOV      R5,(R3)          ;LOAD RECEIVER CONTROL
1847 004452 011304          MOV      (R3),R4         ;(R4)=ACTUAL DATA
1848                                     ;IN RECEIVER CONTROL REGISTER
1849 004454 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1850 004456 001401          BEQ     1S              ;RECEIVER CONTROL REGISTER DATA ERROR
1851 004460 104002          HLT     2               ;CLEAR BITS SET
1852 004462 040513      1S:  BIC      R5,(R3)          ;IN RECEIVER CONTROL REGISTER
1853                                     ;READ RECEIVER CONTROL REGISTER
1854 004464 011304      MOV      (R3),R4          ;(R5)=EXPECTED CONTENTS
1855 004466 005005      CLR      R5              ;OF RECEIVER CONTROL REGISTER, 0
1856                                     ;WAS RECEIVER CONTROL CLEARED
1857 004470 020504          CMP      R5,R4          ;RECEIVER CONTROL REGISTER DATA ERROR
1858 004472 001401          BEQ     2S              ;CHECK FOR ITERATIONS, LOOP
1859 004474 104002          HLT     2               ;RECEIVER CONTROL REGISTER READ/WRITE TEST
1860 004476 104400      2S:  SCOPE          ;SET BIT15, VERIFY BIT15 WAS SET
1861                                     ;CLEAR BIT15, VERIFY BIT15 WAS CLEARED
1862
1863
1864
1865 ; TEST 24
1866 ;*****
1867
1868 004500 012737 000024 001226 TST24: MOV      #24,TSTNO
1869 004506 012737 004554 001216 MOV      #TST25,NEXT
1870 004514 013703 001360          MOV      DQRCSR,R3          ;LOAD R3 WITH ADDRESS
1871                                     ;OF RECEIVER CONTROL REGISTER
1872 004520 012705 100000          MOV      #BIT15,R5         ;RECEIVER CONTROL REGISTER WILL
1873                                     ;BE SET TO BIT15
1874 004524 010513          MOV      R5,(R3)          ;LOAD RECEIVER CONTROL
1875 004526 011304          MOV      (R3),R4         ;(R4)=ACTUAL DATA
1876                                     ;IN RECEIVER CONTROL REGISTER
1877 004530 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1878 004532 001401          BEQ     1S              ;RECEIVER CONTROL REGISTER DATA ERROR
1879 004534 104002          HLT     2
    
```

N03

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 39
DZDQAC.P11 16-DEC-76 11:24 BASIC RECEIVER READ/WRITE TESTS.

1880	004536	040513	1S:	BIC	R5, (R3)
1881					
1882	004540	011304		MOV	(R3), R4
1883	004542	005005		CLR	R5
1884					
1885	004544	020504		CMP	R5, R4
1886	004546	001401		BEQ	2S
1887	004550	104002		HLT	2
1888	004552	104400	2S:	SCOPE	

;CLEAR BITS SET
;IN RECEIVER CONTROL REGISTER
;READ RECEIVER CONTROL REGISTER
;(R5)=EXPECTED CONTENTS
;OF RECEIVER CONTROL REGISTER, 0
;WAS RECEIVER CONTROL CLEARED

;RECEIVER CONTROL REGISTER DATA ERROR
;CHECK FOR ITERATIONS, LOOP


```

1889
1890 ; TRANSMITTER CONTROL REGISTER READ/WRITE TEST
1891 ; SET BIT3, VERIFY BIT3 WAS SET
1892 ; CLEAR BIT3, VERIFY BIT3 WAS CLEARED
1893
1894 ; TEST 25
1895 ; *****
1896 004554 012737 000025 001226 †ST25: MOV #25,TSTNO
1897 004562 012737 004640 001216 MOV #CKBA1,NEXT
1898 004570 013703 001360 MOV DQRCR,R3 ; LOAD R3 WITH ADDRESS
1899 ; OF TRANSMITTER CONTROL REGISTER
1900 004574 012702 001400 MOV #1400,R2 ; LOAD R2 WITH 1400
1901 ; TO CLEAR UNWANTED BITS
1902 004600 012705 000010 MOV #BIT3,R5 ; TRANSMITTER CONTROL REGISTER WILL
1903 ; BE SET TO BIT3
1904 004604 010513 MOV R5,(R3) ; LOAD TRANSMITTER CONTROL
1905 004606 011304 MOV (R3),R4 ; (R4)=ACTUAL DATA
1906 ; IN TRANSMITTER CONTROL REGISTER
1907 004610 040204 BIC R2,R4 ; CLEAR UNWANTED BITS
1908 004612 020504 CMP R5,R4 ; ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1909 004614 001401 BEQ 1$
1910 004616 104003 HLT 3 ; TRANSMITTER CONTROL REGISTER DATA ERROR
1911 004620 040513 1$: BIC R5,(R3) ; CLEAR BITS SET
1912 ; IN TRANSMITTER CONTROL REGISTER
1913 004622 011304 MOV (R3),R4 ; READ TRANSMITTER CONTROL REGISTER
1914 004624 040204 BIC R2,R4 ; CLEAR UNWANTED BITS
1915 004626 005005 CLR R5 ; (R5)=EXPECTED CONTENTS
1916 ; OF TRANSMITTER CONTROL REGISTER, 0
1917 004630 020504 CMP R5,R4 ; WAS TRANSMITTER CONTROL CLEARED
1918 004632 001401 BEQ 2$
1919 004634 104003 HLT 3 ; TRANSMITTER CONTROL REGISTER DATA ERROR
1920 004636 104400 2$: SCOPE ; CHECK FOR ITERATIONS, LOOP
1921 004640 032737 010000 001510 CKBA1: BIT #BABIT,DQSTAT
1922 004646 001432 BEQ CONT.1
1923
1924 ; TRANSMITTER CONTROL REGISTER READ/WRITE TEST
1925 ; SET BIT4, VERIFY BIT4 WAS SET
1926 ; CLEAR BIT4, VERIFY BIT4 WAS CLEARED
1927
1928 ; TEST 26
1929 ; *****
1930 004650 012737 000026 001226 †ST26: MOV #26,TSTNO
1931 004656 012737 004734 001216 MOV #TST27,NEXT
1932 004664 013703 001364 MOV DQTCR,R3 ; LOAD R3 WITH ADDRESS
1933 ; OF TRANSMITTER CONTROL REGISTER
1934 004670 012702 001400 MOV #1400,R2 ; LOAD R2 WITH 1400
1935 ; TO CLEAR UNWANTED BITS
1936 004674 012705 000020 MOV #BIT4,R5 ; TRANSMITTER CONTROL REGISTER WILL
1937 ; BE SET TO BIT4
1938 004700 010513 MOV R5,(R3) ; LOAD TRANSMITTER CONTROL
1939 004702 011304 MOV (R3),R4 ; (R4)=ACTUAL DATA
1940 ; IN TRANSMITTER CONTROL REGISTER
1941 004704 040204 BIC R2,R4 ; CLEAR UNWANTED BITS
1942 004706 020504 CMP R5,R4 ; ARE EXPECTED AND RECEIVED VALUES THE SAME ?
1943 004710 001401 BEQ 1$
1944 004712 104003 HLT 3 ; TRANSMITTER CONTROL REGISTER DATA ERROR
    
```



```

2001 005044 012705 000100      MOV      #BIT6,R5      ; TRANSMITTER CONTROL REGISTER WILL
2002                                ; BE SET TO BIT6
2003 005050 010513      MOV      R5,(R3)      ; LOAD TRANSMITTER CONTROL
2004 005052 011304      MOV      (R3),R4      ; (R4)=ACTUAL DATA
2005                                ; IN TRANSMITTER CONTROL REGISTER
2006 005054 040204      BIC      R2,R4      ; CLEAR UNWANTED BITS
2007 005056 020504      CMP      R5,R4      ; ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2008 005060 001401      BEQ      1$
2009 005062 104003      HLT      3
2010 005064 040513      1$: BIC      R5,(R3)      ; TRANSMITTER CONTROL REGISTER DATA ERROR
2011                                ; CLEAR BITS SET
2012 005066 011304      MOV      (R3),R4      ; IN TRANSMITTER CONTROL REGISTER
2013 005070 040204      BIC      R2,R4      ; READ TRANSMITTER CONTROL REGISTER
2014 005072 005005      CLR      R5      ; CLEAR UNWANTED BITS
2015                                ; (R5)=EXPECTED CONTENTS
2016 005074 020504      CMP      R5,R4      ; OF TRANSMITTER CONTROL REGISTER, 0
2017 005076 001401      BEQ      2$      ; WAS TRANSMITTER CONTROL CLEARED
2018 005100 104003      HLT      3
2019 005102 104400      2$: SCOPE      ; TRANSMITTER CONTROL REGISTER DATA ERROR
2020                                ; CHECK FOR ITERATIONS, LOOP
2021                                ; TRANSMITTER CONTROL REGISTER READ/WRITE TEST
2022                                ; SET BIT7, VERIFY BIT7 WAS SET
2023                                ; CLEAR BIT7, VERIFY BIT7 WAS CLEARED
2024
2025                                ; TEST 31
2026                                ; *****
2027 005104 012737 000031 001226  TST31: MOV      #31,TSTNO
2028 005112 012737 005170 001216  MOV      #CHKBA2,NEXT
2029 005120 013703 001364      MOV      DATCSR,R3      ; LOAD R3 WITH ADDRESS
2030                                ; OF TRANSMITTER CONTROL REGISTER
2031 005124 012702 001400      MOV      #1400,R2      ; LOAD R2 WITH 1400
2032                                ; TO CLEAR UNWANTED BITS
2033 005130 012705 000200      MOV      #BIT7,R5      ; TRANSMITTER CONTROL REGISTER WILL
2034                                ; BE SET TO BIT7
2035 005134 010513      MOV      R5,(R3)      ; LOAD TRANSMITTER CONTROL
2036 005136 011304      MOV      (R3),R4      ; (R4)=ACTUAL DATA
2037                                ; IN TRANSMITTER CONTROL REGISTER
2038 005140 040204      BIC      R2,R4      ; CLEAR UNWANTED BITS
2039 005142 020504      CMP      R5,R4      ; ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2040 005144 001401      BEQ      1$
2041 005146 104003      HLT      3
2042 005150 040513      1$: BIC      R5,(R3)      ; TRANSMITTER CONTROL REGISTER DATA ERROR
2043                                ; CLEAR BITS SET
2044 005152 011304      MOV      (R3),R4      ; IN TRANSMITTER CONTROL REGISTER
2045 005154 040204      BIC      R2,R4      ; READ TRANSMITTER CONTROL REGISTER
2046 005156 005005      CLR      R5      ; CLEAR UNWANTED BITS
2047                                ; (R5)=EXPECTED CONTENTS
2048 005160 020504      CMP      R5,R4      ; OF TRANSMITTER CONTROL REGISTER, 0
2049 005162 001401      BEQ      2$      ; WAS TRANSMITTER CONTROL CLEARED
2050 005164 104003      HLT      3
2051 005166 104400      2$: SCOPE      ; TRANSMITTER CONTROL REGISTER DATA ERROR
2052                                ; CHECK FOR ITERATIONS, LOOP
2053                                ; IF DATASET CONTROL OPTION IS INSTALLED,
2054                                ; TEST 32 AND TEST 33 WILL BE EXECUTED
2055
2056 005170 032737 010000 001510  CHKBA2: BIT      #BABIT,DQSTAT
    
```

```

2057 005176 001002      BNE      +6
2058 005200 000137 005626  JMP      CHKBA3
2059
2060      ; TRANSMITTER CONTROL REGISTER READ/WRITE TEST
2061      ; SET BIT8 IN TRANSMITTER CONTROL REGISTER
2062      ; VERIFY THAT BIT8,BIT 10 AND BIT11 ARE SET
2063      ; CLEAR BIT8
2064      ; VERIFY THAT BIT8,BIT 10 AND BIT11 WERE CLEARED
2065
2066      ; TEST 32
2067      ; *****
2068 005204 012737 000032 001226  TST32:  MOV      #32,TSTNO
2069 005212 012737 005366 001216      MOV      #TST33,NEXT
2070 005220 013703 001364      MOV      DQTCSR,R3      ; ADDRESS OF TRANSMITTER CONTROL REGISTER
2071 005224 012705 000400      MOV      #BIT8,R5      ; (R5)=BIT8
2072 005230 010513      MOV      R5,(R3)      ; LOAD TRANSMITTER CONTROL REGISTER
2073 005232 112777 000012 174130  MOVB     #12,DDQREG     ; TRY TO SEL MISC REGISTER
2074 005240 012777 000002 174124  MOV      #2,DDQSEC     ; TRY TO SET AUTO/STEP
2075 005246 005277 174120      INC      DDQSEC        ; CLOCK UP!!
2076 005252 005377 174114      DEC      DDQSEC        ; CLOCK DN!!
2077 005256 004737 005524      JSR      PC_DELAY     ; DELAY FOR REAL CABLE.
2078 005262 011304      MOV      (R3),R4      ; READ TRANSMITTER CONTROL REGISTER
2079 005264 032. .7 040000 001510  BIT      #JUMBIT,DQSTAT ; IS TEST JUMPER INSTALLED
2080 005272 001404      BEQ      .+12         ; BR IF NO JUMPER
2081 005274 052705 006000      BIS      #BIT10+BIT11,R5 ; EXPECT BIT8,BIT 10 AND BIT11
2082 005300 052705 100000      BIS      #BIT15,R5   ; ADJUST EXPECTED RESULTS.
2083      ; FOR DATA SET INTR
2084 005304 020504      CMP      R5,R4      ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
2085 005306 001401      BEQ      15
2086 005310 104003      HLT      3
2087 005312 042713 000400 15:      BIC      #BIT8,(R3)   ; TRANSMITTER CONTROL REGISTER DATA ERROR
2088 005316 112777 000012 174044  MOVB     #12,DDQREG     ; CLEAR BIT8
2089 005324 012777 000002 174040  MOV      #2,DDQSEC     ; TRY AND SELECT THE MISC REG
2090 005332 005277 174034      INC      DDQSEC        ; TRY AND SET AUTO/STEP TO STEP
2091 005336 005377 174030      DEC      DDQSEC        ; SET CLOCK UP!
2092 005342 004737 005524      JSR      PC_DELAY     ; CLOCK DOWN!
2093 005346 011304      MOV      (R3),R4      ; DELAY.
2094 005350 042704 100000      BIC      #BIT15,R4    ; READ TRANSMITTER CONTROL REGISTER
2095 005354 005005      CLR      R5          ; IGNORE BIT 15 FOR NOW.
2096 005356 020504      CMP      R5,R4      ; EXPECT 0
2097 005360 001401      BEQ      25          ; WAS TRANSMITTER CONTROL REGISTER CLEARED
2098 005362 104003      HLT      3          ; BR IF GOOD
2099 005364 104400 25:      SCOPE      ; TRANSMITTER CONTROL REGISTER DATA ERROR
2100      ; CHECK FOR ITERATIONS, LOOP
2101      ; TRANSMITTER CONTROL REGISTER READ/WRITE TEST
2102      ; SET BIT9 IN TRANSMITTER CONTROL REGISTER
2103      ; VERIFY THAT BIT9,BIT12 AND BIT13 ARE SET
2104      ; CLEAR BIT9
2105      ; VERIFY THAT BIT9,BIT12 AND BIT13 WERE CLEARED
2106
2107      ; TEST 33
2108      ; *****
2109 005366 012737 000033 001226  TST33:  MOV      #33,TSTNO
2110 005374 012737 005540 001216      MOV      #TST34,NEXT
2111 005402 013703 001364      MOV      DQTCSR,R3      ; ADDRESS OF TRANSMITTER CONTROL REGISTER
2112 005406 012705 001000      MOV      #BIT9,R5      ; (R5)=BIT9

```

```

2113 005412 010513          MOV    R5,(R3)          ;LOAD TRANSMITTER CONTROL REGISTER
2114 005414 004737 005524    JSR    PC,DELAY        ;DELAY FOR REAL CABLE.
2115 005420 011304          MOV    (R3),R4         ;READ TRANSMITTER CONTROL REGISTER
2116 005422 032737 040000 001510 BIT    #JUMBIT,DQSTAT  ;IS TEST JUMPER INSTALLED
2117 005430 001404          BEQ    .+12           ;BR IF NO JUMPER
2118 005432 052705 030000    BIS    #BIT12+BIT13,R5 ;EXPECT BIT9,BIT12 AND BIT13
2119 005436 052705 100000    BIS    #BIT15,R5      ;ADJUST EXPECTED RESULTS.
2120                                     ;FOR DATA SET INTR
2121 005442 020504          CMP    R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2122 005444 001401          BEQ    1$            ;
2123 005446 104003          HLT    3              ;TRANSMITTER CONTROL REGISTER DATA ERROR
2124 005450 042713 001000    BIC    #BIT9,(R3)     ;CLEAR BIT9
2125 005454 112777 000012 173706 1$:  MOVB  #12,DQREG        ;TRY AND SELECT THE MISC REG
2126 005462 012777 000002 173702    MOV    #2,DQSEC       ;TRY AND SET AUTO/STEP TO STEP
2127 005470 005277 173676    INC    DQSEC          ;SET CLOCK UP!
2128 005474 005377 173672    DEC    DQSEC          ;CLOCK DOWN!
2129 005500 004737 005524    JSR    PC,DELAY        ;DELAY.
2130 005504 011304          MOV    (R3),R4         ;READ TRANSMITTER CONTROL REGISTER
2131 005506 042704 100000    BIC    #BIT15,R4     ;IGNORE BIT 15 FOR NOW.
2132 005512 005005          CLR    R5             ;EXPECT 0
2133 005514 020504          CMP    R5,R4          ;WAS TRANSMITTER CONTROL REGISTER CLEARED
2134 005516 001401          BEQ    2$            ;BR IF GOOD
2135 005520 104003          HLT    3              ;TRANSMITTER CONTROL REGISTER DATA ERROR
2136 005522 104400          SCOPE 2$:           ;CHECK FOR ITERATIONS, LOOP
2137 005524 010046    DELAY: MOV    R0,-(SP)    ;SAVE R0 ON THE STACK
2138 005526 005000          CLR    R0             ;ZERO R0
2139 005530 105200          INCB  R0             ;DELAY...
2140 005532 100376          BPL    .-2            ;DONE YET?
2141 005534 012600          MOV    (SP)+,R0      ;RESTORE R0
2142 005536 000207          RTS    PC            ;RETURN.
2143
2144                                     ;READ WRITE TEST OF BIT 15 OF TRANSMITTER CSR.
2145                                     ;SET BIT 15 VERIFY SET; CLEAR BIT 15 VERIFY CLEARED.
2146
2147                                     ; TEST 34
2148                                     ;*****
2149 005540 012737 000034 001226 1$TST34: MOV    #34,TSTNO
2150 005546 012737 005626 001216    MOV    #CHKBA3,NEXT
2151 005554 013703 001364    MOV    DQTCR,R3      ;GET TX CSR
2152 005560 005005          CLR    R5             ;CLR TX CSR
2153 005562 005013          CLR    (R3)          ;DO IT AGAIN.
2154 005564 005013          CLR    (R3)          ;READ TX CSR
2155 005566 011304          MOV    (R3),R4
2156 005570 001401          BEQ    1$            ;TX CSR NO ZERO.
2157 005572 104003          HLT    3              ;SET EXPECTED.
2158 005574 052705 100000    1$:  BIS    #BIT15,R5     ;SET BIT 15
2159 005600 010513          MOV    R5,(R3)       ;READ CSR.
2160 005602 011304          MOV    (R3),R4       ;EXPECTED=RECEIVED?
2161 005604 020504          CMP    R5,R4
2162 005606 001401          BEQ    2$            ;TRANSMITTER DATA ERROR.
2163 005610 104003          HLT    3              ;SET EXPECTED
2164 005612 005005    2$:  CLR    R5             ;CLEAR BIT 15
2165 005614 010513          MOV    R5,(R3)       ;READ CSR
2166 005616 011304          MOV    (R3),R4
2167 005620 001401          BEQ    .+4           ;TX CSR NOT ZERO
2168 005622 104003          HLT    3
    
```

2169 005624 104400

SCOPE

2170

2171

2172

2173

2174

2175

2176

2177

2178

2179

2180

2181

2182

2183

2184

2185

2186

2187

2188

2189

2190

2191

2192

2193

2194

2195

2196

2197

2198

2199

2200

2201

2202

2203

2204

2205

2206

2207

2208

2209

2210

2211

2212

2213

2214

2215

2216

2217

2218

2219

2220

2221

2222

2223

2224

005626 032737 010000 001510
005634 001017

CHKBA3: BIT #BABIT,DQSTAT
BNE TST36

; IF DATASET CONTROL OPTION IS NOT INSTALLED,
; TEST 35 WILL BE EXECUTED
; IF DATASET CONTROL OPTION IS NOT INSTALLED,
; THE WHOLE UPPER BYTE OF THE TX CSR SHOULD BE
; EQUAL TO ZERO.

; TEST 35

TST35: MOV #35,TSTNO
MOV #TST36,NEXT
MOV DQTCR,R3
CLR R5
MOV #177400,(R3)
MOV (R3),R4
BEQ +4
HLT 3

; LOAD REG
; SET EXPECTED.
; SET UPPER BYTE TO ALL 1'S
; READ IT BACK.
; TRANSMITTER CSR NOT ZERO.

IS: SCOPE

; ERROR REGISTER READ/WRITE TEST
; SET BIT0, VERIFY BIT0 AND BIT15 WERE SET
; CLEAR BIT0, VERIFY BIT0 AND BIT15 WERE CLEARED

; TEST 36

TST36: MOV #36,TSTNO
MOV #TST37,NEXT
MOV DQERR,R3
MOV #60000,R2
MOV #BIT0,R5
MOV R5,(R3)
MOV (R3),R4
BIS #BIT15,R5
BIC R2,R4
CMP R5,R4
BEQ 1\$
HLT 4

; LOAD R3 WITH ADDRESS
; OF ERROR REGISTER
; LOAD R2 WITH 60000
; TO CLEAR UNWANTED BITS
; ERROR REGISTER WILL
; BE SET TO BIT0
; LOAD ERROR
; (R4)=ACTUAL DATA
; IN ERROR REGISTER
; EXPECT BIT15 TO BE SET ALSO
; CLEAR UNWANTED BITS
; ARE EXPECTED AND RECEIVED VALUES THE SAME ?

IS: BIC R5,(R3)

; ERROR REGISTER DATA ERROR
; CLEAR BITS SET
; IN ERROR REGISTER
; READ ERROR REGISTER
; CLEAR UNWANTED BITS
; (R5)=EXPECTED CONTENTS
; OF ERROR REGISTER, 0
; WAS ERROR CLEARED

MOV (R3),R4
BIC R2,R4
CLR R5
CMP R5,R4
BEQ 2\$
HLT 4

; ERROR REGISTER DATA ERROR

H04

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 46
DZDQAC.P11 16-DEC-76 11:24 ERROR REGISTER READ/WRITE TESTS.

