DQ11

BASIC LOGIC TEST PART 2

CZDQBD0

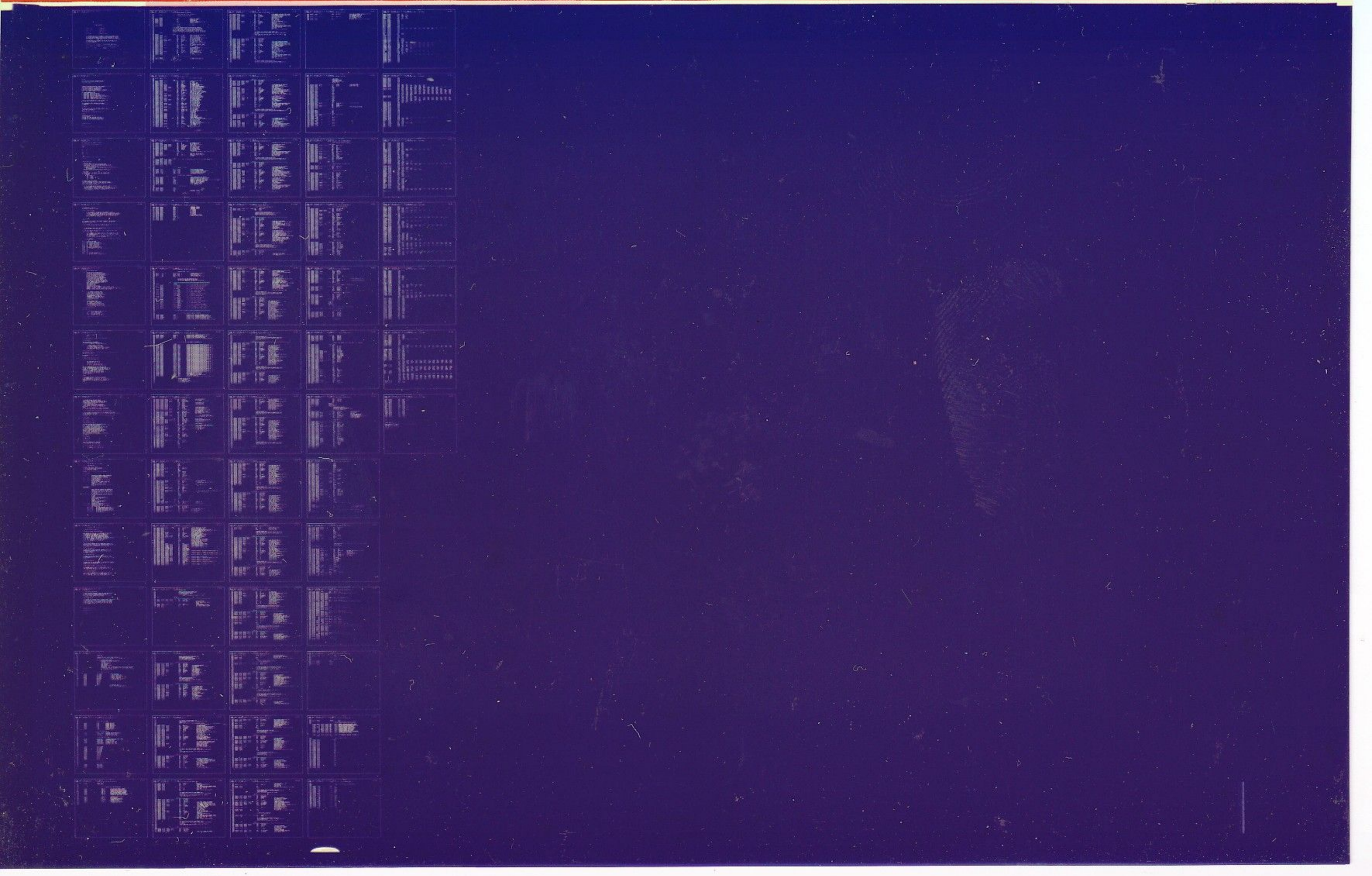
AH-8608D-MC

FICHE 1 OF 1

JAN 1979

digital

MADE IN USA



IDENTIFICATION

PRODUCT CODE: AC-8606D-MC

PRODUCT NAME: CZDQBD0 BLT PRT 2

DATE: JUNE 1978

MAINTAINER: DIAGNOSTIC GROUP

1. ABSTRACT

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

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. CZDQA [REV] BASIS R/W TEST #1
2. CZDQB [REV] BASIC R/W TEST #2
3. CZDQC [REV] BASIC NPR AND INTERUPT TEST
4. CZDQD [REV] RECEIVER TRANSMITTER EXERCISER TEST
5. CZDQE [REV] MISC. RX AND TX TESTS. PLUS BCC TESTS.
6. CZDQF [REV] CHARACTER DETECT TESTS.
7. CZDQH [REV] CHARACTER LENGTH AND INTERRUPT TESTS.

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

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

2. REQUIREMENTS

2.1 EQUIPMENT

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

2.2 STORAGE

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

3. LOADING PROCEEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND

ARE LOADED USING THE ABSOLUTE LOADER.

ABSOLUTE LOADER STARTING ADDRESS *500

MEMORY *
SIZE

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

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

3.1.2 THEN START

4. STARTING PROCEEDURE

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

C.THEN START
THE PROGRAM WILL TYPE MAINDEC NAME AND PROGRAM NAME
IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO
THE FOLLOWING:
'MAP OF DQ11 STATUS''
1400 160010
1402 152300
1404 160020
1406 150310

THE ABOVE IS ONLY AN EXAMPLE!
THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADD.
1400 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE
USER IF AUTO SIZING IS DONE. FOR INFORMATION OF STATUS
TABLE SEE SECTION 8.4 FOR HELP.

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

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

4.1 CONTROL SWITCH SETTINGS

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

CONTROL:

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

1) TYPE CONTROL G <^G>; THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.

2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)

3) AFTER THE ''NEW='' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:

   A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)
   IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.

   B) IF A CONTROL U <^U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

```
SW 15   SET: HALT ON ERROR
SW 14   SET: LOOP ON CURRENT TEST
SW 13   SET: INHIBIT ERROR PRINT OUT
SW 12   SET: INHIBIT TYPE OUT/BELL ON ERROR.
SW 11   SET: INHIBIT ITERATIONS
SW 10   SET: ESCAPE TO NEXT TEST
SW 09   SET: LOOP WITH CURRENT DATA
SW 08   SET: CATCH ERROR AND LOOP ON IT
SW 07   SET: USE PREVIOUS STATUS TABLE. CLR-DO AUTO SIZE.
SW 06   SET:
SW 05   SET:
SW 04   SET:
SW 03   SET:
SW 02   SET: LOCK ON SELECTED TEST
SW 01   SET: RESTART PROGRAM AT SELECTED TEST
SW 00   SET: RESELECT DQ11'S DESIRED ACTIVE.
```

4.1.2 SWITCH REGISTER RESTRICTIONS

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

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

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

SW 09 LOOP ON CURRENT DATA:
THIS SWITCH WILL ONLY WORK IF
CALL ''SCOP1'' IS IN THAT TEST.
THE REASON BEING THAT MOST TESTS
DEAL WITH BLOCKS OF DIFFERENT DATA
TO BE SENT OR RECEIVED ALL AT ONCE
THUS IN BLOCK DATA; ONE PATTERN CANN'T BE SINGLED OUT.