```

2225 005762 104400 25: SCOPE ;CHECK FOR ITERATIONS, LOOP
2226
2227 ;ERROR REGISTER READ/WRITE TEST
2228 ;SET BIT1, VERIFY BIT1 AND BIT15 WERE SET
2229 ;CLEAR BIT1, VERIFY BIT1 AND BIT15 WERE CLEARED
2230
2231 ; TEST 37
2232 ;*****
2233 005764 012737 000037 001226 †ST37: MOV #37,TSTNO
2234 005772 012737 006054 001216 MOV #TST40,NEXT
2235 006000 013703 001366 MOV DQERR,R3 ;LOAD R3 WITH ADDRESS
2236 ;OF ERROR REGISTER
2237 006004 012702 060000 MOV #60000,R2 ;LOAD R2 WITH 60000
2238 ;TO CLEAR UNWANTED BITS
2239 006010 012705 000002 MOV #BIT1,R5 ;ERROR REGISTER WILL
2240 ;BE SET TO BIT1
2241 006014 010513 MOV R5,(R3) ;LOAD ERROR
2242 006016 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
2243 ;IN ERROR REGISTER
2244 006020 052705 100000 BIS #BIT15,R5 ;EXPECT BIT15 TO BE SET ALSO
2245 006024 040204 BIC R2,R4 ;CLEAR UNWANTED BITS
2246 006026 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2247 006030 001401 BEQ 1$
2248 006032 104004 HLT 4 ;ERROR REGISTER DATA ERROR
2249 006034 040513 15: BIC R5,(R3) ;CLEAR BITS SET
2250 ;IN ERROR REGISTER
2251 006036 011304 MOV (R3),R4 ;READ ERROR REGISTER
2252 006040 040204 BIC R2,R4 ;CLEAR UNWANTED BITS
2253 006042 005005 CLR R5 ;(R5)=EXPECTED CONTENTS
2254 ;OF ERROR REGISTER, 0
2255 006044 020504 CMP R5,R4 ;WAS ERROR CLEARED
2256 006046 001401 BEQ 25
2257 006050 104004 HLT 4 ;ERROR REGISTER DATA ERROR
2258 006052 104400 25: SCOPE ;CHECK FOR ITERATIONS, LOOP
2259
2260 ;ERROR REGISTER READ/WRITE TEST
2261 ;SET BIT2, VERIFY BIT2 AND BIT15 WERE SET
2262 ;CLEAR BIT2, VERIFY BIT2 AND BIT15 WERE CLEARED
2263
2264 ; TEST 40
2265 ;*****
2266 006054 012737 000040 001226 †ST40: MOV #40,TSTNO
2267 006062 012737 006144 001216 MOV #TST41,NEXT
2268 006070 013703 001366 MOV DQERR,R3 ;LOAD R3 WITH ADDRESS
2269 ;OF ERROR REGISTER
2270 006074 012702 060000 MOV #60000,R2 ;LOAD R2 WITH 60000
2271 ;TO CLEAR UNWANTED BITS
2272 006100 012705 000004 MOV #BIT2,R5 ;ERROR REGISTER WILL
2273 ;BE SET TO BIT2
2274 006104 010513 MOV R5,(R3) ;LOAD ERROR
2275 006106 011304 MOV (R3),R4 ;(R4)=ACTUAL DATA
2276 ;IN ERROR REGISTER
2277 006110 052705 100000 BIS #BIT15,R5 ;EXPECT BIT15 TO BE SET ALSO
2278 006114 040204 BIC R2,R4 ;CLEAR UNWANTED BITS
2279 006116 020504 CMP R5,R4 ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2280 006120 001401 BEQ 1$

```

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 47
DZDQAC.P11 16-DEC-76 11:24 ERROR REGISTER READ/WRITE TESTS.

```

2281 006122 104004          HLT      4          ;ERROR REGISTER DATA ERROR
2282 006124 040513      1$: BIC      R5,(R3)    ;CLEAR BITS SET
2283                                     ;IN ERROR REGISTER
2284 006126 011304          MOV      (R3),R4    ;READ ERROR REGISTER
2285 006130 040204          BIC      R2,R4      ;CLEAR UNWANTED BITS
2286 006132 005005          CLR      R5        ;(R5)=EXPECTED CONTENTS
2287                                     ;OF ERROR REGISTER, 0
2288 006134 020504          CMP      R5,R4     ;WAS ERROR CLEARED
2289 006136 001401          BEQ      2$
2290 006140 104004          HLT      4
2291 006142 104400      2$: SCOPE          ;ERROR REGISTER DATA ERROR
2292                                     ;CHECK FOR ITERATIONS, LOOP
2293                                     ;ERROR REGISTER READ/WRITE TEST
2294                                     ;SET BIT3, VERIFY BIT3 AND BIT15 WERE SET
2295                                     ;CLEAR BIT3, VERIFY BIT3 AND BIT15 WERE CLEARED
2296
2297 ; TEST 41
2298 ;*****
2299 006144 012737 000041 001226 1$T41: MOV      #41,TSTNO
2300 006152 012737 006234 001216  MOV      #TST42,NEXT
2301 006160 013703 001366  MOV      DQERR,R3          ;LOAD R3 WITH ADDRESS
2302                                     ;OF ERROR REGISTER
2303 006164 012702 060000  MOV      #60000,R2       ;LOAD R2 WITH 60000
2304                                     ;TO CLEAR UNWANTED BITS
2305 006170 012705 000010  MOV      #BIT3,R5        ;ERROR REGISTER WILL
2306                                     ;BE SET TO BIT3
2307 006174 010513          MOV      R5,(R3)        ;LOAD ERROR
2308 006176 011304          MOV      (R3),R4        ;(R4)=ACTUAL DATA
2309                                     ;IN ERROR REGISTER
2310 006200 052705 100000  BIS      #BIT15,R5       ;EXPECT BIT15 TO BE SET ALSO
2311 006204 040204          BIC      R2,R4          ;CLEAR UNWANTED BITS
2312 006206 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2313 006210 001401          BEQ      1$
2314 006212 104004          HLT      4
2315 006214 040513      1$: BIC      R5,(R3)    ;ERROR REGISTER DATA ERROR
2316                                     ;CLEAR BITS SET
2317 006216 011304          MOV      (R3),R4        ;IN ERROR REGISTER
2318 006220 040204          BIC      R2,R4          ;READ ERROR REGISTER
2319 006222 005005          CLR      R5            ;CLEAR UNWANTED BITS
2320                                     ;(R5)=EXPECTED CONTENTS
2321 006224 020504          CMP      R5,R4         ;OF ERROR REGISTER, 0
2322 006226 001401          BEQ      2$            ;WAS ERROR CLEARED
2323 006230 104004          HLT      4
2324 006232 104400      2$: SCOPE          ;ERROR REGISTER DATA ERROR
2325                                     ;CHECK FOR ITERATIONS, LOOP
2326                                     ;ERROR REGISTER READ/WRITE TEST
2327                                     ;SET BIT4, VERIFY BIT4 AND BIT15 WERE SET
2328                                     ;CLEAR BIT4, VERIFY BIT4 AND BIT15 WERE CLEARED
2329
2330 ; TEST 42
2331 ;*****
2332 006234 012737 000042 001226 1$T42: MOV      #42,TSTNO
2333 006242 012737 006324 001216  MOV      #TST43,NEXT
2334 006250 013703 001366  MOV      DQERR,R3          ;LOAD R3 WITH ADDRESS
2335                                     ;OF ERROR REGISTER
2336 006254 012702 060000  MOV      #60000,R2       ;LOAD R2 WITH 60000

```



```

2449 006556 011304          MOV      (R3),R4          ;READ ERROR REGISTER
2450 006560 040204          BIC      R2,R4           ;CLEAR UNWANTED BITS
2451 006562 005005          CLR      R5              ;(R5)=EXPECTED CONTENTS
2452                                     ;OF ERROR REGISTER, 0
2453 006564 020504          CMP      R5,R4           ;WAS ERROR CLEARED
2454 006566 001401          BEQ      25
2455 006570 104004          HLT      4
2456 006572 104400          25: SCOPE                ;ERROR REGISTER DATA ERROR
                                     ;CHECK FOR ITERATIONS, LOOP
2457                                     ;
2458                                     ;ERROR REGISTER READ/WRITE TEST
2459                                     ;SET BIT8, VERIFY BIT8 WAS SET
2460                                     ;CLEAR BIT8, VERIFY BIT8 WAS CLEARED
2461                                     ;
2462                                     ; TEST 46
2463                                     ;*****
2464 006574 012737 000046 001226  TST46: MOV      #46,TSTNO
2465 006602 012737 006660 001216  MOV      #TST47,NEXT
2466 006610 013703 001366          MOV      DGERR,R3
2467                                     ;LOAD R3 WITH ADDRESS
2468 006614 012702 060000          MOV      #60000,R2      ;OF ERROR REGISTER
2469                                     ;LOAD R2 WITH 60000
2470 006620 012705 000400          MOV      #BIT8,R5      ;TO CLEAR UNWANTED BITS
2471                                     ;ERROR REGISTER WILL
2472 006624 010513          MOV      R5,(R3)       ;BE SET TO BIT8
2473 006626 011304          MOV      (R3),R4       ;LOAD ERROR
2474                                     ;(R4)=ACTUAL DATA
2475 006630 040204          BIC      R2,R4         ;IN ERROR REGISTER
2476 006632 020504          CMP      R5,R4         ;CLEAR UNWANTED BITS
2477 006634 001401          BEQ      15            ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2478 006636 104004          HLT      4
2479 006640 040513          15: BIC      R5,(R3)    ;ERROR REGISTER DATA ERROR
2480                                     ;CLEAR BITS SET
2481 006642 011304          MOV      (R3),R4       ;IN ERROR REGISTER
2482 006644 040204          BIC      R2,R4         ;READ ERROR REGISTER
2483 006646 005005          CLR      R5              ;CLEAR UNWANTED BITS
2484                                     ;(R5)=EXPECTED CONTENTS
2485 006650 020504          CMP      R5,R4         ;OF ERROR REGISTER, 0
2486 006652 001401          BEQ      25            ;WAS ERROR CLEARED
2487 006654 104004          HLT      4
2488 006656 104400          25: SCOPE                ;ERROR REGISTER DATA ERROR
2489                                     ;CHECK FOR ITERATIONS, LOOP
2490                                     ;
2491                                     ;ERROR REGISTER READ/WRITE TEST
2492                                     ;SET BIT9, VERIFY BIT9 WAS SET
2493                                     ;CLEAR BIT9, VERIFY BIT9 WAS CLEARED
2494                                     ;
2495                                     ; TEST 47
2496                                     ;*****
2497 006660 012737 000047 001226  TST47: MOV      #47,TSTNO
2498 006666 012737 006744 001216  MOV      #TST50,NEXT
2499 006674 013703 001366          MOV      DGERR,R3
2500 006700 012702 060000          MOV      #60000,R2      ;LOAD R3 WITH ADDRESS
2501                                     ;OF ERROR REGISTER
2502 006704 012705 001000          MOV      #BIT9,R5      ;LOAD R2 WITH 60000
2503                                     ;TO CLEAR UNWANTED BITS
2504 006710 010513          MOV      R5,(R3)       ;ERROR REGISTER WILL
                                     ;BE SET TO BIT9
                                     ;LOAD ERROR

```

```

2505 006712 011304      MOV      (R3),R4      ;(R4)=ACTUAL DATA
2506                                     ;IN ERROR REGISTER
2507 006714 040204      BIC      R2,R4      ;CLEAR UNWANTED BITS
2508 006716 020504      CMP      R5,R4      ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2509 006720 001401      BEQ      1$
2510 006722 104004      HLT      4
2511 006724 040513      1$: BIC      R5,(R3)    ;ERROR REGISTER DATA ERROR
2512                                     ;CLEAR BITS SET
2513                                     ;IN ERROR REGISTER
2514 006726 011304      MOV      (R3),R4      ;READ ERROR REGISTER
2515 006730 040204      BIC      R2,R4      ;CLEAR UNWANTED BITS
2516 006732 005005      CLR      R5          ;(R5)=EXPECTED CONTENTS
2517                                     ;OF ERROR REGISTER, 0
2518 006734 020504      CMP      R5,R4      ;WAS ERROR CLEARED
2519 006736 001401      BEQ      2$
2520 006740 104004      HLT      4
2521 006742 104400      2$: SCOPE          ;ERROR REGISTER DATA ERROR
2522                                     ;CHECK FOR ITERATIONS, LOOP
2523                                     ;ERROR REGISTER READ/WRITE TEST
2524                                     ;SET BIT10, VERIFY BIT10 WAS SET
2525                                     ;CLEAR BIT10, VERIFY BIT10 WAS CLEARED
2526
2527 ; TEST 50
2528 ;*****
2529 006744 012737 000050 001226 1$TST50: MOV      #50,TSTNO
2530 006752 012737 007030 001216      MOV      #TST51,NEXT
2531 006760 013703 001366      MOV      DGERR,R3    ;LOAD R3 WITH ADDRESS
2532                                     ;OF ERROR REGISTER
2533 006764 012702 060000      MOV      #60000,R2   ;LOAD R2 WITH 60000
2534                                     ;TO CLEAR UNWANTED BITS
2535 006770 012705 002000      MOV      #BIT10,R5   ;ERROR REGISTER WILL
2536                                     ;BE SET TO BIT10
2537 006774 010513      MOV      R5,(R3)     ;LOAD ERROR
2538 006776 011304      MOV      (R3),R4     ;(R4)=ACTUAL DATA
2539                                     ;IN ERROR REGISTER
2540 007000 040204      BIC      R2,R4      ;CLEAR UNWANTED BITS
2541 007002 020504      CMP      R5,R4      ;ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2542 007004 001401      BEQ      1$
2543 007006 104004      HLT      4
2544 007010 040513      1$: BIC      R5,(R3)    ;ERROR REGISTER DATA ERROR
2545                                     ;CLEAR BITS SET
2546                                     ;IN ERROR REGISTER
2547 007012 011304      MOV      (R3),R4      ;READ ERROR REGISTER
2548 007014 040204      BIC      R2,R4      ;CLEAR UNWANTED BITS
2549 007016 005005      CLR      R5          ;(R5)=EXPECTED CONTENTS
2550                                     ;OF ERROR REGISTER, 0
2551 007020 020504      CMP      R5,R4      ;WAS ERROR CLEARED
2552 007022 001401      BEQ      2$
2553 007024 104004      HLT      4
2554 007026 104400      2$: SCOPE          ;ERROR REGISTER DATA ERROR
2555                                     ;CHECK FOR ITERATIONS, LOOP
2556                                     ;ERROR REGISTER READ/WRITE TEST
2557                                     ;SET BIT11, VERIFY BIT11 WAS SET
2558                                     ;CLEAR BIT11, VERIFY BIT11 WAS CLEARED
2559
2560 ; TEST 51
2561 ;*****
2562 007030 012737 000051 001226 1$TST51: MOV      #51,TSTNO

```

N04

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 52
DZDQAC.P11 16-DEC-76 11:24 ERROR REGISTER READ/WRITE TESTS.

2561	007036	012737	007114	001216	MOV	#TST52,NEXT	
2562	007044	013703	001366		MOV	DQERR,R3	:LOAD R3 WITH ADDRESS
2563							:OF ERROR REGISTER
2564	007050	012702	060000		MOV	#60000,R2	:LOAD R2 WITH 60000
2565							:TO CLEAR UNWANTED BITS
2566	007054	012705	004000		MOV	#BIT11,R5	:ERROR REGISTER WILL
2567							:BE SET TO BIT11
2568	007060	010513			MOV	R5,(R3)	:LOAD ERROR
2569	007062	011304			MOV	(R3),R4	:(R4)=ACTUAL DATA
2570							:IN ERROR REGISTER
2571	007064	040204			BIC	R2,R4	:CLEAR UNWANTED BITS
2572	007066	020504			CMP	R5,R4	:ARE EXPECTED AND RECEIVED VALUES THE SAME ?
2573	007070	001401			BEQ	1\$	
2574	007072	104004			HLT	4	:ERROR REGISTER DATA ERROR
2575	007074	040513		1\$:	BIC	R5,(R3)	:CLEAR BITS SET
2576							:IN ERROR REGISTER
2577	007076	011304			MOV	(R3),R4	:READ ERROR REGISTER
2578	007100	040204			BIC	R2,R4	:CLEAR UNWANTED BITS
2579	007102	005005			CLR	R5	:(R5)=EXPECTED CONTENTS
2580							:OF ERROR REGISTER, 0
2581	007104	020504			CMP	R5,R4	:WAS ERROR CLEARED
2582	007106	001401			BEQ	2\$	
2583	007110	104004			HLT	4	:ERROR REGISTER DATA ERROR
2584	007112	104400		2\$:	SCOPE		:CHECK FOR ITERATIONS, LOOP

2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640

007114 012737 000052 001226
 007122 012737 007240 001216
 007130 012737 007202 001220
 007136 012700 000010
 007142 012701 020016
 007146 005003
 007150 012105
 007152 110377 172212
 007156 010577 172210
 007162 005203
 007164 005300
 007166 001370
 007170 012700 000010
 007174 012701 020016
 007200 005003
 007202 110377 172162
 007206 017704 172160
 007212 011105
 007214 020504
 007216 001401
 007220 104005
 007222 104401
 007224 005203
 007226 062701 000002
 007232 005300
 007234 001362
 007236 104400

```

; IF CHARACTER DETECT AND BCC OPTIONS ARE OR ARE NOT
; INSTALLED, TEST 52 WILL BE EXECUTED

; SECONDARY REGISTER ADDRESSING TEST

; TEST 52
; *****
TST52: MOV #52, TSTNO
      MOV #OPT1, NEXT
      MOV #2$, LOCK
      MOV #10, RO
      MOV #DATAB, R1
      CLR R3
      ; GET POINTER TO ADDRESS TEST DATA
      ; FIRST SECONDARY REGISTER=0
1$: MOV (R1)+, R5
      ; GET DATA TO BE LOADE
      MOV R3, JDDREG
      ; SELECTED SECONDARY REGISTER
      MOV R5, JDDSEC
      ; LOAD SECONDARY REGISTER
      INC R3
      ; ADDRESS OF NEXT SECONDARY REGISTER
      DEC RO
      ; CONTINUE IF NOT DONE
      BNE 1$
      MOV #10, RO
      MOV #DATAB, R1
      ; GET POINTER TO ADDRESS TEST DATA
      ; FIRST SECONDARY REGISTER TO BE CHECKED
2$: CLR R3
      ; SELECT SECONDARY REGISTER
      MOV R3, JDDREG
      ; READ SECONDARY REGISTER
      MOV JDDSEC, R4
      ; GET TEST DATA
      MOV (R1), R5
      ; CHECK DATA
      CMP R5, R4
      BEQ 3$
      HLT 5
      ; SECONDARY REGISTER ADDRESSING ERROR
3$: SCOP1
      ; CHECK FOR LOOP ON CURENT ADDRESS
      INC R3
      ; UPDATE ADDRESS
      ADD #2, R1
      ; UPDATE REGISTER DATA POINTER
      DEC RO
      ; CONTINUE IF NOT DONE
      BNE 2$
4$: SCOPE
      ; CHECK FOR ITERATIONS, LOOP