4.1.3 SWITCH REGISTER PRIORITYS

ERROR SWITCHES

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

****HLT (ERROR) ROUTINE SUPPORTS <^G> OPERATION****

SCOPE SWITCHES

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

****SCOPE ROUTINE WILL SUPPORT <^G> OPERATION****

4.2 STARTING ADDRESS

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

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

5. OPERATING PROCEDURE

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

AND PROGRAM WILL BEGIN RUNNING THE
DIAGNOSTIC

5.2 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

1. HALT ON ERROR (VIA SW 15=1)
WHEN EVER AN ERROR OCCURS
2. CLEAR SW 15
3. SET SW 14: (LOOP ON THIS TEST)
4. SET SW 13: (INHIBIT ERROR PRINT OUT)

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

6. ERRORS

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

6.2 ERROR RECOVERY

IF FOR SOME REASON THE DQ11 SHOULD
''HANG THE BUS'' (GAIN CONTROL OF BUS SO THAT
CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT
OR POWER DOWN/UP IS NECESSARY FOR OPERATOR
TO REGAIN CONTROL OF CPU.
IF THIS SHOULD HAPPEN; LOOK IN LOCATION
''TSTNO'' (ADDRESS 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 AC-8606D-MC 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 PEFORMED.
TSTNO	(1226)	CONTAINS THE NUMBER OF THE TEST NOW BEING PEFORMED.
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 SEQUENTIALY. THEY CONTAIN THE CSR,VECTOR AND STATUS CONCERNING THE CONFIGURATION OF EACH DQ11.
DQACTV	(1500)	EACH BIT SET IN THIS LOCATION INDICATES THAT THE ASSOCIATED DQ11 WILL BE TESTED IN TURN. EXAMPLE: (DQACTV) 1500/0000000000011111 MEANS THAT DQ11 NO. 00,01,02,03,04 WILL BE TESTED. EXAMPLE: (DQACTV) 1500/0000000000010001 MEANS THAT DQ11 NO. 00,04 WILL BE TESTED.
DQCSR	(1506)	CONTAINS THE RECEIVER CSR OF THE CURRENT DQ11 UNDER TEST.
DQSTAT	(1510)	CONTAINS THE STATUS OF THE CURRENT DQ11 UNDER TEST.
BIT 15		SET: TWO SYNC CHARS/ONE SYNC CHAR
BIT 14		SET: TEST JUMPER INSTALLED/NOT INSTALLED
BIT 13		SET: 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 APPRIOATE 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? ***NOTE:CZDQE ONLY***

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

8.5.7 ACTIVE ON FIRST NON-SYNC?

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

8.5.8 SET FOR ODD OR EVEN PARITY?

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

8.5.9 FINDING THE VECTOR.

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

9. PROGRAM DESCRIPTION

CONTAINED WITHIN LISTING

10. LISTING

FOLLOWING

```
                                .ENABLE AMA

                                ;CZDQBD0/<377>/DQ11 STATIC LOGIC TEST-PART 2
                                ;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 <^G>.
                                ;       ;STARTING PROCEDURE
                                ;       ;LOAD PROGRAM
                                ;       ;LOAD ADDRESS 000200
                                ;       ;PRESS START
                                ;       ;PROGRAM WILL TYPE ''CZDQBD0/<377>/DQ11 STATIC LOGIC TEST-PART 2''
                                ;       ;PROGRAM WILL TYPE 'R'' TO INDICATE THAT TESTING HAS STARTED
                                ;       ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
                                ;       ;AND THEN RESUME TESTING


                                ;SWITCH REGISTER OPTIONS

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

```
                                ;REGISTER DEFINITIONS
                                
000000                          R0=%0           ;GENERAL REGISTER
000001                          R1=%1           ;GENERAL REGISTER
000002                          R2=%2           ;GENERAL REGISTER
000003                          R3=%3           ;GENERAL REGISTER
000004                          R4=%4           ;GENERAL REGISTER
000005                          R5=%5           ;GENERAL REGISTER
000006                          SP=%6           ;PROCESSOR STACK POINTER
000007                          PC=%7           ;PROGRAM COUNTER
                                
                                ;LOCATION EQUIVALENCIES
                                
177570                          DSWR=   177570  ;HARDWARE SWITCH REGISTER LOC.
177570                          DLIGHTS=177570  ;HARDWARE DISPLAY REGISTER LOC.
177776                          PS=177776       ;PROCESSOR STATUS WORD
001200                          STACK=1200      ;START OF PROCESSOR STACK
                                
                                ;INSTRUCTION DEFINITIONS
                                
005746                          PUSH1SP=5746    ;DECREMENT PROCESSOR STACK 1 WORD
005726                          POP1SP=5726     ;INCREMENT PROCESSOR STACK 1 WORD
010046                          PUSHR0=10046    ;SAVE R0 ON STACK
012600                          POPR0=12600     ;RESTORE R0 FROM STACK
024646                          PUSH2SP=24646   ;DECREMENT STACK TWICE
022626                          POP2SP=22626    ;INCREMENT STACK TWICE
                                .EQUIV  EMT,HLT ;BASIC DEFINITION OF ERROR CALL
                                
100000                          BIT15=100000
040000                          BIT14=40000
020000                          BIT13=20000
010000                          BIT12=10000
004000                          BIT11=4000
002000                          BIT10=2000
001000                          BIT9=1000
000400                          BIT8=400
000200                          BIT7=200
000100                          BIT6=100
000040                          BIT5=40
000020                          BIT4=20
000010                          BIT3=10
000004                          BIT2=4
000002                          BIT1=2
000001                          BIT0=1
                                
                                ;DQ11 OPTIONAL DEFINITIONS
                                
002000                          ABBIT=2000
004000                          ACTBIT=4000
010000                          BABIT=10000
020000                          BBBIT=20000
040000                          JUMBIT=40000
```

```
615          001000                  ODDBIT=1000
616          100000                  SYNBIT=100000
617
618
619                                  ;DQ11 SECONDARY REGISTER DEFINATIONS
620
621          000000                  RXBA.P=0        ;RECEIVER BUS ADDRESS PRIMARY.
622          000001                  RXWC.P=1        ;RECEIVER WORD COUNT PRIMARY.
623          000002                  TXBA.P=2        ;TRANSMITTER BUS ADDRESS PRIMARY.
624          000003                  TXWC.P=3        ;TRANSMITTER BUS ADDRESS PRIMARY.
625          000004                  RXBA.S=4        ;RECEIVER BUS ADDRESS SECONDARY.
626          000005                  RXWC.S=5        ;RECEIVER WORD COUNT SECONDARY.
627          000006                  TXBA.S=6        ;TRANSMITTER BUS ADDRESS SECONDARY.
628          000007                  TXWC.S=7        ;TRANSMITTER WORD COUNT SECONDARY.
629
630          000010                  CHARDT=10       ;CHARACTER DETECT REGISTER.
631          000011                  SYNC.=11        ;SYNC REGISTER.
632          000012                  MISC.=12        ;MISCELLANEOUS REGISTER.
633          000013                  TX.MUX=13       ;TRANSMITTER MUX REGISTER.
634          000014                  SEQ.=14         ;SEQUENCE REGISTER.
635          000015                  RX.BCC=15       ;RECEIVER BCC REGISTER.
636          000016                  TX.BCC=16       ;TRANSMITTER BCC REGISTER.
637          000017                  POLY.=17        ;POLYNOMIAL REGISTER.
638
639
```

```
                                        ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
                000000          .=0
                                        ;STANDARD INTERRUPT VECTORS
                000024          .=24
000024  011056                  .PFAIL                  ;POWER FAIL HANDLER
000026  000340                  340                     ;SERVICE AT LEVEL 7
000030  010526                  .HLT                    ;ERROR HANDLER
000032  000340                  340                     ;SERVICE AT LEVEL 7
000034  010474                  .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                  340                     ;SERVICE AT LEVEL 7
                000046          .=46
000046  007254                  LOGICAL                 ;ACT HOOKS
                000052          .=52
000052  000000                  .WORD 0
                                ;THIS ROUTINE TRIES TO FORCE THE RECEIVER TO INTERUPT
                                ;TO ITS VECTOR WHERE IT WILL PICK UP THE STATUS LOCATION
                                ;FOR ITS NEW PC; AND PICK UP AN IOT INSTRUCTION FOR ITS
                                ;NEW PS. WHEN THE NEW PC IS FETCHED AN IOT INSTRUCTION IS
                                ;EXECUTED, TRAPPING TO LOCATION 20 WHERE A ROUTINE IS EXECUTED
                                ;TO TAKE THE PC FROM THE STACK AND US IT AS THE VECTOR ADDRESS
                000056          .=56
000056                  VECMAP:
000056  010120          1$:     MOV     R1,(R0)+        ;START FILLING THE VECTOR AREA
000060  012721  000004          MOV     #4,(R1)+        ;WITH .+2; IOT (4)
000064  022021                  CMP     (R0)+,(R1)+     ;UPDATE THE POINTERS
000066  020127  001000          CMP     R1,#1000        ;IS ALL FLOATING VECTOR AREA DONE
000072  101771                  BLOS    1$              ;BR IF NOT ALL DONE
000074  012737  000146  000020  MOV     #4$,@#20        ;SET FOR IOT TRAP BY DQ11
000102  013737  001500  001244  MOV     DQACTV,TEMP1    ;GET THE ACTIVE DQ11 S
000110  006037  001244  2$:     ROR     TEMP1           ;ARE YOU ACTIVE.. DQ11
000114  103023                  BCC     5$              ;IF CARRY CLEAR.. NO MORE DQ11S
000116  005037  177776          CLR     PS              ;CLEAR PS
000122  005722                  TST     (R2)+           ;PUT POINTER TO STATUS TABLE
000124  012772  000340  177776  MOV     #340,@-2(R2)    ;TRY AND SET PRI/SEC DONE AND IE
000132  105200                  INCB    R0              ;DELAY........
000134  001376                  BNE     .-2             ;.........DELAY
000136  112712  000300          MOVB    #300,(R2)       ;NO INTERUPT ASSUME 300 FIX IN TEST C
000142  005722          3$:     TST     (R2)+           ;UPDATE POINTERS
000144  000761                  BR      2$              ;GO DO IT AGAIN
000146  051612          4$:     BIS     (SP),(R2)       ;ENTERD  BY IOT TRAP BY DQ11
000150  042712  000007          BIC     #7,(R2)         ;CLEAR UNWANTED BITS
000154  022626                  CMP     (SP)+,(SP)+     ;POP IOT JUNK OFF STACK
000156  012716  000142          MOV     #3$,(SP)        ;SET RETURN PC ON STACK
000162  000002                  RTI                     ;GO HOME.
000164  000207          5$:     RTS     PC              ;ALL SIZING IS DONE

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

                                ;PROGRAM START

                000200          .=200
```

```
696  000200  000137  001512                 JMP     .START          ;GO TO START OF PROGRAM
697
698          000220                 .=220
699  000220  012702  001400         CSRMAP: MOV     #1400,R2        ;CLEAR ALL STATUS TABLE
700  000224  005022                 CLR     (R2)+           ;DO CLEAR
701  000226  022702  001512                 CMP     #1512,R2        ;ALL TABLE DONE
702  000232  001374                 BNE     .-6             ;BR IF MORE TO GO
703  000234  005037  001504                 CLR     DQNUM           ;SET NUMBER OF DQ11S TO 0
704  000240  012702  001400                 MOV     #1400,R2        ;SET TABLE POINTER
705  000244  012701  160000                 MOV     #160000,R1      ;GET FIRST FLOATING ADDRESS
706  000250  012737  000614  000004         MOV     #5$,@#4         ;SET FOR TIME OUT TRAP--NO DEVICE--
707  000256  112761  000012  000005  1$:    MOVB    #12,5(R1)       ;TRY AND SEL MISC REGISTER
708  000264  005061  000006                 CLR     6(R1)           ;TRY AND CLEAR MISC REG
709  000270  012711  010000                 MOV     #10000,(R1)     ;TRY AND SET RX ACTIVE
710  000274  022761  030000  000006         CMP     #30000,6(R1)    ;LOOK FOR SYNC 1 AND SYNC 2
711  000302  001071                 BNE     2$              ;THIS IS NOT A DQ11 IF I BRANCH
712  000304  010122                 MOV     R1,(R2)+        ;NOW THIS IS A DQ11 --STORE CSR
713  000306  052712  100000                 BIS     #SYNBIT,(R2)    ;SET FOR TWO SYNC CHARS
714  000312  005011                 CLR     (R1)            ;CLEAR DQ ACTIVE BIT
715  000314  112761  000010  000005         MOVB    #10,5(R1)       ;SEL CHAR DET REGISTER
716  000322  012761  177777  000006         MOV     #-1,6(R1)       ;WRITE INTO CHAR DET REG
717  000330  005761  000006                 TST     6(R1)           ;WAS THE REGISTER WRITTEN?
718  000334  001402                 BEQ     .+6             ;APPERENTLY NO BB OPTION.
719  000336  052712  020000                 BIS     #BBBIT,(R2)     ;SET FOR BB OPTION
720  000342  112761  000017  000005         MOVB    #17,5(R1)       ;SEL POLYNO. REGISTER
721  000350  012761  177777  000006         MOV     #-1,6(R1)       ;WRITE POLYNO.REGISTER
722  000356  005761  000006                 TST     6(R1)           ;WAS REG WRITTEN??
723  000362  001402                 BEQ     .+6             ;BR IF NO AB OPTION
724  000364  052712  002000                 BIS     #ABBIT,(R2)     ;SET FOR AB OPTION
725  000370  012761  001400  000002         MOV     #1400,2(R1)     ;TRY TO SET .DTR. .RS.
726  000376  032761  001400  000002         BIT     #1400,2(R1)     ;DID ANY OF THEM SET
727  000404  001402                 BEQ     .+6             ;BR IF NO BA OPTION
728  000406  052712  010000                 BIS     #BABIT,(R2)     ;SET FOR BA OPTION
729  000412  032761  030000  000002         BIT     #30000,2(R1)    ;DID .CS. .CO. SET
730  000420  001402                 BEQ     .+6             ;BR IF NO JUMPER
731  000422  052712  040000                 BIS     #JUMBIT,(R2)    ;SET FOR JUMPER
732  000426  052712  004000                 BIS     #ACTBIT,(R2)    ;SET FOR ACTIVE ON FIRST NON-SYNC
733  000432  052712  001000                 BIS     #ODDBIT,(R2)    ;SET FOR ODD VRC.......
734  000436  005722                 TST     (R2)+           ;POP POINTER
735  000440  005011                 CLR     (R1)            ;CLEAR RCSR
736  000442  005061  000002                 CLR     2(R1)           ;CLEAR TCSR
737  000446  005061  000002                 CLR     2(R1)           ;CLEAR AGAIN
738  000452  005061  000004                 CLR     4(R1)           ;CLEAR ERROR REG
739  000456  005061  000006                 CLR     6(R1)           ;CLEAR SEC REG
740  000462  005237  001504                 INC     DQNUM           ;UPDATE NUMBER OF DQ11S
741  000466  062701  000010         2$:     ADD     #10,R1          ;UPDATE CSR POINTER BY 10 (8)
742  000472  022701  164000                 CMP     #164000,R1      ;HAVE ALL FLOATING ADDRESSES BEEN CHECKED??
743  000476  001267                 BNE     1$              ;BR IF NOT ALL DONE
744  000500  005037  001500                 CLR     DQACTV          ;ZERO ACTIVE DQ11S
745  000504  005737  001504                 TST     DQNUM           ;WERE ANY DQ11S FOUND
746  000510  001434                 BEQ     4$              ;HEY BUDDY. NO DQ11S FOUND IN SYSTEM
747  000512  013701  001504                 MOV     DQNUM,R1        ;SAVE NUMBER OF DQ11S
748  000516  010137  001276                 MOV     R1,SAVNUM       ;SAVE NUMBER FOR ACT11
749  000522  000241         3$:     CLC                     ;CLEAR CARRY
750  000524  006137  001500                 ROL     DQACTV          ;       ACTIVE ADDRESS
751  000530  005237  001500                 INC     DQACTV          ;SET BIT 0
```

```
000534  005301                          DEC     R1              ;DEC NUMBER OF DQ11S
000536  001371                          BNE     3$              ;BR IF MORE TO GO
000540  012737  000006  000004          MOV     #6,@#4          ;RESET TIME OUT VECTOR
000546  013737  001500  001502          MOV     DQACTV,SAVACT   ;SAVE ACTIVE
000554  012737  000340  000022          MOV     #340,@#22       ;SET IOT TRAP PRIO: TO 7
000562  012702  001400                  MOV     #1400,R2        ;SET TABLE POINTER
000566  012700  000300                  MOV     #300,R0         ;SET VECTOR START
000572  012701  000302                  MOV     #302,R1         ;SET VECTOR+2 START
000576  000137  000056                  JMP     VECMAP          ;GO FIND THE VECTORS
000602  104402                  4$:     TYPE                    ;TYPE MESSAGE
000604  011417                          MERR2                   ;I DIDN'T FIND ANY DQ11S. DON'T USE AUTO SIZE.
000606  005000                          CLR     R0              ;
000610  000000                          HALT                    ;HOW CAN I TEST NO DQ11S
000612  000776                          BR      .-2             ;DON'T LET OPR HIT CONT. SW
000614  012716  000466          5$:     MOV     #2$,(SP)        ;ENTERED BY TIME OUT TRAP
000620  000002                          RTI                     ;GO HOME.