; IF CHARACTER DETECT OPTION IS INSTALLED,
; TEST 53 WILL BE EXECUTED

OPT1: BIT #8BIT, DQSTAT
      BNE .+14
      MOV #OPT2X, RETURN
      JMP JRETURN

; SECONDARY REGISTER ADDRESSING TEST

; TEST 53
; *****
TST53: MOV #53, TSTNO
      MOV #EOPT1, NEXT
      MOV #2$, LOCK
      MOV #13, RO
      MOV #DATAB, R1
      CLR R3
      ; GET POINTER TO ADDRESS TEST DATA
      ; FIRST SECONDARY REGISTER=0
1$: MOV (R1)+, R5
      ; GET DATA TO BE LOADE
    
```

2641	007320	110377	172044			MOVB	R3, 2DQREG		; SELECTED SECONDARY REGISTER
2642	007324	010577	172042			MOV	R5, 2DQSEC		; LOAD SECONDARY REGISTER
2643	007330	005203				INC	R3		; ADDRESS OF NEXT SECONDARY REGISTER
2644	007332	005300				DEC	R0		; CONTINUE IF NOT DONE
2645	007334	001370				BNE	1\$		
2646	007336	012700	000015			MOV	#13, R0		
2647	007342	012701	020016			MOV	#DATAB, R1		; GET POINTER TO ADDRESS TEST DATA
2648	007346	005003				CLR	R3		; FIRST SECONDARY REGISTER TO BE CHECKED
2649	007350	110377	172014	2\$:		MOVB	R3, 2DQREG		; SELECT SECONDARY REGISTER
2650	007354	017704	172012			MOV	2DQSEC, R4		; READ SECONDARY REGISTER
2651	007360	011105				MOV	(R1), R5		; GET TEST DATA
2652	007362	020504				CMP	R5, R4		; CHECK DATA
2653	007364	001401				BEQ	3\$		
2654	007366	104005				HLT	5		; SECONDARY REGISTER ADDRESSING ERROR
2655	007370	104401		3\$:		SCOPI			; CHECK FOR LOOP ON CURENT ADDRESS
2656	007372	005203				INC	R3		; UPDATE ADDRESS
2657	007374	062701	000002			ADD	#2, R1		; UPDATE REGISTER DATA POINTER
2658	007400	005300				DEC	R0		; CONTINUE IF NOT DONE
2659	007402	001362				BNE	2\$		
2660	007404	104400			4\$:	SCOPE			; CHECK FOR ITERATIONS, LOOP
2661	007406	032737	002000	001510	EOPT1:	BIT	#RABBIT, DQSTAT		
2662	007414	001005				BNE	.+14		
2663	007416	012737	007554	001214		MOV	#OPT2X, RETURN		
2664	007424	000177	171564			JMP	2RETURN		

; IF CHARACTER DETECT AND BCC OPTIONS ARE INSTALLED,
 ; EXECUTE TEST 54

; SECONDARY REGISTER ADDRESSING TEST

; TEST 54
 ; *****

2675	007430	012737	000054	001226	TST54:	MOV	#54, TSTNO		
2676	007436	012737	007554	001216		MOV	#TST55, NEXT		
2677	007444	012737	007516	001220		MOV	#2\$, LOCK		
2678	007452	012700	000020			MOV	#16, R0		
2679	007456	012701	020016			MOV	#DATAB, R1		; GET POINTER TO ADDRESS TEST DATA
2680	007462	005003				CLR	R3		; FIRST SECONDARY REGISTER=0
2681	007464	012105			1\$:	MOV	(R1)+, R5		; GET DATA TO BE LOADE
2682	007466	110377	171676			MOVB	R3, 2DQREG		; SELECTED SECONDARY REGISTER
2683	007472	010577	171674			MOV	R5, 2DQSEC		; LOAD SECONDARY REGISTER
2684	007476	005203				INC	R3		; ADDRESS OF NEXT SECONDARY REGISTER
2685	007500	005300				DEC	R0		; CONTINUE IF NOT DONE
2686	007502	001370				BNE	1\$		
2687	007504	012700	000020			MOV	#16, R0		
2688	007510	012701	020016			MOV	#DATAB, R1		; GET POINTER TO ADDRESS TEST DATA
2689	007514	005003				CLR	R3		; FIRST SECONDARY REGISTER TO BE CHECKED
2690	007516	110377	171646	2\$:		MOVB	R3, 2DQREG		; SELECT SECONDARY REGISTER
2691	007522	017704	171644			MOV	2DQSEC, R4		; READ SECONDARY REGISTER
2692	007526	011105				MOV	(R1), R5		; GET TEST DATA
2693	007530	020504				CMP	R5, R4		; CHECK DATA
2694	007532	001401				BEQ	3\$		
2695	007534	104005				HLT	5		; SECONDARY REGISTER ADDRESSING ERROR
2696	007536	104401			3\$:	SCOPI			; CHECK FOR LOOP ON CURENT ADDRESS

005

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 55
DZDQAC.P11 16-DEC-76 11:24 SECONDARY REGISTER ADDRESSING TESTS

2697 007540 005203
2698 007542 062701 000002
2699 007546 005300
2700 007550 001362
2701 007552 104400
2702

45: INC R3
ADD #2,R1
DEC R0
BNE 25
SCOPE

;UPDATE ADDRESS
;UPDATE REGISTER DATA POINTER
;CONTINUE IF NOT DONE
;CHECK FOR ITERATIONS, LOOP

2703 007554

OPT2X:

2704
2705
2706
2707
2708
2709
2710
2711

; SECONDARY REGISTER READ/WRITE TEST
; SET BIT0 IN SYNC REGISTER
; VERIFY THAT BIT0 WAS SET
; CLEAR BIT0
; VERIFY THAT BIT0 WAS CLEARED

2712

; TEST 55

2713 007554 012737 000055 001226
2714 007562 012737 007644 001216
2715 007570 012703 000011
2716
2717 007574 110377 171570
2718 007600 012705 000001
2719 007604 010577 171562
2720
2721 007610 017704 171556
2722
2723 007614 020504
2724 007616 001401
2725 007620 104006
2726 007622 040577 171544
2727 007626 017704 171540
2728 007632 005005
2729
2730 007634 020504
2731 007636 001401
2732 007640 104006
2733 007642 104400
2734
2735
2736
2737
2738
2739
2740
2741
2742

TST55: MOV #55, TSTNO
MOV #TST56, NEXT
MOV #11, R3

; ADDRESS OF SECONDARY REGISTER
; SYNC
; SELECT SYNC REGISTER
; (R5)=BIT0
; SET BIT0 IN
; SYNC REGISTER
; (R4)=ACTUAL DATA IN
; SYNC REGISTER
; ARE EXPECTED AND RECEIVED DATA THE SAME ?
; BR IF GOOD
; SECONDARY REGISTER DATA ERROR
; CLEAR BIT0
; READ SYNC REGISTER
; EXPECT SYNC REGISTER
; TO CONTAIN 0
; ARE EXPECTED AND RECEIVED DATA THE SAME ?
; BR IF GOOD
; SECONDARY REGISTER DATA ERROR
; CHECK FOR ITERATIONS, LOOP

1\$:

BIC R5, @DQSEC
MOV @DQSEC, R4
CLR R5

2\$:

CMP R5, R4
BEQ 2\$
HLT 6

SCOPE

; SECONDARY REGISTER READ/WRITE TEST
; SET BIT1 IN SYNC REGISTER
; VERIFY THAT BIT1 WAS SET
; CLEAR BIT1
; VERIFY THAT BIT1 WAS CLEARED

2741

; TEST 56

2742
2743 007644 012737 000056 001226
2744 007652 012737 007734 001216
2745 007660 012703 000011
2746
2747 007664 110377 171500
2748 007670 012705 000002
2749 007674 010577 171472
2750
2751 007700 017704 171466
2752
2753 007704 020504
2754 007706 001401
2755 007710 104006
2756 007712 040577 171454
2757 007716 017704 171450
2758 007722 005005

TST56: MOV #56, TSTNO
MOV #TST57, NEXT
MOV #11, R3

; ADDRESS OF SECONDARY REGISTER
; SYNC
; SELECT SYNC REGISTER
; (R5)=BIT1
; SET BIT1 IN
; SYNC REGISTER
; (R4)=ACTUAL DATA IN
; SYNC REGISTER
; ARE EXPECTED AND RECEIVED DATA THE SAME ?
; BR IF GOOD
; SECONDARY REGISTER DATA ERROR
; CLEAR BIT1
; READ SYNC REGISTER
; EXPECT SYNC REGISTER

1\$:

BIC R5, @DQSEC
MOV @DQSEC, R4
CLR R5

```

2759
2760 007724 020504          CMP      R5,R4          ; TO CONTAIN 0
2761 007726 001401          BEQ      25             ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
2762 007730 104006          HLT      6             ; BR IF GOOD
2763 007732 104400          25:      SCOPE         ; SECONDARY REGISTER DATA ERROR
2764                                     ; CHECK FOR ITERATIONS, LOOP
2765                                     ; SECONDARY REGISTER READ/WRITE TEST
2766                                     ; SET BIT2 IN SYNC REGISTER
2767                                     ; VERIFY THAT BIT2 WAS SET
2768                                     ; CLEAR BIT2
2769                                     ; VERIFY THAT BIT2 WAS CLEARED
2770
2771 ; TEST 57
2772 ; *****
2773 007734 012737 000057 001226  TST57:  MOV      #57,TSTNO
2774 007742 012737 010024 001216  MOV      #TST60,NEXT
2775 007750 012703 000011          MOV      #11,R3          ; ADDRESS OF SECONDARY REGISTER
2776                                     ; SYNC
2777 007754 110377 171410          MOV      R3,ADQREG       ; SELECT SYNC REGISTER
2778 007760 012705 000004          MOV      #BIT2,R5        ; (R5)=BIT2
2779 007764 010577 171402          MOV      R5,ADQSEC       ; SET BIT2 IN
2780                                     ; SYNC REGISTER
2781 007770 017704 171376          MOV      ADQSEC,R4       ; (R4)=ACTUAL DATA IN
2782                                     ; SYNC REGISTER
2783 007774 020504          CMP      R5,R4          ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
2784 007776 001401          BEQ      15             ; BR IF GOOD
2785 010000 104006          HLT      6             ; SECONDARY REGISTER DATA ERROR
2786 010002 040577 171364          15:      BIC      R5,ADQSEC ; CLEAR BIT2
2787 010006 017704 171360          MOV      ADQSEC,R4       ; READ SYNC REGISTER
2788 010012 005005          CLR      R5             ; EXPECT SYNC REGISTER
2789                                     ; TO CONTAIN 0
2790 010014 020504          CMP      R5,R4          ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
2791 010016 001401          BEQ      25             ; BR IF GOOD
2792 010020 104006          HLT      6             ; SECONDARY REGISTER DATA ERROR
2793 010022 104400          25:      SCOPE         ; CHECK FOR ITERATIONS, LOOP
2794
2795                                     ; SECONDARY REGISTER READ/WRITE TEST
2796                                     ; SET BIT3 IN SYNC REGISTER
2797                                     ; VERIFY THAT BIT3 WAS SET
2798                                     ; CLEAR BIT3
2799                                     ; VERIFY THAT BIT3 WAS CLEARED
2800
2801 ; TEST 60
2802 ; *****
2803 010024 012737 000060 001226  TST60:  MOV      #60,TSTNO
2804 010032 012737 010114 001216  MOV      #TST61,NEXT
2805 010040 012703 000011          MOV      #11,R3          ; ADDRESS OF SECONDARY REGISTER
2806                                     ; SYNC
2807 010044 110377 171320          MOV      R3,ADQREG       ; SELECT SYNC REGISTER
2808 010050 012705 000010          MOV      #BIT3,R5        ; (R5)=BIT3
2809 010054 010577 171312          MOV      R5,ADQSEC       ; SET BIT3 IN
2810                                     ; SYNC REGISTER
2811 010060 017704 171306          MOV      ADQSEC,R4       ; (R4)=ACTUAL DATA IN
2812                                     ; SYNC REGISTER
2813 010064 020504          CMP      R5,R4          ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
2814 010066 001401          BEQ      15             ; BR IF GOOD

```

```

2815 010070 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
2816 010072 040577 171274 1S:  BIC     R5,20QSEC ;CLEAR BIT3
2817 010076 017704 171270    MOV     20QSEC,R4 ;READ SYNC REGISTER
2818 010102 005005          CLR     R5          ;EXPECT SYNC REGISTER
2819                                ;TO CONTAIN 0
2820 010104 020504          CMP     R5,R4       ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2821 010106 001401          BEQ     2S          ;BR IF GOOD
2822 010110 104006          HLT     6          ;SECONDARY REGISTER DATA ERROR
2823 010112 104400 2S:  SCOPE ;CHECK FOR ITERATIONS, LOOP
2824
2825                                ;SECONDARY REGISTER READ/WRITE TEST
2826                                ;SET BIT4 IN SYNC REGISTER
2827                                ;VERIFY THAT BIT4 WAS SET
2828                                ;CLEAR BIT4
2829                                ;VERIFY THAT BIT4 WAS CLEARED
2830
2831                                ; TEST 61
2832                                ;*****
2833 010114 012737 000061 001226 †ST61: MOV     #61,TSTNO
2834 010122 012737 010204 001216    MOV     #TST62,NEXT
2835 010130 012703 000011          MOV     #11,R3          ;ADDRESS OF SECONDARY REGISTER
2836                                ;SYNC
2837 010134 110377 171230    MOVB   R3,20QREG      ;SELECT SYNC REGISTER
2838 010140 012705 000020    MOV     #BIT4,R5      ;(R5)=BIT4
2839 010144 010577 171222    MOV     R5,20QSEC     ;SET BIT4 IN
2840                                ;SYNC REGISTER
2841 010150 017704 171216    MOV     20QSEC,R4     ;(R4)=ACTUAL DATA IN
2842                                ;SYNC REGISTER
2843 010154 020504          CMP     R5,R4       ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2844 010156 001401          BEQ     1S          ;BR IF GOOD
2845 010160 104006          HLT     6          ;SECONDARY REGISTER DATA ERROR
2846 010162 040577 171204 1S:  BIC     R5,20QSEC ;CLEAR BIT4
2847 010166 017704 171200    MOV     20QSEC,R4 ;READ SYNC REGISTER
2848 010172 005005          CLR     R5          ;EXPECT SYNC REGISTER
2849                                ;TO CONTAIN 0
2850 010174 020504          CMP     R5,R4       ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2851 010176 001401          BEQ     2S          ;BR IF GOOD
2852 010200 104006          HLT     6          ;SECONDARY REGISTER DATA ERROR
2853 010202 104400 2S:  SCOPE ;CHECK FOR ITERATIONS, LOOP
2854
2855                                ;SECONDARY REGISTER READ/WRITE TEST
2856                                ;SET BITS IN SYNC REGISTER
2857                                ;VERIFY THAT BITS WAS SET
2858                                ;CLEAR BITS
2859                                ;VERIFY THAT BITS WAS CLEARED
2860
2861                                ; TEST 62
2862                                ;*****
2863 010204 012737 000062 001226 †ST62: MOV     #62,TSTNO
2864 010212 012737 010274 001216    MOV     #TST63,NEXT
2865 010220 012703 000011          MOV     #11,R3          ;ADDRESS OF SECONDARY REGISTER
2866                                ;SYNC
2867 010224 110377 171140    MOVB   R3,20QREG      ;SELECT SYNC REGISTER
2868 010230 012705 000040    MOV     #BITS,R5     ;(R5)=BITS
2869 010234 010577 171132    MOV     R5,20QSEC     ;SET BITS IN
2870                                ;SYNC REGISTER
    
```

H05

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 59
DZDQAC.P11 16-DEC-76 11:24 SYNC REGISTER READ/WRITE TESTS.

```
2871 010240 017704 171126      MOV      2DQSEC,R4      ;(R4)=ACTUAL DATA IN
2872                                ;SYNC REGISTER
2873 010244 020504      CMP      R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2874 010246 001401      BEQ     1$          ;BR IF GOOD
2875 010250 104006      HLT     6          ;SECONDARY REGISTER DATA ERROR
2876 010252 040577 171114 1$:  BIC     R5,2DQSEC   ;CLEAR BITS
2877 010256 017704 171110      MOV     2DQSEC,R4   ;READ SYNC REGISTER
2878 010262 005005      CLR     R5         ;EXPECT SYNC REGISTER
2879                                ;TO CONTAIN 0
2880 010264 020504      CMP     R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2881 010266 001401      BEQ     2$          ;BR IF GOOD
2882 010270 104006      HLT     6          ;SECONDARY REGISTER DATA ERROR
2883 010272 104400 2$:  SCOPE ;CHECK FOR ITERATIONS, LOOP
2884
2885                                ;SECONDARY REGISTER READ/WRITE TEST
2886                                ;SET BIT6 IN SYNC REGISTER
2887                                ;VERIFY THAT BIT6 WAS SET
2888                                ;CLEAR BIT6
2889                                ;VERIFY THAT BIT6 WAS CLEARED
2890
2891                                ; TEST 63
2892                                ;*****
2893 010274 012737 000063 001226 1$T63: MOV     #63,TSTNO
2894 010302 012737 010364 001216      MOV     #TST64,NEXT
2895 010310 012703 000011      MOV     #11,R3      ;ADDRESS OF SECONDARY REGISTER
2896                                ;SYNC
2897 010314 110377 171050      MOVB   R3,2DQREG   ;SELECT SYNC REGISTER
2898 010320 012705 000100      MOV     #BIT6,R5   ;(R5)=BIT6
2899 010324 010577 171042      MOV     R5,2DQSEC  ;SET BIT6 IN
2900                                ;SYNC REGISTER
2901 010330 017704 171036      MOV     2DQSEC,R4  ;(R4)=ACTUAL DATA IN
2902                                ;SYNC REGISTER
2903 010334 020504      CMP     R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2904 010336 001401      BEQ     1$          ;BR IF GOOD
2905 010340 104006      HLT     6          ;SECONDARY REGISTER DATA ERROR
2906 010342 040577 171024 1$:  BIC     R5,2DQSEC   ;CLEAR BIT6
2907 010346 017704 171020      MOV     2DQSEC,R4  ;READ SYNC REGISTER
2908 010352 005005      CLR     R5         ;EXPECT SYNC REGISTER
2909                                ;TO CONTAIN 0
2910 010354 020504      CMP     R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2911 010356 001401      BEQ     2$          ;BR IF GOOD
2912 010360 104006      HLT     6          ;SECONDARY REGISTER DATA ERROR
2913 010362 104400 2$:  SCOPE ;CHECK FOR ITERATIONS, LOOP
2914
2915                                ;SECONDARY REGISTER READ/WRITE TEST
2916                                ;SET BIT7 IN SYNC REGISTER
2917                                ;VERIFY THAT BIT7 WAS SET
2918                                ;CLEAR BIT7
2919                                ;VERIFY THAT BIT7 WAS CLEARED
2920
2921                                ; TEST 64
2922                                ;*****
2923 010364 012737 000064 001226 1$T64: MOV     #64,TSTNO
2924 010372 012737 010454 001216      MOV     #TST65,NEXT
2925 010400 012703 000011      MOV     #11,R3      ;ADDRESS OF SECONDARY REGISTER
2926                                ;SYNC
```

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 60
DZDQAC.P11 16-DEC-76 11:24 SYNC REGISTER READ/WRITE TESTS.