        001000                  .=1000
001000  005377  055103  050504  MTITLE: .ASCIZ  <377><12>/CZDQBD0/<377>/DQ11 STATIC LOGIC TEST-PART 2/<377>
001006  042102  177460  050504
001014  030461  051440  040524
001022  044524  020103  047514
001030  044507  020103  042524
001036  052123  050055  051101
001044  020124  177462     000

        001200                  .=1200
                                ;INDIRECT POINTERS

001200  177570                  SWR:    177570                  ;SWITCH REGISTER POINTER
001202  177570                  LIGHTS: 177570                  ;DISPLAY REGISTER POINTER
001204  177560                  TKCSR:  177560                  ;TELETYPE KEYBOARD CONTROL REGISTER
001206  177562                  TKDBR:  177562                  ;TELETYPE KEYBOARD DATA BUFFER
001210  177564                  TPCSR:  177564                  ;TELEPRINTER CONTROL REGISTER
001212  177566                  TPDBR:  177566                  ;TELEPRINTER DATA BUFFER

                                ;PROGRAM CONTROL PARAMETERS

001214  000000                  RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
001216  000000                  NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
001220  000000                  LOCK:   0                       ;ADDRESS FOR LOCK ON CURRENT DATA
001222  000003                  ICOUNT: 3                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
001224  000000                  LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
001226  000000                  TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
001230  000000                  PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
001232  000000                  ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
001234  000000                  LSTERR: 0                       ;PC OF LAST ERROR CALL

                                ;PROGRAM VARIABLES

001236  000000                  CHAR1:  0
001240  000000                  CHAR2:  0
001242  000000                  CHAR3:  0
001244  000000                  TEMP1:  0                       ;TEMPORARY STORAGE
001246  000000                  TEMP2:  0                       ;TEMPORARY STORAGE
```

```
  808  001250  000000                TEMP3:  0                       ;TEMPORARY STORAGE
  809  001252  000000                TEMP4:  0                       ;TEMPORARY STORAGE
  810  001254  000000                TEMP5:  0                       ;TEMPORARY STORAGE
  811  001256  000000                SAVR0:  0                       ;R0 STORAGE
  812  001260  000000                SAVR1:  0                       ;R1 STORAGE
  813  001262  000000                SAVR2:  0                       ;R2 STORAGE
  814  001264  000000                SAVR3:  0                       ;R3 STORAGE
  815  001266  000000                SAVR4:  0                       ;R4 STORAGE
  816  001270  000000                SAVR5:  0                       ;R5 STORAGE
  817  001272  000000                SAVSP:  0                       ;STACK POINTER STORAGE
  818  001274  000000                SAVPC:  0                       ;PROGRAM COUNTER STORAGE
  819  001276  000000                SAVNUM: 0
  820  001300  000001                CREAM:  .BLKW   1
  821  001302  000000                RUNFLG: 0
  822  001304  000000                RUN:    0
  823  001306  000000                RUNCNT: 0
```

```
                                        ;PROGRAM CONTROL FLAGS

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

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

                        ;*****************************************************************
                        ;*****************************************************************
001314                  .TRPTAB:
        104400          SCOPE=TRAP+0            ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001314  007330                  .SCOPE
        104401          SCOP1=TRAP+1            ;CALL TO LOOP ON CURRENT DATA HANDLER
001316  007442                  .SCOP1
        104402          TYPE=TRAP+2             ;CALL TO TELETYPE OUTPUT ROUTINE
001320  007462                  .TYPE
        104403          INSTR=TRAP+3            ;CALL TO ASCII STRING INPUT ROUTINE
001322  007570                  .INSTR
        104404          INSTER=TRAP+4           ;CALL TO INPUT ERROR HANDLER
001324  007706                  .INSTER
        104405          PARAM=TRAP+5            ;CALL TO NUMERICAL DATA INPUT ROUTINE
001326  007740                  .PARAM
        104406          SAV05=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
001330  010154                  .SAV05
        104407          RES05=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
001332  010214                  .RES05
        104410          CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
001334  010246                  .CONVRT
        104411          CNVRT=TRAP+11           ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
001336  010252                  .CNVRT
        104412          MSTCLR=TRAP+12          ;CALL TO ISSUE MASTER CLEAR
001340  012114                  .MSTCLR
        104413          MEMCLR=TRAP+13          ;CALL TO CLEAR ALL SCRATCH PAD MEMORIES
001342  012112                  .MEMCLR
        104414          CKSWR=TRAP+14           ;CALL TO ALLOW SWREG TO BE LOADED FROM TTY
001344  011154                  .CKSWR
        104415          CNTLU=TRAP+15           ;CALL TO ALLOW LOADING OF SWREG FROM TTY
001346  011230                  .CNTLU

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

                                        ;DQ11 VECTOR AND REGISTER INDIRECT POINTERS

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

```
880  001364  000000                DQTCSR: 0               ;POINTER TO DQ11 TRANSMITTER CONTROL REGISTER
881  001366  000000                DQERR:  0               ;POINTER TO DQ11 ERROR REGISTER
882  001370  000000                DQREG:  0               ;POINTER TO HIGH BYTE OF ERROR REGISTER
883  001372  000000                DQSEC:  0               ;POINTER TO DQ11 SECONDARY REGISTER
884  001374  000000                DQSECH: 0               ;POINTER TO HIGH BYTE OF DQ11 SECONDARY REGISTER
885
886
887
888                                      ;DQ11 STATUS TABLE AND ADDRESS ASSIGNMENTS
889
890           001400                .=1400
891  001400  000001                DQCR00: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 00
892  001402  000001                DQST00: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 00
893  001404  000001                DQCR01: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 01
894  001406  000001                DQST01: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 01
895  001410  000001                DQCR02: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 02
896  001412  000001                DQST02: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 02
897  001414  000001                DQCR03: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 03
898  001416  000001                DQST03: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 03
899  001420  000001                DQCR04: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 04
900  001422  000001                DQST04: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 04
901  001424  000001                DQCR05: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 05
902  001426  000001                DQST05: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 05
903  001430  000001                DQCR06: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 06
904  001432  000001                DQST06: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 06
905  001434  000001                DQCR07: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 07
906  001436  000001                DQST07: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 07
907  001440  000001                DQCR10: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 10
908  001442  000001                DQST10: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 10
909  001444  000001                DQCR11: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 11
910  001446  000001                DQST11: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 11
911  001450  000001                DQCR12: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 12
912  001452  000001                DQST12: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 12
913  001454  000001                DQCR13: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 13
914  001456  000001                DQST13: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 13
915  001460  000001                DQCR14: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 14
916  001462  000001                DQST14: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 14
917  001464  000001                DQCR15: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 15
918  001466  000001                DQST15: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 15
919  001470  000001                DQCR16: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 16
920  001472  000001                DQST16: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 16
921  001474  000001                DQCR17: .BLKW   1       ;CONTROL STATUS REGISTER FOR DEVICE NO: 17
922  001476  000001                DQST17: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 17
923  001500  000001                DQACTV: .BLKW   1       ;HOLD ACTIVE BITS FOR TESTING
924  001502  000001                SAVACT: .BLKW   1       ;SAVE NUMBER OF ACTIVE DQ11S
925  001504  000001                DQNUM:  .BLKW   1       ;OCTAL NUMBER OF TOTAL NUMBER OD DQ11S
926  001506  000001                DQCSR:  .BLKW   1       ;CSR OF DQ11 UNDER TEST
927  001510  000001                DQSTAT: .BLKW   1       ;VECTOR AND CONFIGURATION STATUS OF DQ11 UNDER TEST
928
929                                      ;PROGRAM INITIALIZATION
930                                      ;LOCK OUT INTERRUPTS
931                                      ;SET UP PROCESSOR STACK
932                                      ;SET UP POWER FAIL VECTOR
933                                      ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
934                                      ;TYPE TITLE MESSAGE
935
```

```
936  001512  012737  000340  177776  .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
937  001520  012706  001200                 MOV     #STACK,SP               ;SET UP STACK
938  001524  012737  011056  000024         MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
939  001532  013737  001504  001276         MOV     DQNUM,SAVNUM
940  001540  105037  001311                 CLRB    STFLG                   ;CLEAR START FLAG
941  001544  005037  001230                 CLR     PASCNT                  ;CLEAR PASS COUNT
942  001550  105037  001312                 CLRB    ERRFLG                  ;CLEAR ERROR FLAG
943  001554  005037  001302                 CLR     RUNFLG
944  001560  012737  001400  001300         MOV     #1400,CREAM
945  001566  005037  001232                 CLR     ERRCNT                  ;CLEAR ERROR COUNT
946  001572  005037  001234                 CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
947  001576  012737  000001  001226         MOV     #1,TSTNO                ;SET UP FOR TEST 1
948  001604  012737  001512  001214         MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
949                                                                          ;TESTING STARTS
950  001612  012737  177570  001200         MOV     #DSWR,SWR               ;MOV HARDWARE SWR TO SWR
951  001620  012737  177570  001202         MOV     #DLIGHTS,LIGHTS         ;MOV DISPLAY LIGHTS TO LIGHTS
952  001626  013746  000006                 MOV     @#6,-(SP)               ;SAVE VECTORS
953  001632  013746  000004                 MOV     @#4,-(SP)
954  001636  012737  001656  000004         MOV     #64$,@#4                        ;SET UP FOR TIMEOUT
955  001644  022777  177777  177326         CMP     #-1,@SWR                ;REFERENCE HARDWARE SWITCH REGISTER
956  001652  001402                         BEQ     65$
957  001654  000407                         BR      66$
958  001656  022626                 64$:    CMP     (SP)+,(SP)+             ;ADJUST STACK
959  001660  012737  000176  001200 65$:    MOV     #SWREG,SWR              ;POINT TO SOFTWARE SWITCH REG
960  001666  012737  000174  001202         MOV     #DISPREG,LIGHTS         ;POINT TO SOFT DISPLAY REG
961  001674  012637  000004         66$:    MOV     (SP)+,@#4               ;RESTORE VECTORS
962  001700  012637  000006                 MOV     (SP)+,@#6
963  001704  005737  000042                 TST     @#42                    ;UNDER MONITOR
964  001710  001014                         BNE     67$
965                                 ;;******THE NEXT 4 LINES OF CODE MOVED TO SOLVE PR#2757 (JUNE 78)******
966  001712  105737  001310                 TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED?
967  001716  001035                         BNE     12$                     ;IF YES, BR
968  001720  104402  001000                 TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
969  001724  105137  001310                 COMB    INIFLG          ;IF NOT SET FLAG AND INIT
970  001730  022737  000176  001200         CMP     #SWREG,SWR              ;IS SWREG USED
971  001736  001001                         BNE     67$
972  001740  104415                         CNTLU
973  001742  105777  177232         67$:    TSTB    @SWR
974  001746  100402                         BMI     .+6
975  001750  004737  000220                 JSR     PC,CSRMAP
976  001754  104402  011704                 TYPE    ,XHEAD
977  001760  012737  001400  001244         MOV     #1400,TEMP1
978  001766  017737  177252  001246         MOV     @TEMP1,TEMP2
979  001774  001406                         BEQ     .+16
980  001776  104410                         CONVRT
981  002000  011732                         XSTATQ
982  002002  062737  000002  001244         ADD     #2,TEMP1
983  002010  000766                         BR      .-22
984  002012  032777  000001  177160 12$:    BIT     #SW00,@SWR
985  002020  001424                         BEQ     1$
986  002022  104402                         TYPE
987  002024  011625                         MNEW
988  002026  005000                         CLR     R0
989  002030  000000                         HALT
990  002032  104414                         CKSWR
991  002034  027737  177140  001502         CMP     @SWR,SAVACT.
```

```
  992  002042  101404                          BLOS    11$
  993  002044  104402                          TYPE
  994  002046  011466                          MERR3
  995  002050  000000                          HALT
  996  002052  000776                          BR      .-2
  997  002054  017737  177120  001500  11$:    MOV     @SWR,DQACTV