```

2927 010404 110377 170760      MOV      R3,2DQREG      ;SELECT SYNC REGISTER
2928 010410 012705 000200      MOV      #BIT7,R5      ;(R5)=BIT7
2929 010414 010577 170752      MOV      R5,2DQSEC     ;SET BIT7 IN
2930                                ;SYNC REGISTER
2931 010420 017704 170746      MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
2932                                ;SYNC REGISTER
2933 010424 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2934 010426 001401                BEQ      1$           ;BR IF GOOD
2935 010430 104006                HLT      6           ;SECONDARY REGISTER DATA ERROR
2936 010432 040577 170734      1$:    BIC      R5,2DQSEC ;CLEAR BIT7
2937 010436 017704 170730      MOV      2DQSEC,R4     ;READ SYNC REGISTER
2938 010442 005005                CLR      R5           ;EXPECT SYNC REGISTER
2939                                ;TO CONTAIN 0
2940 010444 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2941 010446 001401                BEQ      2$           ;BR IF GOOD
2942 010450 104006                HLT      6           ;SECONDARY REGISTER DATA ERROR
2943 010452 104400                2$:    SCOPE         ;CHECK FOR ITERATIONS, LOOP
2944
2945                                ;SECONDARY REGISTER READ/WRITE TEST
2946                                ;SET BIT8 IN SYNC REGISTER
2947                                ;VERIFY THAT BIT8 WAS SET
2948                                ;CLEAR BIT8
2949                                ;VERIFY THAT BIT8 WAS CLEARED
2950
2951                                ; TEST 65
2952                                ;*****
2953 010454 012737 000065 001226  TST65: MOV      #65,TSTNO
2954 010462 012737 010544 001216  MOV      #TST66,NEXT
2955 010470 012703 000011                MOV      #11,R3      ;ADDRESS OF SECONDARY REGISTER
2956                                ;SYNC
2957 010474 110377 170670      MOV      R3,2DQREG     ;SELECT SYNC REGISTER
2958 010500 012705 000400      MOV      #BIT8,R5      ;(R5)=BIT8
2959 010504 010577 170662      MOV      R5,2DQSEC     ;SET BIT8 IN
2960                                ;SYNC REGISTER
2961 010510 017704 170656      MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
2962                                ;SYNC REGISTER
2963 010514 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2964 010516 001401                BEQ      1$           ;BR IF GOOD
2965 010520 104006                HLT      6           ;SECONDARY REGISTER DATA ERROR
2966 010522 040577 170644      1$:    BIC      R5,2DQSEC ;CLEAR BIT8
2967 010526 017704 170640      MOV      2DQSEC,R4     ;READ SYNC REGISTER
2968 010532 005005                CLR      R5           ;EXPECT SYNC REGISTER
2969                                ;TO CONTAIN 0
2970 010534 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2971 010536 001401                BEQ      2$           ;BR IF GOOD
2972 010540 104006                HLT      6           ;SECONDARY REGISTER DATA ERROR
2973 010542 104400                2$:    SCOPE         ;CHECK FOR ITERATIONS, LOOP
2974
2975                                ;SECONDARY REGISTER READ/WRITE TEST
2976                                ;SET BIT9 IN SYNC REGISTER
2977                                ;VERIFY THAT BIT9 WAS SET
2978                                ;CLEAR BIT9
2979                                ;VERIFY THAT BIT9 WAS CLEARED
2980
2981                                ; TEST 66
2982                                ;*****

```

DZDGA MACY11 27(1006) 22-DEC-76 10:57 PAGE 61
 DZDQAC.P11 16-DEC-76 11:24 SYNC REGISTER READ/WRITE TESTS.

```

2983 010544 012737 000066 001226 TST66: MOV #66,TSTNO
2984 010552 012737 010634 001216 MOV #TST67,NEXT
2985 010560 012703 000011 MOV #11,R3
2986 MOVB R3,SDQREG ;ADDRESS OF SECONDARY REGISTER
2987 010564 110377 170600 MOV #BIT9,R5 ;SYNC
2988 010570 012705 001000 MOV R5,SDQSEC ;SELECT SYNC REGISTER
2989 010574 010577 170572 MOV SDQSEC,R4 ;(R5)=BIT9
2990 ;SET BIT9 IN
2991 010600 017704 170566 MOV SDQSEC,R4 ;SYNC REGISTER
2992 ;(R4)=ACTUAL DATA IN
2993 010604 020504 CMP R5,R4 ;SYNC REGISTER
2994 010606 001401 BEQ 1$ ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
2995 010610 104006 HLT 6 ;BR IF GOOD
2996 010612 040577 170554 1$: BIC R5,SDQSEC ;SECONDARY REGISTER DATA ERROR
2997 010616 017704 170550 MOV SDQSEC,R4 ;CLEAR BIT9
2998 010622 005005 CLR R5 ;READ SYNC REGISTER
2999 ;EXPECT SYNC REGISTER
3000 010624 020504 CMP R5,R4 ;TO CONTAIN 0
3001 010626 001401 BEQ 2$ ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3002 010630 104006 HLT 6 ;BR IF GOOD
3003 010632 104400 2$: SCOPE ;SECONDARY REGISTER DATA ERROR
3004 ;CHECK FOR ITERATIONS, LOOP
3005 ;SECONDARY REGISTER READ/WRITE TEST
3006 ;SET BIT10 IN SYNC REGISTER
3007 ;VERIFY THAT BIT10 WAS SET
3008 ;CLEAR BIT10
3009 ;VERIFY THAT BIT10 WAS CLEARED
3010
3011 ; TEST 67
3012 ;*****
3013 010634 012737 000067 001226 TST67: MOV #67,TSTNO
3014 010642 012737 010724 001216 MOV #TST70,NEXT
3015 010650 012703 000011 MOV #11,R3
3016 MOVB R3,SDQREG ;ADDRESS OF SECONDARY REGISTER
3017 010654 110377 170510 MOV #BIT10,R5 ;SYNC
3018 010660 012705 002000 MOV R5,SDQSEC ;SELECT SYNC REGISTER
3019 010664 010577 170502 MOV SDQSEC,R4 ;(R5)=BIT10
3020 ;SET BIT10 IN
3021 010670 017704 170476 MOV SDQSEC,R4 ;SYNC REGISTER
3022 ;(R4)=ACTUAL DATA IN
3023 010674 020504 CMP R5,R4 ;SYNC REGISTER
3024 010676 001401 BEQ 1$ ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3025 010700 104006 HLT 6 ;BR IF GOOD
3026 010702 040577 170464 1$: BIC R5,SDQSEC ;SECONDARY REGISTER DATA ERROR
3027 010706 017704 170460 MOV SDQSEC,R4 ;CLEAR BIT10
3028 010712 005005 CLR R5 ;READ SYNC REGISTER
3029 ;EXPECT SYNC REGISTER
3030 010714 020504 CMP R5,R4 ;TO CONTAIN 0
3031 010716 001401 BEQ 2$ ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3032 010720 104006 HLT 6 ;BR IF GOOD
3033 010722 104400 2$: SCOPE ;SECONDARY REGISTER DATA ERROR
3034 ;CHECK FOR ITERATIONS, LOOP
3035 ;SECONDARY REGISTER READ/WRITE TEST
3036 ;SET BIT11 IN SYNC REGISTER
3037 ;VERIFY THAT BIT11 WAS SET
3038 ;CLEAR BIT11
    
```



```

3095                                     ;SECONDARY REGISTER READ/WRITE TEST
3096                                     ;SET BIT13 IN SYNC REGISTER
3097                                     ;VERIFY THAT BIT13 WAS SET
3098                                     ;CLEAR BIT13
3099                                     ;VERIFY THAT BIT13 WAS CLEARED
3100
3101                                     ; TEST 72
3102                                     ;*****
3103 011104 012737 000072 001226 †ST72: MOV #72,TSTNO
3104 011112 012737 011174 001216      MOV #TST73,NEXT
3105 011120 012703 000011              MOV #11,R3
3106                                     ;ADDRESS OF SECONDARY REGISTER
3107 011124 110377 170240              MOVB R3,ADQREG ;SELECT SYNC REGISTER
3108 011130 012705 020000              MOV #BIT13,R5 ;(R5)=BIT13
3109 011134 010577 170232              MOV R5,ADQSEC ;SET BIT13 IN
3110                                     ;SYNC REGISTER
3111 011140 017704 170226              MOV ADQSEC,R4 ;(R4)=ACTUAL DATA IN
3112                                     ;SYNC REGISTER
3113 011144 020504                    CMP R5,R4 ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3114 011146 001401                    BEQ 1$ ;BR IF GOOD
3115 011150 104006                    HLT 6 ;SECONDARY REGISTER DATA ERROR
3116 011152 040577 170214 1$: BIC R5,ADQSEC ;CLEAR BIT13
3117 011156 017704 170210              MOV ADQSEC,R4 ;READ SYNC REGISTER
3118 011162 005005                    CLR R5 ;EXPECT SYNC REGISTER
3119                                     ;TO CONTAIN 0
3120 011164 020504                    CMP R5,R4 ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3121 011166 001401                    BEQ 2$ ;BR IF GOOD
3122 011170 104006                    HLT 6 ;SECONDARY REGISTER DATA ERROR
3123 011172 104400 2$: SCOPE ;CHECK FOR ITERATIONS, LOOP
3124
3125                                     ;SECONDARY REGISTER READ/WRITE TEST
3126                                     ;SET BIT14 IN SYNC REGISTER
3127                                     ;VERIFY THAT BIT14 WAS SET
3128                                     ;CLEAR BIT14
3129                                     ;VERIFY THAT BIT14 WAS CLEARED
3130
3131                                     ; TEST 73
3132                                     ;*****
3133 011174 012737 000073 001226 †ST73: MOV #73,TSTNO
3134 011202 012737 011264 001216      MOV #TST74,NEXT
3135 011210 012703 000011              MOV #11,R3
3136                                     ;ADDRESS OF SECONDARY REGISTER
3137 011214 110377 170150              MOVB R3,ADQREG ;SELECT SYNC REGISTER
3138 011220 012705 040000              MOV #BIT14,R5 ;(R5)=BIT14
3139 011224 010577 170142              MOV R5,ADQSEC ;SET BIT14 IN
3140                                     ;SYNC REGISTER
3141 011230 017704 170136              MOV ADQSEC,R4 ;(R4)=ACTUAL DATA IN
3142                                     ;SYNC REGISTER
3143 011234 020504                    CMP R5,R4 ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3144 011236 001401                    BEQ 1$ ;BR IF GOOD
3145 011240 104006                    HLT 6 ;SECONDARY REGISTER DATA ERROR
3146 011242 040577 170124 1$: BIC R5,ADQSEC ;CLEAR BIT14
3147 011246 017704 170120              MOV ADQSEC,R4 ;READ SYNC REGISTER
3148 011252 005005                    CLR R5 ;EXPECT SYNC REGISTER
3149                                     ;TO CONTAIN 0
3150 011254 020504                    CMP R5,R4 ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
    
```


M05

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 64
DZDQAC.P11 16-DEC-76 11:24

SYNC REGISTER READ/WRITE TESTS.

```

3151 011256 001401          BEQ      25          ;BR IF GOOD
3152 011260 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3153 011262 104400          25:     SCOPE      ;CHECK FOR ITERATIONS, LOOP
3154
3155          ;SECONDARY REGISTER READ/WRITE TEST
3156          ;SET BIT15 IN SYNC REGISTER
3157          ;VERIFY THAT BIT15 WAS SET
3158          ;CLEAR BIT15
3159          ;VERIFY THAT BIT15 WAS CLEARED
3160
3161          ; TEST 74
3162          ;*****
3163 011264 012737 000074 001226 1ST74: MOV      #74,TSTNO
3164 011272 012737 011354 001216      MOV      #TST75,NEXT
3165 011300 012703 000011          MOV      #11,R3          ;ADDRESS OF SECONDARY REGISTER
3166          ;SYNC
3167 011304 110377 170060          MOVB     R3,2DQREG      ;SELECT SYNC REGISTER
3168 011310 012705 100000          MOV      #BIT15,R5     ;(R5)=BIT15
3169 011314 010577 170052          MOV      R5,2DQSEC     ;SET BIT15 IN
3170          ;SYNC REGISTER
3171 011320 017704 170046          MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
3172          ;SYNC REGISTER
3173 011324 020504          CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3174 011326 001401          BEQ      15          ;BR IF GOOD
3175 011330 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3176 011332 040577 170034          15:     BIC      R5,2DQSEC ;CLEAR BIT15
3177 011336 017704 170030          MOV      2DQSEC,R4     ;READ SYNC REGISTER
3178 011342 005005          CLR      R5          ;EXPECT SYNC REGISTER
3179          ;TO CONTAIN 0
3180 011344 020504          CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3181 011346 001401          BEQ      25          ;BR IF GOOD
3182 011350 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3183 011352 104400          25:     SCOPE      ;CHECK FOR ITERATIONS, LOOP

```

N05

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 65
 DZDQAC.P11 16-DEC-76 11:24 MISCELLANEOUS REGISTER READ/WRITE TESTS.

```

3184
3185
3186
3187
3188
3189
3190
3191
3192
3193 011354 012737 000075 001226
3194 011362 012737 011444 001216
3195 011370 012703 000012
3196
3197 011374 110377 167770
3198 011400 012705 000001
3199 011404 010577 167762
3200
3201 011410 017704 167756
3202
3203 011414 020504
3204 011416 001401
3205 011420 104006
3206 011422 040577 167744
3207 011426 017704 167740
3208 011432 005005
3209
3210 011434 020504
3211 011436 001401
3212 011440 104006
3213 011442 104400
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223 011444 012737 000076 001226
3224 011452 012737 011534 001216
3225 011460 012703 000012
3226
3227 011464 110377 167700
3228 011470 012705 000002
3229 011474 010577 167672
3230
3231 011500 017704 167666
3232
3233 011504 020504
3234 011506 001401
3235 011510 104006
3236 011512 040577 167654
3237 011516 017704 167650
3238 011522 005005
3239

```

```

: SECONDARY REGISTER READ/WRITE TEST
: SET BIT0 IN MISCELLANEOUS REGISTER
: VERIFY THAT BIT0 WAS SET
: CLEAR BIT0
: VERIFY THAT BIT0 WAS CLEARED

: TEST 75
: *****
TST75: MOV #75, TSTNO
      MOV #TST76, NEXT
      MOV #12, R3
      MOVB R3, @DQREG
      MOV #BIT0, R5
      MOV R5, @DQSEC
      MOV @DQSEC, R4
      CMP R5, R4
      BEQ 1$
      HLT 6
1$: BIC R5, @DQSEC
   MOV @DQSEC, R4
   CLR R5
      CMP R5, R4
      BEQ 2$
      HLT 6
2$: SCOPE

```

```

: ADDRESS OF SECONDARY REGISTER
: MISCELLANEOUS
: SELECT MISCELLANEOUS REGISTER
: (R5)=BIT0
: SET BIT0 IN
: MISCELLANEOUS REGISTER
: (R4)=ACTUAL DATA IN
: MISCELLANEOUS REGISTER
: ARE EXPECTED AND RECEIVED DATA THE SAME ?
: BR IF GOOD
: SECONDARY REGISTER DATA ERROR
: CLEAR BIT0
: READ MISCELLANEOUS REGISTER
: EXPECT MISCELLANEOUS REGISTER
: TO CONTAIN 0
: ARE EXPECTED AND RECEIVED DATA THE SAME ?
: BR IF GOOD
: SECONDARY REGISTER DATA ERROR
: CHECK FOR ITERATIONS, LOOP

```

```

: SECONDARY REGISTER READ/WRITE TEST
: SET BIT1 IN MISCELLANEOUS REGISTER
: VERIFY THAT BIT1 WAS SET
: CLEAR BIT1
: VERIFY THAT BIT1 WAS CLEARED

: TEST 76
: *****
TST76: MOV #76, TSTNO
      MOV #TST77, NEXT
      MOV #12, R3
      MOVB R3, @DQREG
      MOV #BIT1, R5
      MOV R5, @DQSEC
      MOV @DQSEC, R4
      CMP R5, R4
      BEQ 1$
      HLT 6
1$: BIC R5, @DQSEC
   MOV @DQSEC, R4
   CLR R5

```

```

: ADDRESS OF SECONDARY REGISTER
: MISCELLANEOUS
: SELECT MISCELLANEOUS REGISTER
: (R5)=BIT1
: SET BIT1 IN
: MISCELLANEOUS REGISTER
: (R4)=ACTUAL DATA IN
: MISCELLANEOUS REGISTER
: ARE EXPECTED AND RECEIVED DATA THE SAME ?
: BR IF GOOD
: SECONDARY REGISTER DATA ERROR
: CLEAR BIT1
: READ MISCELLANEOUS REGISTER
: EXPECT MISCELLANEOUS REGISTER
: TO CONTAIN 0

```

MISCELLANEOUS REGISTER READ/WRITE TESTS.

```

3240 011524 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3241 011526 001401          BEQ      2$            ;BR IF GOOD
3242 011530 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3243 011532 104400          2$:      SCOPE        ;CHECK FOR ITERATIONS, LOOP
3244
3245          ;SECONDARY REGISTER READ/WRITE TEST
3246          ;SET BIT3 IN MISCELLANEOUS REGISTER
3247          ;VERIFY THAT BIT3 WAS SET
3248          ;CLEAR BIT3
3249          ;VERIFY THAT BIT3 WAS CLEARED
3250
3251          ; TEST 77
3252          ;*****
3253 011534 012737 000077 001226  †TST77: MOV      #77,TSTNO
3254 011542 012737 011624 001216  MOV      #TST100,NEXT
3255 011550 012703 000012          MOV      #12,R3          ;ADDRESS OF SECONDARY REGISTER
3256          ;MISCELLANEOUS
3257 011554 110377 167610          MOVB     R3,200REG      ;SELECT MISCELLANEOUS REGISTER
3258 011560 012705 000010          MOV      #BIT3,R5      ;(R5)=BIT3
3259 011564 010577 167602          MOV      R5,200SEC     ;SET BIT3 IN
3260          ;MISCELLANEOUS REGISTER
3261 011570 017704 167576          MOV      200SEC,R4     ;(R4)=ACTUAL DATA IN
3262          ;MISCELLANEOUS REGISTER
3263 011574 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3264 011576 001401          BEQ      1$            ;BR IF GOOD
3265 011600 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3266 011602 040577 167564          1$:      BIC      R5,200SEC ;CLEAR BIT3
3267 011606 017704 167560          MOV      200SEC,R4     ;READ MISCELLANEOUS REGISTER
3268 011612 005005          CLR      R5            ;EXPECT MISCELLANEOUS REGISTER
3269          ;TO CONTAIN 0
3270 011614 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3271 011616 001401          BEQ      2$            ;BR IF GOOD
3272 011620 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3273 011622 104400          2$:      SCOPE        ;CHECK FOR ITERATIONS, LOOP
3274
3275          ;SECONDARY REGISTER READ/WRITE TEST
3276          ;SET BIT6 IN MISCELLANEOUS REGISTER
3277          ;VERIFY THAT BIT6 WAS SET
3278          ;CLEAR BIT6
3279          ;VERIFY THAT BIT6 WAS CLEARED
3280
3281          ; TEST 100
3282          ;*****
3283 011624 012737 000100 001226  †TST100: MOV     #100,TSTNO
3284 011632 012737 011714 001216  MOV     #TST101,NEXT
3285 011640 012703 000012          MOV     #12,R3          ;ADDRESS OF SECONDARY REGISTER
3286          ;MISCELLANEOUS
3287 011644 110377 167520          MOVB    R3,200REG      ;SELECT MISCELLANEOUS REGISTER
3288 011650 012705 000100          MOV     #BIT6,R5      ;(R5)=BIT6
3289 011654 010577 167512          MOV     R5,200SEC     ;SET BIT6 IN
3290          ;MISCELLANEOUS REGISTER
3291 011660 017704 167506          MOV     200SEC,R4     ;(R4)=ACTUAL DATA IN
3292          ;MISCELLANEOUS REGISTER
3293 011664 020504          CMP     R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3294 011666 001401          BEQ     1$            ;BR IF GOOD
3295 011670 104006          HLT     6             ;SECONDARY REGISTER DATA ERROR

```

MISCELLANEOUS REGISTER READ/WRITE TESTS.