  998  002062  013700  001500                  MOV     DQACTV,R0
  999  002066  000000                          HALT
 1000  002070  104414                          CKSWR
 1001  002072  012700  000300          1$:     MOV     #300,R0
 1002  002076  012701  000302                  MOV     #302,R1
 1003  002102  010120                  2$:     MOV     R1,(R0)+
 1004  002104  005021                          CLR     (R1)+
 1005  002106  022021                          CMP     (R0)+,(R1)+
 1006  002110  022700  001000                  CMP     #1000,R0
 1007  002114  001372                          BNE     2$
 1008
 1009                                          ;TEST START AND RESTART
 1010
 1011  002116  012737  000340  177776  .BEGIN: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
 1012  002124  012706  001200                  MOV     #STACK,SP               ;SET UP STACK
 1013  002130  005737  000042                  TST     @#42                    ;IS PROGRAM UNDER MONITOR CONTROL
 1014  002134  001040                          BNE     3$
 1015  002136  104414                          CKSWR                           ;CHECK FOR <^G>
 1016  002140  032777  000004  177032          BIT     #BIT2,@SWR              ;CHECK FOR LOCK ON TEST
 1017  002146  001411                          BEQ     1$
 1018  002150  104402  011524                  TYPE    ,MLOCK
 1019  002154  012737  000240  007340          MOV     #NOP,TTST
 1020  002162  012737  000240  007342          MOV     #NOP,TTST+2             ;SET UP TO LOCK
 1021  002170  000406                          BR      2$
 1022  002172  013737  007436  007340  1$:     MOV     BRW,TTST
 1023  002200  013737  007440  007342          MOV     BRX,TTST+2              ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
 1024  002206  032777  000002  176764  2$:     BIT     #SW01,@SWR              ;IF SW01=1, GET STARTING PC
 1025  002214  001410                          BEQ     3$
 1026  002216  104403                          INSTR
 1027  002220  011512                          MTSTPC
 1028  002222  104405                          PARAM
 1029  002224  002254                          TST1
 1030  002226  007046                          TLAST
 1031  002230  001214                          #RETURN
 1032  002232     001                  .BYTE   1
 1033  002233     001                  .BYTE   1
 1034  002234  000403                          BR      4$
 1035  002236  012737  002254  001214  3$:     MOV     #TST1,RETURN            ;START AT TEST 1
 1036  002244  104402  011414          4$:     TYPE    ,MR                     ;TYPE R
 1037  002250  000177  176740                  JMP     @RETURN                 ;START TESTING
 1038                                  ;  TEST 1
 1039                                  ;*****************************************************************************
 1040  002254  012737  000001  001226  TST1:   MOV     #1,TSTNO
 1041  002262  012737  002644  001214          MOV     #TST2,RETURN
 1042  002270  012737  002644  001216          MOV     #TST2,NEXT
 1043  002276  105737  001302                  TSTB    RUNFLG          ;IS THIS MY FIRST TIME HERE?
 1044  002302  001010                          BNE     1$              ;BR IF FLAG IS SET
 1045  002304  012737  000001  001304          MOV     #BIT0,RUN       ;SET RUN POINTER.
 1046  002312  012737  000020  001306          MOV     #16.,RUNCNT     ;SET FOR MAX OF 16 DQ11'S PER SYSTEM
 1047  002320  105137  001302                  COMB    RUNFLG          ;SET RUN FLAG
```

```
1048  002324  033737  001304  001500  1$:     BIT     RUN,DQACTV      ;FIND AN ACTIVE DQ11 TO TEST.
1049  002332  001032                          BNE     3$              ;BR IF I FOUND ONE TO TEST.
1050  002334  005737  001500                  TST     DQACTV          ;FIND OUT IF THERE ARE NO DQ11 ACTIVE.
1051  002340  001423                          BEQ     2$              ;BR TO FATAL ERROR. WHY AM I HERE IF NO ACTIVE DQ11'S???
1052  002342  000257                          CCC                     ;CLEAR ALL THE CONDITION CODES OF CPU
1053  002344  006137  001304                  ROL     RUN             ;UPDATE RUN POINTER
1054  002350  062737  000004  001300          ADD     #4,CREAM        ;UPDATE ADDRESS POINTER.
1055  002356  005337  001306                  DEC     RUNCNT          ;DEC NUMBER OF TIMES I LOOKED AT ACTIVE.
1056  002362  001360                          BNE     1$              ;BR AND KEEP LOOKING.
1057  002364  012737  000020  001306          MOV     #16.,RUNCNT     ;START RESTORING MY POINTERS.
1058  002372  012737  001400  001300          MOV     #1400,CREAM     ;RESTORE  ADDRESS POINTER
1059  002400  012737  000001  001304          MOV     #1,RUN          ;RESTORE RUN POINTER.
1060  002406  000746                          BR      1$              ;KEEP ON TESTING.
1061  002410  104402                  2$:     TYPE                    ;ALLERT OPERATOR OF FATAL ERROR
1062  002412  011417                          MERR2                   ;NO DQ11 ACTIVE. WHY AM I HERE???
1063  002414  000000                          HALT                    ;YOU MUST RELOAD DQ11 DIAGNOSTIC!!
1064  002416  000776                          BR      .-2             ;STICK HERE ON CONT.
1065  002420  000257                  3$:     CCC                     ;CLEAR CPU COND. CODES
1066  002422  006137  001304                  ROL     RUN             ;UPDATE RUN. ACTIVE DQ11 FOUND.
1067  002426  017737  176646  001506          MOV     @CREAM,DQCSR    ;PLACE ADDRESSS OF DQ11 AT DQCSR
1068  002434  062737  000002  001300          ADD     #2,CREAM        ;UPDATE ADDRESS POINTER
1069  002442  017737  176632  001510          MOV     @CREAM,DQSTAT   ;PLACE STATUS OF DQ11 AT DQSTAT
1070  002450  062737  000002  001300          ADD     #2,CREAM        ;UPDATE ADDRESS POINTER
1071  002456  013737  001506  001360          MOV     DQCSR,DQRCSR
1072  002464  013737  001510  001350          MOV     DQSTAT,DQRVEC
1073  002472  042737  177007  001350          BIC     #177007,DQRVEC
1074  002500  013737  001350  001352          MOV     DQRVEC,DQRLVL   ;GENERATE ADDRESS OF RECEIVER INTERRUPT SERVICE PS
1075  002506  062737  000002  001352          ADD     #2,DQRLVL
1076  002514  013737  001352  001354          MOV     DQRLVL,DQTVEC   ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT VECTOR
1077  002522  062737  000002  001354          ADD     #2,DQTVEC
1078  002530  013737  001354  001356          MOV     DQTVEC,DQTLVL   ;GENERATE ADDRESS  OF TRANSMITTER INTERRUPT SERVICE PS
1079  002536  062737  000002  001356          ADD     #2,DQTLVL
1080  002544  013737  001360  001362          MOV     DQRCSR,DQRCSH
1081  002552  005237  001362                  INC     DQRCSH          ;GENERATE ADDRESS OF HIGH BYTE
1082  002556  013737  001360  001364          MOV     DQRCSR,DQTCSR   ;GENERATE ADDRESS OF TRANSMITTER CONTROL REGISTER
1083  002564  062737  000002  001364          ADD     #2,DQTCSR
1084  002572  013737  001364  001366          MOV     DQTCSR,DQERR    ;GENERATE ADDRESS OF ERROR REGISTER
1085  002600  062737  000002  001366          ADD     #2,DQERR
1086  002606  013737  001366  001370          MOV     DQERR,DQREG     ;GENERATE ADDRESS OF HIGH BYTE OF ERROR REGISTER
1087  002614  005237  001370                  INC     DQREG
1088  002620  013737  001370  001372          MOV     DQREG,DQSEC     ;GENERATE ADDRESS OF SECONDARY REGISTER
1089  002626  005237  001372                  INC     DQSEC
1090  002632  013737  001372  001374          MOV     DQSEC,DQSECH    ;GENERATE ADDRESS OF HIGH BYTE
1091  002640  005237  001374                  INC     DQSECH
```

```
1092                                    ;MEMORY EXTENSION WRITE ENABLE TEST
1093                                    ;SET MEMORY EXTENSION WRITE ENABLE
1094                                    ;ADDRESS BUS ADDRESS MEMORY
1095                                    ;EXPECTED RESULTS
1096                                    ;       MEMORY EXTENSION WRITE ENABLE IS CLEARED
1097
1098  002644                            OPT2X:
1099                                    ;  TEST 2
1100                                    ;*************************************************************************
1101  002644  012737  000002  001226    TST2:   MOV     #2,TSTNO
1102  002652  012737  002706  001216            MOV     #TST3,NEXT
1103  002660  112777  000020  176502            MOVB    #BIT4,@DQREG            ;SET WRITE ENABLE,
1104                                                                            ;SELECT RECEVIER BUS ADDRESS
1105                                                                            ;MEMORY
1106  002666  105077  176500                   CLRB    @DQSEC                  ;CLEAR MEMORY
1107  002672  132777  000020  176470            BITB    #BIT4,@DQREG            ;WAS WRITE ENABLE CLEARED
1108  002700  001401                           BEQ     1$                      ;BR IF WRITE ENABLE CLEARED
1109  002702  104012                           HLT     12                      ;WRITE ENABLE NOT CLEARED
1110  002704  104400                    1$:     SCOPE
```

```
1111
1112                                    ;MEMORY EXTENTION BITS READ/WRITE TEST
1113                                    ;READ MEMORY EXTENSION BITS
1114                                    ;WITH WRITE ENABLE BITS CLEARED, ATTEMPT
1115                                    ;TO CHANGE MEMORY EXTENSION BITS
1116                                    ;VERITY THAT NO CHANGE OCCURS
1117
1118                                ;  TEST 3
1119                                ;*************************************************************************
1120  002706  012737  000003  001226 TST3:   MOV     #3,TSTNO
1121  002714  012737  002774  001216         MOV     #TST4,NEXT
1122  002722  013703  001370                MOV     DQREG,R3        ;SET FOR ERROR PRINTOUT
1123  002726  142777  000037  176434        BICB    #37,@DQREG      ;CLEAR MEM EXT AND WRITE ENABLE
1124  002734  117705  176430                MOVB    @DQREG,R5       ;SAVE REGISTER
1125  002740  010500                        MOV     R5,R0           ;STORE REG
1126  002742  105100                        COMB    R0              ;CHANE CONTENTS
1127  002744  142700  000237                BICB    #237,R0         ;CLEAR UNWANTED BITS
1128  002750  110077  176414                MOVB    R0,@DQREG       ;WRITE REGISTER
1129  002754  117704  176410                MOVB    @DQREG,R4       ;READ REGISTER
1130  002760  142704  000037                BICB    #37,R4          ;CLEAR UNWANTED BITS
1131  002764  020504                        CMP     R5,R4           ;DID IT CHANGE
1132  002766  001401                        BEQ     1$              ;BR IF NO CHANGE OCCURED
1133  002770  104000                        HLT     0               ;REPORT ERROR
1134  002772  104400                1$:     SCOPE                   ;SCOPE THIS TEST
1135
1136                                    ;MEMORY EXTENSION READ/WRITE TEST
1137                                    ;READ MEMORY EXTENSION BITS
1138                                    ;WITH WRITE ENABLE SET, ATTEMPT TO CHANGE
1139                                    ;MEMORY EXTENSION BITS
1140                                    ;VERIFY THAT MEMORY EXTENSION BITS WERE CHANGED
1141
1142                                ;  TEST 4
1143                                ;*************************************************************************
1144  002774  012737  000004  001226 TST4:   MOV     #4,TSTNO
1145  003002  012737  003222  001216         MOV     #TST5,NEXT
1146  003010  013703  001370                MOV     DQREG,R3        ;SET FOR ERROR REPORT
1147  003014  142777  000037  176346        BICB    #37,@DQREG      ;CLEAR UNWANTED BITS.
1148  003022  117700  176342                MOVB    @DQREG,R0       ;READ DQREG FOR DATA
1149  003026  105100                        COMB    R0              ;DO A COMPLETE CHANGE OF DATA
1150  003030  142700  000237                BICB    #237,R0         ;CLEAR WRITE ENABLE AND GARBAGE
1151  003034  010005                        MOV     R0,R5           ;STORE DATA.
1152  003036  152705  000020                BISB    #BIT4,R5        ;SET WRITE ENABLE
1153  003042  110577  176322                MOVB    R5,@DQREG       ;WRITE DQREG WITH NEW DATA
1154  003046  005077  176320                CLR     @DQSEC          ;REFERANCE SEL 6 (DQSEC)
1155  003052  117704  176312                MOVB    @DQREG,R4       ;READ DQREG
1156  003056  042704  000037                BIC     #37,R4          ;CLEAR UNWANTED BITS.
1157  003062  042705  000020                BIC     #20,R5          ;CLEAR WRITE ENABLE
1158  003066  020504                        CMP     R5,R4           ;ARE THEY EQUAL?
1159  003070  001401                        BEQ     1$              ;BR IF GOOD
1160  003072  104000                        HLT     0               ;MEMORY EXTENSION READ/WRITE ERROR
1161  003074  104400                1$:     SCOPE                   ;SCOPE THE TEST.
```

```
1162
1163                                        ;BUS ADDRESS EXTENSION MEMORY TEST
1164                                        ;LOAD EACH BUS ADDRESS MEMORY LOCATION WITH A DIFFERENT
1165                                        ;NUMBER (0-3)
1166                                        ;VERIFY THAT EACH LOCATION WAS ADDRESSED PROPERLY
1167
1168  003076  005003                        CLR     R3                      ;FIRST MEMORY ADDRESS =0
1169  003100  052701  000020                BIS     #BIT4,R1                ;SET WRITE ENABLE
1170  003104  112777  000020  176256        MOVB    #BIT4,@DQREG            ;SET MEMORY EXTENSION WRITE ENABLE
1171  003112  150377  176252                BISB    R3,@DQREG               ;SELECT LOCATION TO BE TESTED
1172  003116  110177  176246                MOVB    R1,@DQREG               ;LOAD DATA INTO MEMORY EXTENSION BITS
1173  003122  005077  176244                CLR     @DQSEC                  ;REFERENCE SEL 6 (DQSEC)
1174  003126  062703  000002                ADD     #2,R3                   ;NEXT LOCATION TO BE TESTED
1175  003132  062705  000040                ADD     #40,R5                  ;DATA TO BE LOADED
1176  003136  005300                        DEC     R0                      ;CONTINUE IF ALL LOCATIONS NOT LOADED
1177  003140  001355                        BNE     1$                      ;BR IF MORE TO GO
1178  003142  005005                        CLR     R5                      ;INITIAL DATA = 0
1179  003144  012700  000004                MOV     #4,R0                   ;4 LOCATIONS TO BE TESTED
1180  003150  005003                        CLR     R3                      ;FIRST LOCATION
1181  003152  142777  000037  176210  2$:   BICB    #37,@DQREG              ;REINITIALIZE  ADDRESS
1182  003160  150377  176204                BISB    R3,@DQREG               ;RESELECT SECONDARY ADDRESS.
1183  003164  117704  176200                MOVB    @DQREG,R4               ;READ MEMORY EXTENSION BITS
1184  003170  142704  000037                BICB    #37,R4                  ;CLEAR UNWANTED BITS
1185  003174  020504                        CMP     R5,R4                   ;ARE EXPECTED AND RECEIVED DATA THE SAME
1186  003176  001401                        BEQ     3$
1187  003200  104001                        HLT     1                       ;BUS ADDRESS EXTENSION MEMORY ERROR
1188  003202  104401                3$:     SCOP1                           ;CHECK FOR LOOP ON CUREENT ADDRESS
1189  003204  062703  000002                ADD     #2,R3                   ;UPDATE SELECT ADDRESS
1190  003210  062705  000040                ADD     #40,R5                  ;UPDATE EXPECTED ATA
1191  003214  005300                        DEC     R0                      ;CONTINUE IF NOT DONE
1192  003216  001355                        BNE     2$                      ;BR IF MORE TO CHECK.
1193  003220  104400                4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1194
1195                                        ;BUS ADDRESS AND CHARACTER COUNT MEMORY TEST
1196                                        ;SET EACH LOCATION IN BUS ADDRESS AND CHARACTER COUNT MEMORY
1197                                        ;TO 0
1198                                        ;VERIFY THAT EACH LOCATION IN BUS ADDRESS AND CHARACTER
1199                                        ;COUNT MEMORY WAS SET TO 0
1200
1201                                        ;  TEST 5
1202                                        ;*************************************************************************
1203  003222  012737  000005  001226  TST5:  MOV     #5,TSTNO
1204  003230  012737  003336  001216        MOV     #TST6,NEXT
1205  003236  012737  003306  001220        MOV     #2$,LOCK
1206  003244  012705  000000                MOV     #0,R5           ;DATA TO BE LOADED INTO MEMORY
1207  003250  013703  001372                MOV     DQSEC,R3                ;ADDRESS OF SECONDARY REGISTER
1208  003254  005002                        CLR     R2                      ;FIRST MEMORY LOCATION TO BE TESTED
1209  003256  012700  000010                MOV     #10,R0                  ;# OF MEMORY LOCATIONS TO BE TESTED
1210  003262  110277  176102          1$:   MOVB    R2,@DQREG               ;SELECT MEMORY LOCATION TO BE TESTED
1211  003266  010577  176100                MOV     R5,@DQSEC               ;LOAD DATA INTO MEMORY
1212  003272  005202                        INC     R2                      ;GO TO NEXT LOCATION
1213  003274  005300                        DEC     R0                      ;MORE TO GO??
1214  003276  001371                        BNE     1$                      ;BR IF NOT DONE
1215  003300  005002                        CLR     R2                      ;SET UP TO TEST
1216  003302  012700  000010                MOV     #10,R0                  ;10 (OCTAL) LOCATIONS
1217  003306  110277  176056          2$:   MOVB    R2,@DQREG               ;SELECT LOCATION TO BE TESTED
```

```
1218  003312  017704  176054          MOV     @DQSEC,R4               ;READ DATA
1219  003316  020504                  CMP     R5,R4                   ;COMPARE TO EXPECTED DATA
1220  003320  001401                  BEQ     3$                      ;BR IF GOOD
1221  003322  104002                  HLT     2                       ;CHAR COUNT-BUS ADDRESS MEMORY ERROR
1222  003324  104401          3$:     SCOP1                           ;CHECK FOR FREEZE ON CURRENT LOCATION
1223  003326  005202                  INC     R2                      ;GO TO NEXT LOCATION
1224  003330  005300                  DEC     R0                      ;DONE YET?
1225  003332  001365                  BNE     2$                      ;BR IF FURTHER CHECKING NEEDED
1226  003334  104400          4$:     SCOPE
1227
1228                                  ;BUS ADDRESS AND CHARACTER COUNT MEMORY TEST
1229                                  ;SET EACH LOCATION IN BUS ADDRESS AND CHARACTER COUNT MEMORY
1230                                  ;TO 177777