```

3296 011672 040577 167474      1S:  BIC      R5,2DQSEC      ;CLEAR BIT6
3297 011676 017704 167470      MOV      2DQSEC,R4      ;READ MISCELLANEOUS REGISTER
3298 011702 005005      CLR      R5              ;EXPECT MISCELLANEOUS REGISTER
3299                                     ;TO CONTAIN 0
3300 011704 020504      CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3301 011706 001401      BEQ      2S              ;BR IF GOOD
3302 011710 104006      HLT      6              ;SECONDARY REGISTER DATA ERROR
3303 011712 104400      2S:  SCOPE              ;CHECK FOR ITERATIONS, LOOP
3304
3305                                     ;SECONDARY REGISTER READ/WRITE TEST
3306                                     ;SET BIT7 IN MISCELLANEOUS REGISTER
3307                                     ;VERIFY THAT BIT7 WAS SET
3308                                     ;CLEAR BIT7
3309                                     ;VERIFY THAT BIT7 WAS CLEARED
3310
3311                                     ; TEST 101
3312                                     ;*****
3313 011714 012737 000101 001226  †TST101: MOV      #101,TSTNO
3314 011722 012737 012004 001216  MOV      #TST102,NEXT
3315 011730 012703 000012      MOV      #12,R3          ;ADDRESS OF SECONDARY REGISTER
3316                                     ;MISCELLANEOUS
3317 011734 110377 167430      MOV      R3,2DQREG      ;SELECT MISCELLANEOUS REGISTER
3318 011740 012705 000200      MOV      #BIT7,R5      ;(R5)=BIT7
3319 011744 010577 167422      MOV      R5,2DQSEC      ;SET BIT7 IN
3320                                     ;MISCELLANEOUS REGISTER
3321 011750 017704 167416      MOV      2DQSEC,R4      ;(R4)=ACTUAL DATA IN
3322                                     ;MISCELLANEOUS REGISTER
3323 011754 020504      CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3324 011756 001401      BEQ      1S              ;BR IF GOOD
3325 011760 104006      HLT      6              ;SECONDARY REGISTER DATA ERROR
3326 011762 040577 167404      1S:  BIC      R5,2DQSEC      ;CLEAR BIT7
3327 011766 017704 167400      MOV      2DQSEC,R4      ;READ MISCELLANEOUS REGISTER
3328 011772 005005      CLR      R5              ;EXPECT MISCELLANEOUS REGISTER
3329                                     ;TO CONTAIN 0
3330 011774 020504      CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3331 011776 001401      BEQ      2S              ;BR IF GOOD
3332 012000 104006      HLT      6              ;SECONDARY REGISTER DATA ERROR
3333 012002 104400      2S:  SCOPE              ;CHECK FOR ITERATIONS, LOOP
3334
3335                                     ;SECONDARY REGISTER READ/WRITE TEST
3336                                     ;SET BIT8 IN MISCELLANEOUS REGISTER
3337                                     ;VERIFY THAT BIT8 WAS SET
3338                                     ;CLEAR BIT8
3339                                     ;VERIFY THAT BIT8 WAS CLEARED
3340
3341                                     ; TEST 102
3342                                     ;*****
3343 012004 012737 000102 001226  †TST102: MOV      #102,TSTNO
3344 012012 012737 012074 001216  MOV      #TST103,NEXT
3345 012020 012703 000012      MOV      #12,R3          ;ADDRESS OF SECONDARY REGISTER
3346                                     ;MISCELLANEOUS
3347 012024 110377 167340      MOV      R3,2DQREG      ;SELECT MISCELLANEOUS REGISTER
3348 012030 012705 000400      MOV      #BIT8,R5      ;(R5)=BIT8
3349 012034 010577 167332      MOV      R5,2DQSEC      ;SET BIT8 IN
3350                                     ;MISCELLANEOUS REGISTER
3351 012040 017704 167326      MOV      2DQSEC,R4      ;(R4)=ACTUAL DATA IN

```

```

3352                                     ; MISCELLANEOUS REGISTER
3353 012044 020504                       CMP    R5,R4      ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
3354 012046 001401                       BEQ    1$        ; BR IF GOOD
3355 012050 104006                       HLT    6         ; SECONDARY REGISTER DATA ERROR
3356 012052 040577 167314                1$: BIC    R5,20QSEC ; CLEAR BIT8
3357 012056 017704 167310                MOV    20QSEC,R4 ; READ MISCELLANEOUS REGISTER
3358 012062 005005                       CLR    R5        ; EXPECT MISCELLANEOUS REGISTER
3359                                     ; TO CONTAIN 0
3360 012064 020504                       CMP    R5,R4      ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
3361 012066 001401                       BEQ    2$        ; BR IF GOOD
3362 012070 104006                       HLT    6         ; SECONDARY REGISTER DATA ERROR
3363 012072 104400                2$: SCOPE        ; CHECK FOR ITERATIONS, LOOP
3364
3365                                     ; SECONDARY REGISTER READ/WRITE TEST
3366                                     ; SET BIT9 IN MISCELLANEOUS REGISTER
3367                                     ; VERIFY THAT BIT9 WAS SET
3368                                     ; CLEAR BIT9
3369                                     ; VERIFY THAT BIT9 WAS CLEARED
3370
3371                                     ; TEST 103
3372                                     ; *****
3373 012074 012737 000103 001226 1$TST103: MOV    #103,TSTNO
3374 012102 012737 012164 001216        MOV    #TST104,NEXT
3375 012110 012703 000012                MOV    #12,R3      ; ADDRESS OF SECONDARY REGISTER
3376                                     ; MISCELLANEOUS
3377 012114 110377 167250                MOV    R3,20QREG   ; SELECT MISCELLANEOUS REGISTER
3378 012120 012705 001000                MOV    #BIT9,R5    ; (R5)=BIT9
3379 012124 010577 167242                MOV    R5,20QSEC   ; SET BIT9 IN
3380                                     ; MISCELLANEOUS REGISTER
3381 012130 017704 167236                MOV    20QSEC,R4   ; (R4)=ACTUAL DATA IN
3382                                     ; MISCELLANEOUS REGISTER
3383 012134 020504                       CMP    R5,R4      ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
3384 012136 001401                       BEQ    1$        ; BR IF GOOD
3385 012140 104006                       HLT    6         ; SECONDARY REGISTER DATA ERROR
3386 012142 040577 167224                1$: BIC    R5,20QSEC ; CLEAR BIT9
3387 012146 017704 167220                MOV    20QSEC,R4   ; READ MISCELLANEOUS REGISTER
3388 012152 005005                       CLR    R5        ; EXPECT MISCELLANEOUS REGISTER
3389                                     ; TO CONTAIN 0
3390 012154 020504                       CMP    R5,R4      ; ARE EXPECTED AND RECEIVED DATA THE SAME ?
3391 012156 001401                       BEQ    2$        ; BR IF GOOD
3392 012160 104006                       HLT    6         ; SECONDARY REGISTER DATA ERROR
3393 012162 104400                2$: SCOPE        ; CHECK FOR ITERATIONS, LOOP
3394
3395                                     ; SECONDARY REGISTER READ/WRITE TEST
3396                                     ; SET BIT10 IN MISCELLANEOUS REGISTER
3397                                     ; VERIFY THAT BIT10 WAS SET
3398                                     ; CLEAR BIT10
3399                                     ; VERIFY THAT BIT10 WAS CLEARED
3400
3401                                     ; TEST 104
3402                                     ; *****
3403 012164 012737 000104 001226 1$TST104: MOV    #104,TSTNO
3404 012172 012737 012254 001216        MOV    #TST105,NEXT
3405 012200 012703 000012                MOV    #12,R3      ; ADDRESS OF SECONDARY REGISTER
3406                                     ; MISCELLANEOUS
3407 012204 110377 167160                MOV    R3,20QREG   ; SELECT MISCELLANEOUS REGISTER

```

```

3408 012210 012705 002000      MOV      #BIT10,R5      ;(R5)=BIT10
3409 012214 010577 167152      MOV      R5,20QSEC     ;SET BIT10 IN
3410                                ;MISCELLANEOUS REGISTER
3411 012220 017704 167146      MOV      20QSEC,R4     ;(R4)=ACTUAL DATA IN
3412                                ;MISCELLANEOUS REGISTER
3413 012224 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3414 012226 001401                BEQ      1$           ;BR IF GOOD
3415 012230 104006                HLT      6            ;SECONDARY REGISTER DATA ERROR
3416 012232 040577 167134      1$:    BIC      R5,20QSEC ;CLEAR BIT10
3417 012236 017704 167130      MOV      20QSEC,R4     ;READ MISCELLANEOUS REGISTER
3418 012242 005005                CLR      R5           ;EXPECT MISCELLANEOUS REGISTER
3419                                ;TO CONTAIN 0
3420 012244 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3421 012246 001401                BEQ      2$           ;BR IF GOOD
3422 012250 104006                HLT      6            ;SECONDARY REGISTER DATA ERROR
3423 012252 104400      2$:    SCOPE          ;CHECK FOR ITERATIONS, LOOP
3424
3425                                ;SECONDARY REGISTER READ/WRITE TEST
3426                                ;SET BIT11 IN MISCELLANEOUS REGISTER
3427                                ;VERIFY THAT BIT11 WAS SET
3428                                ;CLEAR BIT11
3429                                ;VERIFY THAT BIT11 WAS CLEARED
3430
3431                                ; TEST 105
3432                                ;*****
3433 012254 012737 000105 001226  TST105: MOV      #105,TSTNO
3434 012262 012737 012344 001216  MOV      #TST106,NEXT
3435 012270 012703 000012                MOV      #12,R3       ;ADDRESS OF SECONDARY REGISTER
3436                                ;MISCELLANEOUS
3437 012274 110377 167070      MOV      R3,20QREG     ;SELECT MISCELLANEOUS REGISTER
3438 012300 012705 004000      MOV      #BIT11,R5     ;(R5)=BIT11
3439 012304 010577 167062      MOV      R5,20QSEC     ;SET BIT11 IN
3440                                ;MISCELLANEOUS REGISTER
3441 012310 017704 167056      MOV      20QSEC,R4     ;(R4)=ACTUAL DATA IN
3442                                ;MISCELLANEOUS REGISTER
3443 012314 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3444 012316 001401                BEQ      1$           ;BR IF GOOD
3445 012320 104006                HLT      6            ;SECONDARY REGISTER DATA ERROR
3446 012322 040577 167044      1$:    BIC      R5,20QSEC ;CLEAR BIT11
3447 012326 017704 167040      MOV      20QSEC,R4     ;READ MISCELLANEOUS REGISTER
3448 012332 005005                CLR      R5           ;EXPECT MISCELLANEOUS REGISTER
3449                                ;TO CONTAIN 0
3450 012334 020504                CMP      R5,R4         ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3451 012336 001401                BEQ      2$           ;BR IF GOOD
3452 012340 104006                HLT      6            ;SECONDARY REGISTER DATA ERROR
3453 012342 104400      2$:    SCOPE          ;CHECK FOR ITERATIONS, LOOP
3454
3455                                ;SECONDARY REGISTER READ/WRITE TEST
3456                                ;SET BIT15 IN MISCELLANEOUS REGISTER
3457                                ;VERIFY THAT BIT15 WAS SET
3458                                ;CLEAR BIT15
3459                                ;VERIFY THAT BIT15 WAS CLEARED
3460
3461                                ; TEST 106
3462                                ;*****
3463 012344 012737 000106 001226  TST106: MOV      #106,TSTNO

```

F06

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 70
 DZDQAC.P11 16-DEC-76 11:24 MISCELLANEOUS REGISTER READ/WRITE TESTS.

3464	012352	012737	012434	001216	MOV	#CHKAB1,NEXT	
3465	012360	012703	000012		MOV	#12,R3	: ADDRESS OF SECONDARY REGISTER
3466							: MISCELLANEOUS
3467	012364	110377	167000		MOVB	R3,2DQREG	: SELECT MISCELLANEOUS REGISTER
3468	012370	012705	100000		MOV	#BIT15,R5	: (R5)=BIT15
3469	012374	010577	166772		MOV	R5,2DQSEC	: SET BIT15 IN
3470							: MISCELLANEOUS REGISTER
3471	012400	017704	166766		MOV	2DQSEC,R4	: (R4)=ACTUAL DATA IN
3472							: MISCELLANEOUS REGISTER
3473	012404	020504			CMP	R5,R4	: ARE EXPECTED AND RECEIVED DATA THE SAME ?
3474	012406	001401			BEQ	1\$: BR IF GOOD
3475	012410	104006			HLT	6	: SECONDARY REGISTER DATA ERROR
3476	012412	040577	166754	1\$:	BIC	R5,2DQSEC	: CLEAR BIT15
3477	012416	017704	166750		MOV	2DQSEC,R4	: READ MISCELLANEOUS REGISTER
3478	012422	005005			CLR	R5	: EXPECT MISCELLANEOUS REGISTER
3479							: TO CONTAIN 0
3480	012424	020504			CMP	R5,R4	: ARE EXPECTED AND RECEIVED DATA THE SAME ?
3481	012426	001401			BEQ	2\$: BR IF GOOD
3482	012430	104006			HLT	6	: SECONDARY REGISTER DATA ERROR
3483	012432	104400		2\$:	SCOPE		: CHECK FOR ITERATIONS, LOOP

```

3484                                     ; IF BCC OPTION IS INSTALLED
3485                                     ; TESTS 107 THRU 126 WILL BE EXECUTED
3486
3487 012434 032737 002000 001510 CHKAB1: BIT      @RBBIT,DQSTAT
3488 012442 001002                BNE      TST107
3489 012444 000137 014250                JMP      .EOP
3490
3491                                     ; SECONDARY REGISTER READ/WRITE TEST
3492                                     ; SET BIT0 IN BCC POLYNOMIAL REGISTER
3493                                     ; VERIFY THAT BIT0 WAS SET
3494                                     ; CLEAR BIT0
3495                                     ; VERIFY THAT BIT0 WAS CLEARED
3496
3497                                     ; TEST 107
3498                                     ; *****
3499 012450 012737 000107 001226 †TST107: MOV      @107,TSTNO
3500 012456 012737 012540 001216                MOV      @TST110,NEXT
3501 012464 012703 000017                MOV      @17,R3
3502                                     ; ADDRESS OF SECONDARY REGISTER
3503                                     ; BCC POLYNOMIAL
3504 012470 110377 166674                MOV      R3,@DQREG
3505 012474 012705 000001                MOV      @BIT0,R5
3506 012500 010577 166666                MOV      R5,@DQSEC
3507 012504 017704 166662                MOV      @DQSEC,R4
3508                                     ; (R4)=ACTUAL DATA IN
3509                                     ; BCC POLYNOMIAL REGISTER
3510 012510 020504                CMP      R5,R4
3511 012512 001401                BEQ      1$
3512 012514 104006                HLT      6
3513 012516 040577 166650 1$: BIC      R5,@DQSEC
3514 012522 017704 166644                MOV      @DQSEC,R4
3515 012526 005005                CLR      R5
3516                                     ; (R4)=ACTUAL DATA IN
3517 012530 020504                CMP      R5,R4
3518 012532 001401                BEQ      2$
3519 012534 104006                HLT      6
3520 012536 104400 2$: SCOPE
3521                                     ; SECONDARY REGISTER READ/WRITE TEST
3522                                     ; SET BIT1 IN BCC POLYNOMIAL REGISTER
3523                                     ; VERIFY THAT BIT1 WAS SET
3524                                     ; CLEAR BIT1
3525                                     ; VERIFY THAT BIT1 WAS CLEARED
3526
3527                                     ; TEST 110
3528                                     ; *****
3529 012540 012737 000110 001226 †TST110: MOV      @110,TSTNO
3530 012546 012737 012630 001216                MOV      @TST111,NEXT
3531 012554 012703 000017                MOV      @17,R3
3532                                     ; ADDRESS OF SECONDARY REGISTER
3533                                     ; BCC POLYNOMIAL
3534 012560 110377 166604                MOV      R3,@DQREG
3535 012564 012705 000002                MOV      @BIT1,R5
3536 012570 010577 166576                MOV      R5,@DQSEC
3537 012574 017704 166572                MOV      @DQSEC,R4
3538                                     ; (R4)=ACTUAL DATA IN
3539 012600 020504                CMP      R5,R4
3539                                     ; BCC POLYNOMIAL REGISTER
3539                                     ; ARE EXPECTED AND RECEIVED DATA THE SAME ?

```



```

3540 012602 001401          BEQ      15          ;BR IF GOOD
3541 012604 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3542 012606 040577 166560 15:  BIC      R5,2DQSEC ;CLEAR BIT1
3543 012612 017704 166554  MOV      2DQSEC,R4 ;READ BCC POLYNOMIAL REGISTER
3544 012616 005005          CLR      R5          ;EXPECT BCC POLYNOMIAL REGISTER
3545                                ;TO CONTAIN 0
3546 012620 020504          CMP      R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3547 012622 001401          BEQ      25          ;BR IF GOOD
3548 012624 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3549 012626 104400 25:  SCOPE      ;CHECK FOR ITERATIONS, LOOP
3550
3551                                ;SECONDARY REGISTER READ/WRITE TEST
3552                                ;SET BIT2 IN BCC POLYNOMIAL REGISTER
3553                                ;VERIFY THAT BIT2 WAS SET
3554                                ;CLEAR BIT2
3555                                ;VERIFY THAT BIT2 WAS CLEARED
3556
3557                                ; TEST 111
3558                                ;*****
3559 012630 012737 000111 001226 †TST111: MOV      #111,TSTNO
3560 012636 012737 012720 001216  MOV      #TST112,NEXT
3561 012644 012703 000017          MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
3562                                ;BCC POLYNOMIAL
3563 012650 110377 166514          MOVVB   R3,2DQREG      ;SELECT BCC POLYNOMIAL REGISTER
3564 012654 012705 000004          MOV      #BIT2,R5      ;(R5)=BIT2
3565 012660 010577 166506          MOV      R5,2DQSEC     ;SET BIT2 IN
3566                                ;BCC POLYNOMIAL REGISTER
3567 012664 017704 166502          MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
3568                                ;BCC POLYNOMIAL REGISTER
3569 012670 020504          CMP      R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3570 012672 001401          BEQ      15          ;BR IF GOOD
3571 012674 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3572 012676 040577 166470 15:  BIC      R5,2DQSEC ;CLEAR BIT2
3573 012702 017704 166464  MOV      2DQSEC,R4 ;READ BCC POLYNOMIAL REGISTER
3574 012706 005005          CLR      R5          ;EXPECT BCC POLYNOMIAL REGISTER
3575                                ;TO CONTAIN 0
3576 012710 020504          CMP      R5,R4      ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3577 012712 001401          BEQ      25          ;BR IF GOOD
3578 012714 104006          HLT      6          ;SECONDARY REGISTER DATA ERROR
3579 012716 104400 25:  SCOPE      ;CHECK FOR ITERATIONS, LOOP
3580
3581                                ;SECONDARY REGISTER READ/WRITE TEST
3582                                ;SET BIT3 IN BCC POLYNOMIAL REGISTER
3583                                ;VERIFY THAT BIT3 WAS SET
3584                                ;CLEAR BIT3
3585                                ;VERIFY THAT BIT3 WAS CLEARED
3586
3587                                ; TEST 112
3588                                ;*****
3589 012720 012737 000112 001226 †TST112: MOV      #112,TSTNO
3590 012726 012737 013010 001216  MOV      #TST113,NEXT
3591 012734 012703 000017          MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
3592                                ;BCC POLYNOMIAL
3593 012740 110377 166424          MOVVB   R3,2DQREG      ;SELECT BCC POLYNOMIAL REGISTER
3594 012744 012705 000010          MOV      #BIT3,R5      ;(R5)=BIT3
3595 012750 010577 166416          MOV      R5,2DQSEC     ;SET BIT3 IN

```

```

3596
3597 012754 017704 166412          MOV      2DQSEC,R4          ;BCC POLYNOMIAL REGISTER
3598                                     ;(R4)=ACTUAL DATA IN
3599 012760 020504          CMP      R5,R4          ;BCC POLYNOMIAL REGISTER
3600 012762 001401          BEQ      1$            ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3601 012764 104006          HLT      6             ;BR IF GOOD
3602 012766 040577 166400      1$: BIC      R5,2DQSEC    ;SECONDARY REGISTER DATA ERROR
3603 012772 017704 166374      MOV      2DQSEC,R4    ;CLEAR BIT3
3604 012776 005005          CLR      R5           ;READ BCC POLYNOMIAL REGISTER
3605                                     ;EXPECT BCC POLYNOMIAL REGISTER
3606 013000 020504          CMP      R5,R4          ;TO CONTAIN 0
3607 013002 001401          BEQ      2$            ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3608 013004 104006          HLT      6             ;BR IF GOOD
3609 013006 104400      2$: SCOPE            ;SECONDARY REGISTER DATA ERROR
3610                                     ;CHECK FOR ITERATIONS, LOOP
3611                                     ;SECONDARY REGISTER READ/WRITE TEST
3612                                     ;SET BIT4 IN BCC POLYNOMIAL REGISTER
3613                                     ;VERIFY THAT BIT4 WAS SET
3614                                     ;CLEAR BIT4
3615                                     ;VERIFY THAT BIT4 WAS CLEARED
3616
3617 ; TEST 113
3618 ;*****
3619 013010 012737 000113 001226  TST113: MOV      #113,TSTNO
3620 013016 012737 013100 001216  MOV      #TST114,NEXT
3621 013024 012703 000017          MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
3622                                     ;BCC POLYNOMIAL
3623 013030 110377 166334          MOV      R3,2DQREG    ;SELECT BCC POLYNOMIAL REGISTER
3624 013034 012705 000020          MOV      #BIT4,R5     ;(R5)=BIT4
3625 013040 010577 166326          MOV      R5,2DQSEC    ;SET BIT4 IN
3626                                     ;BCC POLYNOMIAL REGISTER
3627 013044 017704 166322          MOV      2DQSEC,R4    ;(R4)=ACTUAL DATA IN
3628                                     ;BCC POLYNOMIAL REGISTER
3629 013050 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3630 013052 001401          BEQ      1$            ;BR IF GOOD
3631 013054 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3632 013056 040577 166310      1$: BIC      R5,2DQSEC    ;CLEAR BIT4
3633 013062 017704 166304      MOV      2DQSEC,R4    ;READ BCC POLYNOMIAL REGISTER
3634 013066 005005          CLR      R5           ;EXPECT BCC POLYNOMIAL REGISTER
3635                                     ;TO CONTAIN 0
3636 013070 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3637 013072 001401          BEQ      2$            ;BR IF GOOD
3638 013074 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3639 013076 104400      2$: SCOPE            ;CHECK FOR ITERATIONS, LOOP
3640                                     ;SECONDARY REGISTER READ/WRITE TEST
3641                                     ;SET BITS IN BCC POLYNOMIAL REGISTER
3642                                     ;VERIFY THAT BITS WAS SET
3643                                     ;CLEAR BITS
3644                                     ;VERIFY THAT BITS WAS CLEARED
3645
3646 ; TEST 114
3647 ;*****
3648
3649 013100 012737 000114 001226  TST114: MOV      #114,TSTNO
3650 013106 012737 013170 001216  MOV      #TST115,NEXT
3651 013114 012703 000017          MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
    
```

```

3652 ;BCC POLYNOMIAL
3653 013120 110377 166244      MOVB   R3,2DQREG ;SELECT BCC POLYNOMIAL REGISTER
3654 013124 012705 000040      MOV    #BIT5,R5  ;(R5)=BIT5
3655 013130 010577 166236      MOV    R5,2DQSEC ;SET BITS IN
3656 ;BCC POLYNOMIAL REGISTER
3657 013134 017704 166232      MOV    2DQSEC,R4 ;(R4)=ACTUAL DATA IN
3658 ;BCC POLYNOMIAL REGISTER
3659 013140 020504              CMP    R5,R4     ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3660 013142 001401              BEQ    1$        ;BR IF GOOD
3661 013144 104006              HLT    6         ;SECONDARY REGISTER DATA ERROR
3662 013146 040577 166220      BIC    R5,2DQSEC ;CLEAR BITS
3663 013152 017704 166214      MOV    2DQSEC,R4 ;READ BCC POLYNOMIAL REGISTER
3664 013156 005005              CLR    R5        ;EXPECT BCC POLYNOMIAL REGISTER
3665 ;TO CONTAIN 0
3666 013160 020504              CMP    R5,R4     ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3667 013162 001401              BEQ    2$        ;BR IF GOOD
3668 013164 104006              HLT    6         ;SECONDARY REGISTER DATA ERROR
3669 013166 104400      2$:   SCOPE      ;CHECK FOR ITERATIONS, LOOP
3670
3671 ;SECONDARY REGISTER READ/WRITE TEST
3672 ;SET BIT6 IN BCC POLYNOMIAL REGISTER
3673 ;VERIFY THAT BIT6 WAS SET
3674 ;CLEAR BIT6
3675 ;VERIFY THAT BIT6 WAS CLEARED
3676
3677 ; TEST 115
3678 ;*****
3679 013170 012737 000115 001226 1$T115: MOV    #115,TSTNO
3680 013176 012737 013260 001216      MOV    #TST116,NEXT
3681 013204 012703 000017      MOV    #17,R3    ;ADDRESS OF SECONDARY REGISTER
3682 ;BCC POLYNOMIAL
3683 013210 110377 166154      MOVB   R3,2DQREG ;SELECT BCC POLYNOMIAL REGISTER
3684 013214 012705 000100      MOV    #BIT6,R5  ;(R5)=BIT6
3685 013220 010577 166146      MOV    R5,2DQSEC ;SET BIT6 IN
3686 ;BCC POLYNOMIAL REGISTER
3687 013224 017704 166142      MOV    2DQSEC,R4 ; (R4)=ACTUAL DATA IN
3688 ;BCC POLYNOMIAL REGISTER
3689 013230 020504              CMP    R5,R4     ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3690 013232 001401              BEQ    1$        ;BR IF GOOD
3691 013234 104006              HLT    6         ;SECONDARY REGISTER DATA ERROR
3692 013236 040577 166130      BIC    R5,2DQSEC ;CLEAR BIT6
3693 013242 017704 166124      MOV    2DQSEC,R4 ;READ BCC POLYNOMIAL REGISTER
3694 013246 005005              CLR    R5        ;EXPECT BCC POLYNOMIAL REGISTER
3695 ;TO CONTAIN 0
3696 013250 020504              CMP    R5,R4     ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3697 013252 001401              BEQ    2$        ;BR IF GOOD
3698 013254 104006              HLT    6         ;SECONDARY REGISTER DATA ERROR
3699 013256 104400      2$:   SCOPE      ;CHECK FOR ITERATIONS, LOOP
3700
3701 ;SECONDARY REGISTER READ/WRITE TEST
3702 ;SET BIT7 IN BCC POLYNOMIAL REGISTER
3703 ;VERIFY THAT BIT7 WAS SET
3704 ;CLEAR BIT7
3705 ;VERIFY THAT BIT7 WAS CLEARED
3706
3707 ; TEST 116

```

K06

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 75
 DZDQAC.P11 16-DEC-76 11:24 POLYNOMIAL REGISTER READ/WRITE TESTS.

```

3708
3709 013260 012737 000116 001226 *ST116: MOV #116,TSTNO
3710 013266 012737 013350 001216 MOV #TST117,NEXT
3711 013274 012703 000017 MOV #17,R3
3712
3713 013300 110377 166064 MOV# R3,ADQREG
3714 013304 012705 000200 MOV #BIT7,R5
3715 013310 010577 166056 MOV R5,ADQSEC
3716
3717 013314 017704 166052 MOV ADQSEC,R4
3718
3719 013320 020504 CMP R5,R4
3720 013322 001401 BEQ 1$
3721 013324 104006 HLT 6
3722 013326 040577 166040 1$: BIC R5,ADQSEC
3723 013332 017704 166034 MOV ADQSEC,R4
3724 013336 005005 CLR R5
3725
3726 013340 020504 CMP R5,R4
3727 013342 001401 BEQ 2$
3728 013344 104006 HLT 6
3729 013346 104400 2$: SCOPE
3730
3731 ;SECONDARY REGISTER READ/WRITE TEST
3732 ;SET BIT8 IN BCC POLYNOMIAL REGISTER
3733 ;VERIFY THAT BIT8 WAS SET
3734 ;CLEAR BIT8
3735 ;VERIFY THAT BIT8 WAS CLEARED
3736
3737 ; TEST 117
3738 ;*****
3739 013350 012737 000117 001226 *ST117: MOV #117,TSTNO
3740 013356 012737 013440 001216 MOV #TST120,NEXT
3741 013364 012703 000017 MOV #17,R3
3742
3743 013370 110377 165774 MOV# R3,ADQREG
3744 013374 012705 000400 MOV #BIT8,R5
3745 013400 010577 165766 MOV R5,ADQSEC
3746
3747 013404 017704 165762 MOV ADQSEC,R4
3748
3749 013410 020504 CMP R5,R4
3750 013412 001401 BEQ 1$
3751 013414 104006 HLT 6
3752 013416 040577 165750 1$: BIC R5,ADQSEC
3753 013422 017704 165744 MOV ADQSEC,R4
3754 013426 005005 CLR R5
3755
3756 013430 020504 CMP R5,R4
3757 013432 001401 BEQ 2$
3758 013434 104006 HLT 6
3759 013436 104400 2$: SCOPE
3760
3761 ;SECONDARY REGISTER READ/WRITE TEST
3762 ;SET BIT9 IN BCC POLYNOMIAL REGISTER
3763 ;VERIFY THAT BIT9 WAS SET
  
```

```

;*****
;ADDRESS OF SECONDARY REGISTER
;BCC POLYNOMIAL
;SELECT BCC POLYNOMIAL REGISTER
;(R5)=BIT7
;SET BIT7 IN
;BCC POLYNOMIAL REGISTER
;(R4)=ACTUAL DATA IN
;BCC POLYNOMIAL REGISTER
;ARE EXPECTED AND RECEIVED DATA THE SAME ?
;BR IF GOOD
;SECONDARY REGISTER DATA ERROR
;CLEAR BIT7
;READ BCC POLYNOMIAL REGISTER
;EXPECT BCC POLYNOMIAL REGISTER
;TO CONTAIN 0
;ARE EXPECTED AND RECEIVED DATA THE SAME ?
;BR IF GOOD
;SECONDARY REGISTER DATA ERROR
;CHECK FOR ITERATIONS, LOOP
  
```

```

;*****
;ADDRESS OF SECONDARY REGISTER
;BCC POLYNOMIAL
;SELECT BCC POLYNOMIAL REGISTER
;(R5)=BIT8
;SET BIT8 IN
;BCC POLYNOMIAL REGISTER
;(R4)=ACTUAL DATA IN
;BCC POLYNOMIAL REGISTER
;ARE EXPECTED AND RECEIVED DATA THE SAME ?
;BR IF GOOD
;SECONDARY REGISTER DATA ERROR
;CLEAR BIT8
;READ BCC POLYNOMIAL REGISTER
;EXPECT BCC POLYNOMIAL REGISTER
;TO CONTAIN 0
;ARE EXPECTED AND RECEIVED DATA THE SAME ?
;BR IF GOOD
;SECONDARY REGISTER DATA ERROR
;CHECK FOR ITERATIONS, LOOP
  
```

```

3764                                     ;CLEAR BIT9
3765                                     ;VERIFY THAT BIT9 WAS CLEARED
3766
3767                                     ; TEST 120
3768                                     ;*****
3769 013440 012737 000120 001226 †TST120: MOV #120,TSTNO
3770 013446 012737 013530 001216      MOV #TST121,NEXT
3771 013454 012703 000017              MOV #17,R3
3772                                     ; ADDRESS OF SECONDARY REGISTER
3773 013460 110377 165704              MOVB R3,2DQREG
3774 013464 012705 001000              MOV #BIT9,R5
3775 013470 010577 165676              MOV R5,2DQSEC
3776                                     ; BCC POLYNOMIAL REGISTER
3777 013474 017704 165672              MOV 2DQSEC,R4
3778                                     ; (R4)=ACTUAL DATA IN
3779 013500 020504                      CMP R5,R4
3780 013502 001401                      BEQ 1$
3781 013504 104006                      HLT 6
3782 013506 040577 165660              1$: BIC R5,2DQSEC
3783 013512 017704 165654              MOV 2DQSEC,R4
3784 013516 005005                      CLR R5
3785                                     ; BCC POLYNOMIAL REGISTER
3786 013520 020504                      CMP R5,R4
3787 013522 001401                      BEQ 2$
3788 013524 104006                      HLT 6
3789 013526 104400                      2$: SCOPE
3790                                     ; SECONDARY REGISTER READ/WRITE TEST
3791                                     ; SET BIT10 IN BCC POLYNOMIAL REGISTER
3792                                     ; VERIFY THAT BIT10 WAS SET
3793                                     ; CLEAR BIT10
3794                                     ; VERIFY THAT BIT10 WAS CLEARED
3795
3796                                     ; TEST 121
3797                                     ;*****
3798                                     ;*****
3799 013530 012737 000121 001226 †TST121: MOV #121,TSTNO
3800 013536 012737 013620 001216      MOV #TST122,NEXT
3801 013544 012703 000017              MOV #17,R3
3802                                     ; ADDRESS OF SECONDARY REGISTER
3803 013550 110377 165614              MOVB R3,2DQREG
3804 013554 012705 002000              MOV #BIT10,R5
3805 013560 010577 165606              MOV R5,2DQSEC
3806                                     ; BCC POLYNOMIAL REGISTER
3807 013564 017704 165602              MOV 2DQSEC,R4
3808                                     ; (R4)=ACTUAL DATA IN
3809 013570 020504                      CMP R5,R4
3810 013572 001401                      BEQ 1$
3811 013574 104006                      HLT 6
3812 013576 040577 165570              1$: BIC R5,2DQSEC
3813 013602 017704 165564              MOV 2DQSEC,R4
3814 013606 005005                      CLR R5
3815                                     ; BCC POLYNOMIAL REGISTER
3816 013610 020504                      CMP R5,R4
3817 013612 001401                      BEQ 2$
3818 013614 104006                      HLT 6
3819 013616 104400                      2$: SCOPE

```

```

3820
3821      ; SECONDARY REGISTER READ/WRITE TEST
3822      ; SET BIT11 IN BCC POLYNOMIAL REGISTER
3823      ; VERIFY THAT BIT11 WAS SET
3824      ; CLEAR BIT11
3825      ; VERIFY THAT BIT11 WAS CLEARED
3826
3827      ; TEST 122
3828      ; *****
3829 013620 012737 000122 001226 TST122: MOV #122,TSTNO
3830 013626 012737 013710 001216      MOV #TST123,NEXT
3831 013634 012703 000017      MOV #17,R3
3832      ; ADDRESS OF SECONDARY REGISTER
3833 013640 110377 165524      MOVB R3,JDQREG
3834 013644 012705 004000      MOV #BIT11,R5
3835 013650 010577 165516      MOV R5,JDQSEC
3836      ; BCC POLYNOMIAL REGISTER
3837 013654 017704 165512      MOV JDQSEC,R4
3838      ; (R4)=ACTUAL DATA IN
3839 013660 020504      CMP R5,R4
3840 013662 001401      BEQ 1$
3841 013664 104006      HLT 6
3842 013666 040577 165500 1$: BIC R5,JDQSEC
3843 013672 017704 165474      MOV JDQSEC,R4
3844 013676 005005      CLR R5
3845      ; BCC POLYNOMIAL REGISTER
3846 013700 020504      CMP R5,R4
3847 013702 001401      BEQ 2$
3848 013704 104006      HLT 6
3849 013706 104400 2$: SCOPE
3850      ; CHECK FOR ITERATIONS, LOOP
3851      ; SECONDARY REGISTER READ/WRITE TEST
3852      ; SET BIT12 IN BCC POLYNOMIAL REGISTER
3853      ; VERIFY THAT BIT12 WAS SET
3854      ; CLEAR BIT12
3855      ; VERIFY THAT BIT12 WAS CLEARED
3856
3857      ; TEST 123
3858      ; *****
3859 013710 012737 000123 001226 TST123: MOV #123,TSTNO
3860 013716 012737 014000 001216      MOV #TST124,NEXT
3861 013724 012703 000017      MOV #17,R3
3862      ; ADDRESS OF SECONDARY REGISTER
3863 013730 110377 165434      MOVB R3,JDQREG
3864 013734 012705 010000      MOV #BIT12,R5
3865 013740 010577 165426      MOV R5,JDQSEC
3866      ; BCC POLYNOMIAL REGISTER
3867 013744 017704 165422      MOV JDQSEC,R4
3868      ; (R4)=ACTUAL DATA IN
3869 013750 020504      CMP R5,R4
3870 013752 001401      BEQ 1$
3871 013754 104006      HLT 6
3872 013756 040577 165410 1$: BIC R5,JDQSEC
3873 013762 017704 165404      MOV JDQSEC,R4
3874 013766 005005      CLR R5
3875      ; BCC POLYNOMIAL REGISTER
3876      ; EXPECT BCC POLYNOMIAL REGISTER
3877      ; TO CONTAIN 0

```

```

3876 013770 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3877 013772 001401          BEQ      25             ;BR IF GOOD
3878 013774 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3879 013776 104400          25:     SCOPE          ;CHECK FOR ITERATIONS, LOOP
3880
3881                          ;SECONDARY REGISTER READ/WRITE TEST
3882                          ;SET BIT13 IN BCC POLYNOMIAL REGISTER
3883                          ;VERIFY THAT BIT13 WAS SET
3884                          ;CLEAR BIT13
3885                          ;VERIFY THAT BIT13 WAS CLEARED
3886
3887                          ; TEST 124
3888                          ;*****
3889 014000 012737 000124 001226 1ST124: MOV      #124,TSTNO
3890 014006 012737 014070 001216      MOV      #TST125,NEXT
3891 014014 012703 000017              MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
3892                          ;BCC POLYNOMIAL
3893 014020 110377 165344          MOVB     R3,2DQREG      ;SELECT BCC POLYNOMIAL REGISTER
3894 014024 012705 020000          MOV      #BIT13,R5     ;(R5)=BIT13
3895 014030 010577 165336          MOV      R5,2DQSEC     ;SET BIT13 IN
3896                          ;BCC POLYNOMIAL REGISTER
3897 014034 017704 165332          MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
3898                          ;BCC POLYNOMIAL REGISTER
3899 014040 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3900 014042 001401          BEQ      15             ;BR IF GOOD
3901 014044 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3902 014046 040577 165320          15:     BIC      R5,2DQSEC ;CLEAR BIT13
3903 014052 017704 165314          MOV      2DQSEC,R4     ;READ BCC POLYNOMIAL REGISTER
3904 014056 005005          CLR      R5            ;EXPECT BCC POLYNOMIAL REGISTER
3905                          ;TO CONTAIN 0
3906 014060 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3907 014062 001401          BEQ      25             ;BR IF GOOD
3908 014064 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
3909 014066 104400          25:     SCOPE          ;CHECK FOR ITERATIONS, LOOP
3910
3911                          ;SECONDARY REGISTER READ/WRITE TEST
3912                          ;SET BIT14 IN BCC POLYNOMIAL REGISTER
3913                          ;VERIFY THAT BIT14 WAS SET
3914                          ;CLEAR BIT14
3915                          ;VERIFY THAT BIT14 WAS CLEARED
3916
3917                          ; TEST 125
3918                          ;*****
3919 014070 012737 000125 001226 1ST125: MOV      #125,TSTNO
3920 014076 012737 014160 001216      MOV      #TST126,NEXT
3921 014104 012703 000017              MOV      #17,R3          ;ADDRESS OF SECONDARY REGISTER
3922                          ;BCC POLYNOMIAL
3923 014110 110377 165254          MOVB     R3,2DQREG      ;SELECT BCC POLYNOMIAL REGISTER
3924 014114 012705 040000          MOV      #BIT14,R5     ;(R5)=BIT14
3925 014120 010577 165246          MOV      R5,2DQSEC     ;SET BIT14 IN
3926                          ;BCC POLYNOMIAL REGISTER
3927 014124 017704 165242          MOV      2DQSEC,R4     ;(R4)=ACTUAL DATA IN
3928                          ;BCC POLYNOMIAL REGISTER
3929 014130 020504          CMP      R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3930 014132 001401          BEQ      15             ;BR IF GOOD
3931 014134 104006          HLT      6             ;SECONDARY REGISTER DATA ERROR
    
```

```

3932 014136 040577 165230      1S:  BIC    R5,2DQSEC      ;CLEAR BIT14
3933 014142 017704 165224      MOV    2DQSEC,R4      ;READ BCC POLYNOMIAL REGISTER
3934 014146 005005                CLR    R5              ;EXPECT BCC POLYNOMIAL REGISTER
3935                                ;TO CONTAIN 0
3936 014150 020504                CMP    R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3937 014152 001401                BEQ    2S              ;BR IF GOOD
3938 014154 104006                HLT    6               ;SECONDARY REGISTER DATA ERROR
3939 014156 104400      2S:  SCOPE            ;CHECK FOR ITERATIONS, LOOP
3940
3941                                ;SECONDARY REGISTER READ/WRITE TEST
3942                                ;SET BIT15 IN BCC POLYNOMIAL REGISTER
3943                                ;VERIFY THAT BIT15 WAS SET
3944                                ;CLEAR BIT15
3945                                ;VERIFY THAT BIT15 WAS CLEARED
3946
3947                                ; TEST 126
3948                                ;*****
3949 014160 012737 000126 001226  TST126: MOV    #126,TSTNO
3950 014166 012737 014250 001216  MOV    #.EOP,NEXT
3951 014174 012703 000017                MOV    #17,R3        ;ADDRESS OF SECONDARY REGISTER
3952                                ;BCC POLYNOMIAL
3953 014200 110377 165164                MOVB   R3,2DQREG     ;SELECT BCC POLYNOMIAL REGISTER
3954 014204 012705 100000                MOV    #BIT15,R5     ;(R5)=BIT15
3955 014210 010577 165156                MOV    R5,2DQSEC     ;SET BIT15 IN
3956                                ;BCC POLYNOMIAL REGISTER
3957 014214 017704 165152                MOV    2DQSEC,R4     ;(R4)=ACTUAL DATA IN
3958                                ;BCC POLYNOMIAL REGISTER
3959 014220 020504                CMP    R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3960 014222 001401                BEQ    1S              ;BR IF GOOD
3961 014224 104006                HLT    6               ;SECONDARY REGISTER DATA ERROR
3962 014226 040577 165140      1S:  BIC    R5,2DQSEC     ;CLEAR BIT15
3963 014232 017704 165134      MOV    2DQSEC,R4     ;READ BCC POLYNOMIAL REGISTER
3964 014236 005005                CLR    R5              ;EXPECT BCC POLYNOMIAL REGISTER
3965                                ;TO CONTAIN 0
3966 014240 020504                CMP    R5,R4          ;ARE EXPECTED AND RECEIVED DATA THE SAME ?
3967 014242 001401                BEQ    2S              ;BR IF GOOD
3968 014244 104006                HLT    6               ;SECONDARY REGISTER DATA ERROR
3969 014246 104400      2S:  SCOPE            ;CHECK FOR ITERATIONS, LOOP

```



```

3970
3971
3972
3973
3974
3975
3976
3977 014250 005037 001234 - .EOP: CLR LSTERR ;CLEAR LAST ERROR PC
3978 014254 005037 001312 CLR ERRFLG ;CLEAR ERROR FLAG
3979 014260 005237 001230 INC PASCNT ;UPDATE PASS COUNT
3980 014264 104402 TYPE
3981 014266 016500 MEPASS
3982 014270 104402 TYPE
3983 014272 016661 MCSRX
3984 014274 104411 CNVRT
3985 014276 014406 XCSR
3986 014300 104402 TYPE
3987 014302 016667 MVECX
3988 014304 104411 CNVRT
3989 014306 014414 XVEC
3990 014310 104402 TYPE
3991 014312 016675 MPASSX
3992 014314 104411 CNVRT
3993 014316 014422 XPASS
3994 014320 104402 TYPE
3995 014322 016706 MERRX
3996 014324 104411 CNVRT
3997 014326 014430 XERR
3998 014330 013777 001230 164644 MOV PASCNT,ALIGHTS ;DISPLAY PASS COUNT
3999 014336 005337 001276 DEC SAVNUM
4000 014342 001013 BNE RESTRT
4001 014344 013737 001504 001276 MOV DQNUM,SAVNUM
4002 014352 013701 000042 MOV #42,R1 ;CHECK FOR ACT-11 OR DDP
4003 014356 001405 BEQ RESTRT ;IF NOT, CONTINUE TESTING
4004 014360 000005 RESET
4005 014362 LOGICAL:
4006 014362 004711 JSR PC,(R1)
4007 014364 000240 NOP
4008 014366 000240 NOP
4009 014370 000240 NOP
4010 014372 104414 RESTRT: CKSWR
4011 014374 012737 002254 001214 MOV #TST1,RETURN
4012 014402 000137 002254 JMP TST1
4013 014406 000001 XCSR: 1
4014 014410 006 002 .BYTE 6,2
4015 014412 001360 DQRCSR
4016 014414 000001 XVEC: 1
4017 014416 003 002 .BYTE 3,2
4018 014420 001350 DQRVEC
4019 014422 000001 XPASS: 1
4020 014424 006 002 .BYTE 6,2
4021 014426 001230 PASCNT
4022 014430 000001 XERR: 1
4023 014432 006 002 .BYTE 6,2
4024 014434 001232 ERRCNT
4025
    
```

```

;SCOPE LOOP AND INTERATION HANDLER
4026
4027
4028 014436 104414 .SCOPE: CKSWR
4029 014440 032777 040000 164532 BIT #BIT14,ASWR
4030 014446 001407 TTST: BEQ 1$
4031 014450 000432 BR 3$
4032 014452 105777 164526 TSTB @TKCSR
4033 014456 100027 BPL 3$
4034 014460 017700 164522 MOV @TKDBR,RO
4035 014464 000412 BR 2$
4036 014466 032777 004000 164504 1$: BIT #SW11,ASWR
4037 014474 001006 BNE 2$
4038 014476 005237 001224 INC LPCNT
4039 014502 023737 001224 001222 CMP LPENT,ICOUNT
4040 014510 001012 BNE 3$
4041 014512 105037 001312 2$: CLRB ERRFLG
4042 014516 005037 001224 CLR LPCNT
4043 014522 012737 002000 001222 MOV #2000,ICOUNT
4044 014530 013737 001216 001214 MOV NEXT,RETURN
4045 014536 013716 001214 3$: MOV RETURN,(SP)
4046 014542 000002 RTI
4047 014544 001407 BRW: 1407
4048 014546 000432 BRX: 432
4049
4050 ;CHECK FOR FREEZE ON CURRENT DATA
4051
4052 014550 104414 .SCOPE1: CKSWR
4053 014552 032777 001000 164420 BIT #SW09,ASWR
4054 014560 001402 BEQ 1$
4055 014562 013716 001220 MOV LOCK,(SP)
4056 014566 000002 1$: RTI
4057
4058 ;TELETYPE OUTPUT ROUTINE
4059
4060 014570 010546 .TYPE: MOV R5, -(SP)
4061 014572 017605 000002 MOV @2(SP),R5
4062 014576 062766 000002 000002 ADD #2,2(SP)
4063 014604 005737 016260 1$: TST @RDSW
4064 014610 001004 BNE 300$
4065 014612 032777 010000 164360 BIT #SW12,ASWR
4066 014620 001024 BNE 3$
4067 014622 105715 300$: TSTB (R5)
4068 014624 100014 BPL 2$
4069 014626 105777 164356 TSTB @TPCSR
4070 014632 100375 BPL .-4
4071 014634 012777 000015 164350 MOV #15,@TPDBR
4072 014642 105777 164342 TSTB @TPCSR
4073 014646 100375 BPL .-4
4074 014650 012777 000012 164334 MOV #12,@TPDBR
4075 014656 105777 164326 2$: TSTB @TPCSR
4076 014662 100375 BPL 2$
4077 014664 112577 164322 MOVB (R5)+,@TPDBR
4078 014670 001345 BNE 1$
4079 014672 012605 3$: MOV (SP)+,R5
4080 014674 000002 RTI
4081

```

```

;ASCII STRING INPUT ROUTINE
4082
4083
4084 014676 010346 .INSTR: MOV R3,-(SP)
4085 014700 010446 MOV R4,-(SP)
4086 014702 017637 000004 014720 MOV 24(SP),MSG
4087 014710 062766 000002 000004 ADD #2,4(SP)
4088 014716 104402 .INST1: TYPE
4089 014720 000000 .MSG: 0
4090 014722 012704 017052 MOV #INBUF,R4
4091 014726 012703 000007 MOV #7,R3
4092 014732 105777 164246 15: TSTB @TKCSR
4093 014736 100375 BPL 15
4094 014740 117714 164242 MOVB @TKDBR,(R4)
4095 014744 142714 000200 BICB #200,(R4)
4096 014750 121427 000025 CMPB (R4),#25 ;IS IT (<G)
4097 014754 001003 BNE 200$
4098 014756 104402 016440 TYPE,MCRLF
4099 014762 000755 BR .INST1
4100 014764 122427 000015 200$: CMPB (R4)+,#15
4101 014770 001423 BEQ INSTR2
4102 014772 117777 164210 164212 MOVB @TKDBR,@TPDBR
4103 015000 105777 164204 25: TSTB @TPCSR
4104 015004 100375 BPL 25
4105 015006 005303 DEC R3
4106 015010 001350 BNE 15
4107 015012 000402 BR .INSTG
4108 015014 010346 .INSTE: MOV R3,-(SP)
4109 015016 010446 MOV R4,-(SP)
4110 015020 104402 .INSTG: TYPE
4111 015022 016434 MQM
4112 015024 005737 016260 TST @RDSW
4113 015030 001402 BEQ 400$
4114 015032 104402 016440 TYPE,MCRLF
4115 015036 000727 400$: BR .INST1
4116 015040 012604 INSTR2: MOV (SP)+,R4
4117 015042 012603 MOV (SP)+,R3
4118 015044 000002 RTI

;CONVERT ASCII STRING TO OCTAL
4120
4121
4122 015046 010546 .PARAM: MOV R5,-(SP)
4123 015050 010446 MOV R4,-(SP)
4124 015052 016605 000004 MOV 4(SP),R5
4125 015056 012537 015252 MOV (R5)+,LOLIM
4126 015062 012537 015254 MOV (R5)+,HILIM
4127 015066 012537 015256 MOV (R5)+,DEVADR
4128 015072 112537 015260 MOVB (R5)+,LOBITS
4129 015076 112537 015261 MOVB (R5)+,ADRCNT
4130 015102 010566 000004 MOV R5,4(SP)
4131 015106 005005 PARAM1: CLR R5
4132 015110 012704 017052 MOV #INBUF,R4
4133 015114 122714 000015 CMPB #15,(R4)
4134 015120 001420 BEQ PARERR
4135 015122 121427 000060 15: CMPB (R4),#60
4136 015126 002415 BLT PARERR
4137 015130 121427 000067 CMPB (R4),#67
    
```

4138	015134	003012			BGT	PARERR	
4139	015136	142714	000060		BICB	#60,(R4)	
4140	015142	152405			BISB	(R4)+,R5	
4141	015144	122714	000015		CMPB	#15,(R4)	
4142	015150	001414			BEQ	LIMITS	
4143	015152	006305			ASL	R5	
4144	015154	006305			ASL	R5	
4145	015156	006305			ASL	R5	
4146	015160	000760			BR	1\$	
4147	015162	122714	000015		PARERR: CMPB	#15,(R4)	; IS FIRST CHARACTER A <CR>
4148	015166	001003			BNE	120\$	
4149	015170	005737	016260		TST	0#RDSW	; IS CKSWR ROUTINE BEING USED
4150	015174	001023			BNE	PARTI	
4151	015176	104404			120\$: INSTER		
4152	015200	000742			BR	PARAM1	
4153							
4154							
4155							
4156	015202	020537	015254		LIMITS: CMP	R5,HILIM	
4157	015206	101365			BHI	PARERR	
4158	015210	020537	015252		CMP	R5,LOLIM	
4159	015214	103762			BLO	PARERR	
4160	015216	133705	015260		BITB	LOBITS,R5	
4161	015222	001357			BNE	PARERR	
4162							
4163							
4164							
4165	015224	013704	015256		1\$: MOV	DEVADR,R4	
4166	015230	010524			MOV	R5,(R4)+	
4167	015232	062705	000002		ADD	#2,R5	
4168	015236	105337	015261		DECB	ADRCNT	
4169	015242	001372			BNE	1\$	
4170	015244	012604			PARTI: MOV	(SP)+,R4	
4171	015246	012605			MOV	(SP)+,R5	
4172	015250	000002			RTI		
4173	015252	000000			LOLIM: 0		
4174	015254	000000			HILIM: 0		
4175	015256	000000			DEVADR: 0		
4176	015260	000000			LOBITS: 0		
4177		015261			ADRCNT=LOBITS+1		
4178							
4179							
4180							
4181	015262	016637	000004	001274	.SAV05: MOV	4(SP),SAVPC	
4182							
4183							
4184							
4185	015270	010537	001270		SV05: MOV	R5,SAVR5	
4186	015274	010437	001266		MOV	R4,SAVR4	
4187	015300	010337	001264		MOV	R3,SAVR3	
4188	015304	010237	001262		MOV	R2,SAVR2	
4189	015310	010137	001260		MOV	R1,SAVR1	
4190	015314	010037	001256		MOV	R0,SAVR0	
4191	015320	000002			RTI		
4192							
4193							

;RESTORE RO-R5

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 84
 DZDQAC.P11 16-DEC-76 11:24 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

4194							
4195	015322	013700	001256			.RESOS:	MOV SAVR0,R0
4196	015326	013701	001260				MOV SAVR1,R1
4197	015332	013702	001262				MOV SAVR2,R2
4198	015336	013703	001264				MOV SAVR3,R3
4199	015342	013704	001266				MOV SAVR4,R4
4200	015346	013705	001270				MOV SAVR5,R5
4201	015352	000002					RTI
4202							
4203							;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
4204							
4205	015354	104402				.CONVR:	TYPE
4206	015356	016440					MCRLF
4207	015360	010046				.CNVRT:	MOV R0,-(SP)
4208	015362	010146					MOV R1,-(SP)
4209	015364	010346					MOV R3,-(SP)
4210	015366	010446					MOV R4,-(SP)
4211	015370	010546					MOV R5,-(SP)
4212	015372	017601	000012				MOV @12(SP),R1
4213	015376	013737	017114	001250			MOV TEMP,TEMP3
4214	015404	062766	000002	000012			ADD #2,12(SP)
4215	015412	012137	015574				MOV (R1)+,WRDCNT
4216	015416	112137	015576			15:	MOVB (R1)+,CHRCNT
4217	015422	112137	015577				MOVB (R1)+,SPACNT
4218	015426	013137	015600				MOV @2(R1)+,BINWRD
4219	015432	013704	015600			25:	MOV BINWRD,R4
4220	015436	113705	015576				MOVB CHRCNT,R5
4221	015442	012700	017114				MOV #TEMP,R0
4222	015446	010403				35:	MOV R4,R3
4223	015450	042703	177770				BIC #177770,R3
4224	015454	062703	000060				ADD #060,R3
4225	015460	110320					MOVB R3,(R0)+
4226	015462	000241					CLC
4227	015464	006004					ROR R4
4228	015466	000241					CLC
4229	015470	006004					ROR R4
4230	015472	000241					CLC
4231	015474	006004					ROR R4
4232	015476	005305					DEC R5
4233	015500	001362					BNE 35
4234	015502	012703	017156				MOV #MDATA,R3
4235	015506	114023				45:	MOVB -(R0),(R3)+
4236	015510	105337	015576				DECB CHRCNT
4237	015514	001374					BNE 45
4238	015516	105737	015577				TSTB SPACNT
4239	015522	001405					BEQ 65
4240	015524	112723	000040			55:	MOVB #040,(R3)+
4241	015530	105337	015577				DECB SPACNT
4242	015534	001373					BNE 55
4243	015536	105013				65:	CLRB (R3)
4244	015540	104402					TYPE
4245	015542	017156					MDATA
4246	015544	005337	015574				DEC WRDCNT
4247	015550	001322					BNE 15
4248	015552	013737	001250	017114			MOV TEMP3,TEMP
4249	015560	012605					MOV (SP)+,R5

H07

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 85
 DZDQAC.P11 16-DEC-76 11:24 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

4250	015562	012604				MOV	(SP)+,R4		
4251	015564	012603				MOV	(SP)+,R3		
4252	015566	012601				MOV	(SP)+,R1		
4253	015570	012600				MOV	(SP)+,R0		
4254	015572	000002				RTI			
4255	015574	000000				WRDCNT:	0		
4256	015576	000000				CHRCNT:	0		
4257		015577				SPACNT=	CHRCNT+1		
4258	015600	000000				BINWRD:	0		
4259								;TRAP DISPATCH SERVICE	
4260								;ARGUMENT OF TRAP IS EXTRACTED	
4261								;AND USED AS OFFSET TO OBTAIN POINTER	
4262								;TO SELECTED SUBROUTINE	
4263									
4264	015603	011646				.TRPSR:	MOV	(SP),-(SP)	;GET PC OF RETURN
4265	015604	162716	000002				SUB	#2,(SP)	;=PC OF TRAP
4266	015610	017616	000000				MOV	@(SP),(SP)	;GET TRP
4267	015614	006316				TRPOK:	ASL	(SP)	;MULTIPLY TRAP ARG BY 2
4268	015616	042716	177001				BIC	#177001,(SP)	;CLEAR UNWANTED BITS
4269	015622	062716	001314				ADD	#.TRPTAB,(SP)	;POINTER TO SUBROUTINE ADDRESS
4270	015626	017616	000000				MOV	@(SP),(SP)	;SUBROUTINE ADDRESS
4271	015632	000136					JMP	@(SP)+	;GO TO SUBROUTINE
4272									
4273									
4274									;ERROR HANDLER
4275	015634	104414				.HLT:	CKSWR		
4276	015636	032777	010000	163334			BIT	#SW12,@SWR	
4277	015644	001406					BEQ	XBX	
4278	015646	105777	163336				TSTB	@TPCSR	
4279	015652	100003					BPL	XBX	
4280	015654	112777	000207	163330			MOVB	#207,@TPDBR	
4281	015662	032777	020000	163310	XBX:		BIT	#SW13,@SWR	
4282	015670	001074					BNE	HALTS	
4283	015672	021637	001234				CMP	(SP),LSTERR	
4284	015676	001404					BEQ	1\$	
4285	015700	011637	001234				MOV	(SP),LSTERR	
4286	015704	105037	001312				CLRB	ERRFLG	
4287	015710	104406			1\$:		SAVOS		
4288	015712	011605					MOV	(SP),R5	
4289	015714	162705	000002				SUB	#2,R5	
4290	015720	011504					MOV	(R5),R4	
4291	015722	006304					ASL	R4	
4292	015724	061504					ADD	(R5),R4	
4293	015726	006304					ASL	R4	
4294	015730	042704	177001				BIC	#177001,R4	
4295	015734	062704	017670				ADD	#.ERRTAB,R4	
4296	015740	012437	016032				MOV	(R4)+,ERRMSG	
4297	015744	012437	016044				MOV	(R4)+,DATAHD	
4298	015750	011437	016056				MOV	(R4),DATABP	
4299	015754	105737	001312				TSTB	ERRFLG	
4300	015760	001403					BEQ	TYPMSG	
4301	015762	005737	016056				TST	DATABP	
4302	015766	001027					BNE	TYPDAT	
4303	015770	104402				TYPMSG:	TYPE		
4304	015772	016717					MTSTN		
4305	015774	104411					CNVRT		

4306	015776	016156				XTSTN	
4307	016000	104402				TYPE	
4308	016002	017005				MERRPC	
4309	016004	104411				CNVRT	
4310	016006	016150				ERTABO	
4311	016010	104402				TYPE	
4312	016012	016440				MCRLF	
4313	016014	112737	177777	001312		MOV8	#-1,ERRFLG
4314	016022	005737	016032			TST	ERRMSG
4315	016026	001402				BEQ	WRKO.FM
4316	016030	104402				TYPE	
4317	016032	000000				ERRMSG: 0	
4318	016034					WRKO.FM:	
4319	016034	005737	016044			TST	DATAHD
4320	016040	001402				BEQ	TYPDAT
4321	016042	104402				TYPE	
4322	016044	000000				DATAHD: 0	
4323	016046	005737	016056			TYPDAT: TST	DATABP
4324	016052	001402				BEQ	RESREG
4325	016054	104410				CONVRT	
4326	016056	000000				DATABP: 0	
4327	016060	104407				RESREG: RES05	
4328	016062	005777	163112			HALTS: TST	2SWR
4329	016066	100005				BPL	EXITER
4330	016070	010046				PUSHRO	
4331	016072	016600	000002			MOV	2(SP),RO
4332	016076	000000				HALT	
4333	016100	012600				POPPO	
4334	016102	104414				EXITER: CKSWR	
4335	016104	005237	001232			INC	ERRCNT
4336	016110	032777	000400	163062		BIT	#SW08,2SWR
4337	016116	001007				BNE	1\$
4338	016120	032777	002000	163052		BIT	#SW10,2SWR
4339	016126	001407				BEQ	2\$
4340	016130	013737	001216	001214		MOV	NEXT,RETURN
4341	016136	012706	001200			1\$: MOV	#STACK,SP
4342	016142	000177	163046			JMP	2RETURN
4343	016146	000002				2\$: RTI	
4344	016150	000001				ERTABO: 1	
4345	016152	006	002			.BYTE	6,2
4346	016154	001274				SAVPC	
4347	016156	000001				XTSTN: 1	
4348	016160	003	002			.BYTE	3,2
4349	016162	001226				TSTNO	
4350						;ENTER HERE ON POWER FAILURE	
4351							
4352							
4353	016164					.PFAIL:	
4354	016164	012737	016176	000024		MOV	#RESTART,24 ;SET UP FOR POWER UP TRAP
4355	016172	000000				HALT	;HALT ON POWER DOWN NORMAL
4356	016174	000777				BR	
4357							
4358						;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED	
4359							
4360	016176					RESTAR:	
4361	016176	012737	016164	000024		MOV	1.PFAIL,24 ;SET UP FOR POWER FAILURE

```

4362 016204 012706 001200
4363 016210 005037 017114
4364 016214 005237 017114
4365 016220 001375
4366 016222 104402
4367 016224 016442
4368 016226 104411
4369 016230 016252
4370 016232 005037 001312
4371 016236 005037 001234
4372 016242 104412
4373 016244 104413
4374 016246 000177 162742
4375 016252 000001
4376 016254 003 002
4377 016256 001226
4378
4379
4380
4381
4382
4383 016260 000000
4384
4385
4386 016262 005737 000042
4387 016266 001042
4388 016270 022737 000176 001200
4389 016276 001036
4390 016300 105777 162700
4391 016304 100033
4392 016306 017737 162674 014720
4393 016314 042737 177600 014720
4394 016322 122737 000007 014720
4395 016330 001021
4396 016332 104402 016410
4397 016336 005137 016260
4398 016342 104402 016414
4399 016346 104411 016402
4400 016352 104403 016423
4401 016356 104405
4402 016360 000000
4403 016362 177777
4404 016364 000176
4405 016366 000 001
4406 016370 104402 016440
4407 016374 005037 016260
4408 016400 000002
4409 016402 000001
4410 016404 006 002
4411 016406 000176
4412 016410 057377 000107
4413 016414 051777 051127 020075
4414 016422 000
4415 016423 040 047040 053505
4416 016430 020075 000
4417 016434

```

```

MOV #STACK, SP
CLR TEMP
INC TEMP
BNE .-4
TYPE
MPFAIL
CNVRT
PFTAB
CLR ERRFLG
CLR LSTERR
MSTCLR
MEMCLR
JMP @RETURN
PFTAB: 1
.BYTE 3,2
TSTNO

```

```

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

```

```

.CKSWR: TST @42
BNE OUT
CMP #SWREG, SWR ;SOFTWARE SWITCH REGISTER PRESENT
BNE OUT ;NO GET OUT
TSTB @TKCSR ;YES, WAIT FOR
BPL OUT ;READY, GET CHARACTER
MOV @TKDBR, .MSG ;AND STRIP OFF
BIC @177600, .MSG ;THE GARBAGE
CMPB #7, .MSG ;IS IT A ↑G
BNE OUT
TYPE, SCNTG
.CNTLU: COM @RDSW
TYPE, SMSWR
CNVRT, SWREGC
INSTR, SMNEW
PARAM
0
177777
SWREG
.BYTE 0,1
OUT: CLR @RDSW
RTI
SWREGC: 1
.BYTE 6,2
SWREG
SCNTG: .ASCIZ <377>/↑G/
SMSWR: .ASCIZ <377>/SWR= /
SMNEW: .ASCIZ / NEW= /
.EVEN

```


K07

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 88
DZDQAC.P11 16-DEC-76 11:24

GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

4418	016434	020040	000077		
4419	016440	000377			
4420	016442	050377	051127	043040	
4421	016450	044501	042514	027104	
4422	016456	051040	051505	040524	
4423	016464	052122	040440	020124	
4424	016472	042524	052123	000040	
4425	016500	042777	042116	050040	MEPASS: .ASCIZ <377>/END PASS DZDQA /
4426	016506	051501	020123	055104	
4427	016514	050504	020101	000040	
4428	016522	051377	000		MR: .ASCIZ <377>/R/
4429	016525	377	051120	043517	MERR2: .ASCIZ <377>/PROGRAM INDICATES NO DEVICES PRESENT./
4430	016532	040522	020115	047111	
4431	016540	044504	040503	042524	
4432	016546	020123	047516	042040	
4433	016554	053105	041511	051505	
4434	016562	050040	042522	042523	
4435	016570	052116	000056		
4436	016574	044777	051516	043125	MERR3: .ASCIZ <377>/INSUFFICIENT DATA! /
4437	016602	044506	044503	047105	
4438	016610	020124	040504	040524	
4439	016616	000041			
4440	016620	052377	051505	020124	MTSTPC: .ASCIZ <377>/TEST PC-/
4441	016626	041520	000055		
4442	016632	046377	041517	020113	MLOCK: .ASCIZ <377>/LOCK ON SELECTED TEST/
4443	016640	047117	051440	046105	
4444	016646	041505	042524	020104	
4445	016654	042524	052123	000	
4446	016661	103	051123	020072	MCSRX: .ASCIZ /CSR: /
4447	016666	000			
4448	016667	126	041505	020072	MVECX: .ASCIZ /VEC: /
4449	016674	000			
4450	016675	120	051501	042523	MPASSX: .ASCIZ /PASSES: /
4451	016702	035123	000040		
4452	016706	051105	047522	051522	MERRX: .ASCIZ /ERRORS: /
4453	016714	020072	000		
4454	016717	377	052377	051505	MTSTN: .ASCIZ <377><377> /TEST NO: /
4455	016724	020124	047516	020072	
4456	016732	000			
4457	016733	377	042523	020124	MNEW: .ASCIZ <377>/SET SWITCH REG TO DQ11'S DESIRED ACTIVE./
4458	016740	053523	052111	044103	
4459	016746	051040	043505	052040	
4460	016754	020117	050504	030461	
4461	016762	051447	042040	051505	
4462	016770	051111	042105	040440	
4463	016776	052103	053111	027105	
4464	017004	000			
4465	017005	120	035103	000040	MERRPC: .ASCIZ /PC: /
4466	017012	046777	050101	047440	XHEAD: .ASCIZ <377>/MAP OF DQ11 STATUS/<377>
4467	017020	020106	050504	030461	
4468	017026	051440	040524	052524	
4469	017034	177523	000		
4470		017040			.EVEN
4471	017040	000002			XSTATQ: 2
4472	017042	006	003		.BYTE 6,3
4473	017044	001244			TEMP1

L07

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 89
DZDQAC.P11 16-DEC-76 11:24 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

4474	017046	006	002	.BYTE	6,2
4475	017050	001246		TEMP2	
4476				.EVEN	
4477					
4478				;BUFFERS FOR INPUT-OUTPUT	
4479					
4480	017052	000000		INBUF:	0
4481		017114		.=. +40	
4482	017114	000000		TEMP:	0
4483		017156		.=. +40	
4484	017156	000000		MDATA:	0
4485		017220		.=. +40	

```

4486 017220 000002 .MEMCLR: RTI
4487 017222 000002 .MSTCLR: RTI
4488
017224 042101 051104 051505 EMO: ;TABLE OF ERROR MESSAGES AND ERROR DATA POINTERS
017261 120 044522 040515 EM1: .ASCIZ /ADDRESS SELECT ERROR-TIMEOUT/
017316 042522 042503 053111 EM2: .ASCIZ /PRIMARY REG ADDRESSING ERROR/
017356 051124 047101 046523 EM3: .ASCIZ /RECEIVER CONTROL REG DATA ERROR/
017421 105 051122 051117 EM4: .ASCIZ /TRANSMITTER CONTROL REG DATA ERROR/
017446 051777 041505 047117 EM5: .ASCIZ /ERROR REG DATA ERROR/
017503 123 041505 047117 EM6: .ASCIZ (<377>)/SECONDARY REG ADDRESS ERROR/
017534 051377 043505 051511 DH0: .ASCIZ /SECONDARY REG DATA ERROR/
017560 042777 050130 041505 DH1: .ASCIZ (<377>)/REGISTER ADDRESSED/
017621 377 054105 042520 DH2: .ASCIZ (<377>)/EXPECTED RECEIVED REG ADDRESS/
    (<377>)/EXPECTED RECEIVED SEC ADR SEC REG/
    .EVEN
    
```

```

4489 017670 017224 .ERRTAB:EMO
4490 017672 017534 DH0
4491 017674 017750 DTO
4492 017676 017261 EM1
4493 017700 017560 DH1
4494 017702 017756 DT1
4495 017704 017316 EM2
4496 017706 017560 DH1
4497 017710 017756 DT1
4498 017712 017356 EM3
4499 017714 017560 DH1
4500 017716 017756 DT1
4501 017720 017421 EM4
4502 017722 017560 DH1
4503 017724 017756 DT1
4504 017726 000000 0
4505 017730 000000 0
4506 017732 000000 0
4507 017734 017503 EM6
4508 017736 017621 DH2
4509 017740 017774 DT2
4510 017742 017446 EM5
4511 017744 017621 DH2
4512 017746 017774 DT2
4513
4514 017750 000001 DTO: 1
4515 017752 006 000 .BYTE 6,0
4516 017754 001270 SAVRS
4517 017756 000003 DT1: 3
4518 017760 006 004 .BYTE 6,4
4519 017762 001270 SAVRS
4520 017764 006 004 .BYTE 6,4
4521 017766 001266 SAVR4
4522 017770 006 000 .BYTE 6,0
4523 017772 001264 SAVR3
4524 017774 000004 DT2: 4
4525 017776 006 004 .BYTE 6,4
4526 020000 001270 SAVRS
4527 020002 006 004 .BYTE 6,4
4528 020004 001266 SAVR4
4529 020006 006 004 .BYTE 6,4
    
```

N07

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 91
DZDQAC.P11 16-DEC-76 11:24 TABLES FOR ERROR MESSAGES.

4530	020010	001372			DQSEC
4531	020012	002	000	.BYTE	2 0
4532	020014	001264			SAVR3
4533					;DATA TABLE FOR SECONDARY REGISTER ADDRESSING TEST
4534					
4535	020016	000000		DATAB:	0
4536	020020	010421			10421
4537	020022	021042			21042
4538	020024	031463			31463
4539	020026	042104			42104
4540	020030	052525			52525
4541	020032	063146			63146
4542	020034	073567			73567
4543	020036	104210			104210
4544	020040	114631			114631
4545	020042	005212			5212
4546	020044	000000			0
4547	020046	146314			146314
4548	020050	000000			0
4549	020052	000000			0
4550	020054	177777			177777
4551		000001		.END	

F08

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 97
 DZDQAC.P11 16-DEC-76 11:24 CROSS REFERENCE TABLE -- USER SYMBOLS

SAVR1	001260	1079#	4189*	4196										
SAVR2	001262	1080#	4188*	4197										
SAVR3	001264	1081#	4187*	4198	4523	4532								
SAVR4	001266	1082#	4186*	4199	4521	4528								
SAVRS	001270	1083#	4185*	4200	4516	4519	4526							
SAVSP	001272	1084#												
SAVDS =	104406	1119#	4287											
SCOPE =	104400	1107#	1378	1398	1418	1438	1487	1515	1543	1571	1599	1627	1655	1685
		1713	1741	1769	1798	1832	1860	1888	1920	1954	1987	2019	2051	2099
		2136	2169	2191	2225	2258	2291	2324	2357	2390	2423	2456	2488	2520
		2552	2584	2620	2660	2701	2733	2763	2793	2823	2853	2883	2913	2943
		2973	3003	3033	3063	3093	3123	3153	3183	3213	3243	3273	3303	3333
		3363	3393	3423	3453	3483	3519	3549	3579	3609	3639	3669	3699	3729
		3759	3789	3819	3849	3879	3909	3939	3969					
		1109#	2615	2655	2696									
SCOPI =	104401	643#												
SEQ. =	000014	4217*	4238	4241*	4257#									
SPACNT=	015577	586#	1204	1279	1375	1395	1415	1435	4341	4362				
STACK =	001200	1095#	1207*											
STFLG	001311	4185#												
SVDS	015270	1049#	1222*	1227	1231*	1237	1240	1251	1258	1264	1283	1291	4029	4036
SWR	001200	4053	4065	4276	4281	4328	4336	4338	4388					
		956#	1231	1237	4388	4404	4411							
SWREG	000176	4399	4409#											
SWREGC	016402	566#	1251											
SW00 =	000001	565#	1291											
SW01 =	000002	564#												
SW02 =	000004	563#												
SW03 =	000010	562#												
SW04 =	000020	561#												
SW05 =	000040	560#												
SW06 =	000100	559#	4336											
SW08 =	000400	558#	4053											
SW09 =	001000	557#	4338											
SW10 =	002000	556#	4036											
SW11 =	004000	555#	4065	4276										
SW12 =	010000	554#	4281											
SW13 =	020000	553#												
SW14 =	040000	552#												
SW15 =	100000	625#	978											
SYNBIT=	100000	640#												
SYNC. =	000011	4213	4221	4248*	4363*	4364*	4482#							
TEMP	017114	935#	936*	1073#	1244*	1245	1249*	4473						
TEMP1	001244	1074#	1245*	4475										
TEMP2	001246	1075#	4213*	4248										
TEMP3	001250	1076#												
TEMP4	001252	1077#												
TEMPS	001254	1051#	4032	4092	4390									
TKCSR	001204	1052#	4034	4094	4102	4392								
TKDBR	001206	1297	3971#											
TLAST =	014160	1053#	4069	4072	4075	4103	4278							
TPCSR	001210	1054#	4071*	4074*	4077*	4102*	4280*							
TPDBR	001212	4267#												
TRPOK	015614	1063#	1214*	1307*	1366*	1386*	1406*	1426*	1447*	1495*	1523*	1551*	1579*	1607*
TSTNO	001226	1635#	1665*	1693*	1721*	1749*	1778*	1812*	1840*	1868*	1896*	1930*	1963*	1995*
		2027#	2068*	2109*	2149*	2183*	2200*	2233*	2266*	2299*	2332*	2365*	2398*	2431*

TST40	006054	2234	2266#											
TST41	006144	2267	2299#											
TST42	006234	2300	2332#											
TST43	006324	2333	2365#											
TST44	006414	2366	2398#											
TST45	006504	2399	2431#											
TST46	006574	2432	2464#											
TST47	006660	2465	2496#											
TST5	003100	1407	1426#											
TST50	006744	2497	2528#											
TST51	007030	2529	2560#											
TST52	007114	2561	2594#											
TST53	007262	2634#												
TST54	007430	2675#												
TST55	007554	2676	2713#											
TST56	007644	2714	2743#											
TST57	007734	2744	2773#											
TST6	003164	1427	1447#											
TST60	010024	2774	2803#											
TST61	010114	2804	2833#											
TST62	010204	2834	2863#											
TST63	010274	2864	2893#											
TST64	010364	2894	2923#											
TST65	010454	2924	2953#											
TST66	010544	2954	2983#											
TST67	010634	2984	3013#											
TST7	003364	1448	1495#											
TST70	010724	3014	3043#											
TST71	011014	3044	3073#											
TST72	011104	3074	3103#											
TST73	011174	3104	3133#											
TST74	011264	3134	3163#											
TST75	011354	3164	3193#											
TST76	011444	3194	3223#											
TST77	011534	3224	3253#											
TTST	014446	1286#	1287#	1289#	1290#	4030#								
TXBA.P=	000002	632#												
TXBA.S=	000006	636#												
TXMC.P=	000003	633#												
TXMC.S=	000007	637#												
TX.BCC=	000016	645#												
TX.MUX=	000013	642#												
TYPDAT	016046	4302	4320	4323#										
TYPE =	104402	1026	1111#	1219	1243	1253	1260	1285	1303	1328	3980	3982	3986	3990
		3994	4088	4098	4110	4114	4205	4244	4303	4307	4311	4316	4321	4366
		4396	4398	4406										
TYPMSG	015770	4300	4303#											
VECMAP	000056	928#	1025											
MRDCNT	015574	4215#	4246#	4255#										
MRKO.F	016034	4315	4318#											
XBX	015662	4277	4279	4281#										
XB1 =	000020	2193#	2194	2226#	2227	2259#	2260	2292#	2293	2325#	2326	2358#	2359	2391#
		2392	2424#	2425	2457#	2458	2489#	2490	2521#	2522	2553#	2703#	2706	2734#
		2736	2764#	2766	2794#	2796	2824#	2826	2854#	2856	2884#	2886	2914#	2916
		2944#	2946	2974#	2976	3004#	3006	3034#	3036	3064#	3066	3094#	3096	3124#
		3126	3154#	3156	3184#	3186	3214#	3216	3244#	3246	3274#	3276	3304#	3306

		3334#	3336	3364#	3366	3394#	3396	3424#	3490#	3492	3520#	3522	3550#	3552
		3580#	3582	3610#	3612	3640#	3642	3670#	3672	3700#	3702	3730#	3732	3760#
		3762	3790#	3792	3820#	3822	3850#	3852	3880#	3882	3910#	3912	3940#	3942
		3970#												
XCSR	014406	3985	4013#											
XERR	014430	3997	4022#											
XHEAD	017012	1243	4466#											
XPASS	014422	3993	4019#											
XSTATQ	017040	1248	4471#											
XTSTN	016156	4306	4347#											
XVEC	014414	3989	4016#											
SCNTG	016410	4396	4412#											
SE	= 000130	1#	1309	1310#	1367	1368#	1387	1388#	1407	1408#	1427	1428#	1448	1449#
		1496	1497#	1524	1525#	1552	1553#	1580	1581#	1608	1609#	1637#	1666	1667#
		1694	1695#	1722	1723#	1750	1751#	1780#	1813	1814#	1841	1842#	1869	1870#
		1898#	1931	1932#	1964	1965#	1996	1997#	2029#	2069	2070#	2110	2111#	2151#
		2184	2185#	2201	2202#	2234	2235#	2267	2268#	2300	2301#	2333	2334#	2366
		2367#	2399	2400#	2432	2433#	2465	2466#	2497	2498#	2529	2530#	2561	2562#
		2597#	2637#	2676	2678#	2714	2715#	2744	2745#	2774	2775#	2804	2805#	2834
		2835#	2864	2865#	2894	2895#	2924	2925#	2954	2955#	2984	2985#	3014	3015#
		3044	3045#	3074	3075#	3104	3105#	3134	3135#	3164	3165#	3194	3195#	3224
		3225#	3254	3255#	3284	3285#	3314	3315#	3344	3345#	3374	3375#	3404	3405#
		3434	3435#	3465	3500	3501#	3530	3531#	3560	3561#	3590	3591#	3620	3621#
		3650	3651#	3680	3681#	3710	3711#	3740	3741#	3770	3771#	3800	3801#	3830
		3831#	3860	3861#	3890	3891#	3920	3921#	3950	3951#				
		4400	4415#											
SMNEW	016423	4398	4413#											
SMSWR	016414	1#	1305	1310#	1364	1368#	1384	1388#	1404	1408#	1424	1428#	1445	1449#
SN	= 000126	1493	1497#	1521	1525#	1549	1553#	1577	1581#	1605	1609#	1633	1637#	1663
		1667#	1691	1695#	1719	1723#	1747	1751#	1776	1780#	1799	1810	1814#	1838
		1842#	1866	1870#	1894	1898#	1928	1932#	1961	1965#	1993	1997#	2025	2029#
		2052	2066	2070#	2107	2111#	2147	2151#	2170	2181	2185#	2198	2202#	2231
		2235#	2264	2268#	2297	2301#	2330	2334#	2363	2367#	2396	2400#	2429	2433#
		2462	2466#	2494	2498#	2526	2530#	2558	2562#	2585	2592	2597#	2621	2632
		2637#	2661	2673	2678#	2711	2715#	2741	2745#	2771	2775#	2801	2805#	2831
		2835#	2861	2865#	2891	2895#	2921	2925#	2951	2955#	2981	2985#	3011	3015#
		3041	3045#	3071	3075#	3101	3105#	3131	3135#	3161	3165#	3191	3195#	3221
		3225#	3251	3255#	3281	3285#	3311	3315#	3341	3345#	3371	3375#	3401	3405#
		3431	3435#	3461	3465#	3484	3497	3501#	3527	3531#	3557	3561#	3587	3591#
		3617	3621#	3647	3651#	3677	3681#	3707	3711#	3737	3741#	3767	3771#	3797
		3801#	3827	3831#	3857	3861#	3887	3891#	3917	3921#	3947	3951#	3971#	
SN1	= 000107	1799#	2052#	2170#	2621#	2661#	3484#							
SN2	= 000126	1799#	2052#	2170#	2621#	2661#	3484#							
SN3	= 000127	1799#	2052#	2170#	2621#	3484#								
SY	= 000016	1#	1098#	1107	1109#	1111#	1113#	1115#	1117#	1119#	1121#	1123#	1125#	1127#
		1129#	1131#	1133#	1135#									
		650#	651	653	655	657	659	661	663	665	667	669	671	673
		675	677	679	681	683	685	687	689	691	693	695	697	699
		701	703	705	707	709	711	713	715	717	719	721	723	725
		727	729	731	733	735	737	739	741	743	745	747	749	751
		753	755	757	759	761	763	765	767	769	771	773	775	777
		779	781	783	785	787	789	791	793	795	797	799	801	803
		805	807	809	811	813	815	817	819	821	823	825	827	829
		831	833	835	837	839	841	843	845	847	849	851	853	855
		857	859	861	863	865	867	869	871	873	875	877	879	881
		883	885	887	889	891	893	895	897	899	901	903	905	909#

DZDQA MACY11 27(1006) 22-DEC-76 10:57 PAGE 104
DZDQAC.P11 16-DEC-76 11:24 CROSS REFERENCE TABLE -- MACRO NAMES

\$START	18	1195													
\$SYMB0	18	549													
\$TRAPS	18	1098													
\$TRPDE	18	1107	1109	1111	1113	1115	1117	1119	1121	1123	1125	1127	1129	1131	1133
\$TRPSR	18	4259													
\$TSTN	18	1305	1364	1384	1404	1424	1445	1493	1521	1549	1577	1605	1633	1663	1691
	1719	1747	1776	1810	1838	1866	1894	1928	1961	1993	2025	2066	2107	2147	2181
	2198	2231	2264	2297	2330	2363	2396	2429	2462	2494	2526	2558	2592	2632	2673
	2711	2741	2771	2801	2831	2861	2891	2921	2951	2981	3011	3041	3071	3101	3131
	3161	3191	3221	3251	3281	3311	3341	3371	3401	3431	3461	3497	3527	3557	3587
	3617	3647	3677	3707	3737	3767	3797	3827	3857	3887	3917	3947			
\$TYPE	18	4057													
\$VARIA	18	1034													

. ABS. 020056 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

MULE:DZDQAC.BIN, MULE:DZDQAC.SEQ/SOL/CRF=DSKZ:UNIV.P11, DSKZ:DZDQAC.P11
RUN-TIME: 46 63 3 SECONDS
RUN-TIME RATIO: 414/113=3.6
CORE USED: 21K (41 PAGES)