1231                                  ;VERIFY THAT EACH LOCATION IN BUS ADDRESS AND CHARACTER
1232                                  ;COUNT MEMORY WAS SET TO 177777
1233
1234                          ;  TEST 6
1235                          ;*************************************************************************
1236  003336  012737  000006  001226  TST6:   MOV     #6,TSTNO
1237  003344  012737  003452  001216          MOV     #TST7,NEXT
1238  003352  012737  003422  001220          MOV     #2$,LOCK
1239  003360  012705  177777                  MOV     #177777,R5              ;DATA TO BE LOADED INTO MEMORY
1240  003364  013703  001372                  MOV     DQSEC,R3                ;ADDRESS OF SECONDARY REGISTER
1241  003370  005002                          CLR     R2                      ;FIRST MEMORY LOCATION TO BE TESTED
1242  003372  012700  000010                  MOV     #10,R0                  ;# OF MEMORY LOCATIONS TO BE TESTED
1243  003376  110277  175766          1$:     MOVB    R2,@DQREG               ;SELECT MEMORY LOCATION TO BE TESTED
1244  003402  010577  175764                  MOV     R5,@DQSEC               ;LOAD DATA INTO MEMORY
1245  003406  005202                          INC     R2                      ;GO TO NEXT LOCATION
1246  003410  005300                          DEC     R0                      ;MORE TO GO??
1247  003412  001371                          BNE     1$                      ;BR IF NOT DONE
1248  003414  005002                          CLR     R2                      ;SET UP TO TEST
1249  003416  012700  000010                  MOV     #10,R0                  ;10 (OCTAL) LOCATIONS
1250  003422  110277  175742          2$:     MOVB    R2,@DQREG               ;SELECT LOCATION TO BE TESTED
1251  003426  017704  175740                  MOV     @DQSEC,R4               ;READ DATA
1252  003432  020504                          CMP     R5,R4                   ;COMPARE TO EXPECTED DATA
1253  003434  001401                          BEQ     3$                      ;BR IF GOOD
1254  003436  104002                          HLT     2                       ;CHAR COUNT-BUS ADDRESS MEMORY ERROR
1255  003440  104401          3$:     SCOP1                           ;CHECK FOR FREEZE ON CURRENT LOCATION
1256  003442  005202                  INC     R2                      ;GO TO NEXT LOCATION
1257  003444  005300                  DEC     R0                      ;DONE YET?
1258  003446  001365                  BNE     2$                      ;BR IF FURTHER CHECKING NEEDED
1259  003450  104400          4$:     SCOPE
1260
1261                                  ;BUS ADDRESS AND CHARACTER COUNT MEMORY TEST
1262                                  ;SET EACH LOCATION IN BUS ADDRESS AND CHARACTER COUNT MEMORY
1263                                  ;TO 125252
1264                                  ;VERIFY THAT EACH LOCATION IN BUS ADDRESS AND CHARACTER
1265                                  ;COUNT MEMORY WAS SET TO 125252
1266
1267                          ;  TEST 7
1268                          ;*************************************************************************
1269  003452  012737  000007  001226  TST7:   MOV     #7,TSTNO
1270  003460  012737  003566  001216          MOV     #TST10,NEXT
1271  003466  012737  003536  001220          MOV     #2$,LOCK
1272  003474  012705  125252                  MOV     #125252,R5              ;DATA TO BE LOADED INTO MEMORY
1273  003500  013703  001372                  MOV     DQSEC,R3                ;ADDRESS OF SECONDARY REGISTER
```

```
1274  003504  005002                          CLR     R2              ;FIRST MEMORY LOCATION TO BE TESTED
1275  003506  012700  000010                  MOV     #10,R0          ;# OF MEMORY LOCATIONS TO BE TESTED
1276  003512  110277  175652          1$:     MOVB    R2,@DQREG       ;SELECT MEMORY LOCATION TO BE TESTED
1277  003516  010577  175650                  MOV     R5,@DQSEC       ;LOAD DATA INTO MEMORY
1278  003522  005202                          INC     R2              ;GO TO NEXT LOCATION
1279  003524  005300                          DEC     R0              ;MORE TO GO??
1280  003526  001371                          BNE     1$              ;BR IF NOT DONE
1281  003530  005002                          CLR     R2              ;SET UP TO TEST
1282  003532  012700  000010                  MOV     #10,R0          ;10 (OCTAL) LOCATIONS
1283  003536  110277  175626          2$:     MOVB    R2,@DQREG       ;SELECT LOCATION TO BE TESTED
1284  003542  017704  175624                  MOV     @DQSEC,R4       ;READ DATA
1285  003546  020504                          CMP     R5,R4           ;COMPARE TO EXPECTED DATA
1286  003550  001401                          BEQ     3$              ;BR IF GOOD
1287  003552  104002                          HLT     2               ;CHAR COUNT-BUS ADDRESS MEMORY ERROR
1288  003554  104401                  3$:     SCOP1                   ;CHECK FOR FREEZE ON CURRENT LOCATION
1289  003556  005202                          INC     R2              ;GO TO NEXT LOCATION
1290  003560  005300                          DEC     R0              ;DONE YET?
1291  003562  001365                          BNE     2$              ;BR IF FURTHER CHECKING NEEDED
1292  003564  104400                  4$:     SCOPE
1293
1294                                          ;BUS ADDRESS AND CHARACTER COUNT MEMORY TEST
1295                                          ;SET EACH LOCATION IN BUS ADDRESS AND CHARACTER COUNT MEMORY
1296                                          ;TO 52525
1297                                          ;VERIFY THAT EACH LOCATION IN BUS ADDRESS AND CHARACTER
1298                                          ;COUNT MEMORY WAS SET TO 52525
1299
1300                                  ;  TEST 10
1301                                  ;*************************************************************************
1302  003566  012737  000010  001226  TST10:  MOV     #10,TSTNO
1303  003574  012737  003702  001216          MOV     #TST11,NEXT
1304  003602  012737  003652  001220          MOV     #2$,LOCK
1305  003610  012705  052525                  MOV     #52525,R5       ;DATA TO BE LOADED INTO MEMORY
1306  003614  013703  001372                  MOV     DQSEC,R3        ;ADDRESS OF SECONDARY REGISTER
1307  003620  005002                          CLR     R2              ;FIRST MEMORY LOCATION TO BE TESTED
1308  003622  012700  000010                  MOV     #10,R0          ;# OF MEMORY LOCATIONS TO BE TESTED
1309  003626  110277  175536          1$:     MOVB    R2,@DQREG       ;SELECT MEMORY LOCATION TO BE TESTED
1310  003632  010577  175534                  MOV     R5,@DQSEC       ;LOAD DATA INTO MEMORY
1311  003636  005202                          INC     R2              ;GO TO NEXT LOCATION
1312  003640  005300                          DEC     R0              ;MORE TO GO??
1313  003642  001371                          BNE     1$              ;BR IF NOT DONE
1314  003644  005002                          CLR     R2              ;SET UP TO TEST
1315  003646  012700  000010                  MOV     #10,R0          ;10 (OCTAL) LOCATIONS
1316  003652  110277  175512          2$:     MOVB    R2,@DQREG       ;SELECT LOCATION TO BE TESTED
1317  003656  017704  175510                  MOV     @DQSEC,R4       ;READ DATA
1318  003662  020504                          CMP     R5,R4           ;COMPARE TO EXPECTED DATA
1319  003664  001401                          BEQ     3$              ;BR IF GOOD
1320  003666  104002                          HLT     2               ;CHAR COUNT-BUS ADDRESS MEMORY ERROR
1321  003670  104401                  3$:     SCOP1                   ;CHECK FOR FREEZE ON CURRENT LOCATION
1322  003672  005202                          INC     R2              ;GO TO NEXT LOCATION
1323  003674  005300                          DEC     R0              ;DONE YET?
1324  003676  001365                          BNE     2$              ;BR IF FURTHER CHECKING NEEDED
1325  003700  104400                  4$:     SCOPE
1326
1327                                          ;BUS ADDRESS MEMORY EXTENSION DATA TEST
1328                                          ;LOAD EACH LOCATION IN BUS ADDRESS EXTENTION MEMORY WITH 40
1329                                          ;VERIFY THAT EACH ADDRESS WAS LOADED WITH THE CORRECT DATA
```

```
1330
1331                                    ;  TEST 11
1332                                    ;*************************************************************************
1333  003702  012737  000011  001226    TST11:  MOV     #11,TSTNO
1334  003710  012737  004052  001216            MOV     #TST12,NEXT
1335  003716  012737  004006  001220            MOV     #2$,LOCK
1336  003724  012705  000040                    MOV     #40,R5
1337  003730  012700  000004                    MOV     #4,R0           ;4 LOCATIONS WILL BE ADDRESSED
1338  003734  005003                            CLR     R3              ;FIRST MEMORY ADDRESS =0
1339                                                                    ;(RECEIVER PRIMARY BUS ADDRESS)
1340  003736  010501                    1$:     MOV     R5,R1           ;LOAD 'DATA''
1341  003740  050301                            BIS     R3,R1           ;LOAD SECONDARY ADDRESS
1342  003742  052701  000020                    BIS     #BIT4,R1        ;SET WRITE ENABLE.
1343  003746  112777  000020  175414            MOVB    #BIT4,@DQREG    ;SET MEMORY EXTENSION WRITE ENABLE
1344  003754  150377  175410                    BISB    R3,@DQREG       ;SELECT LOCATION TO BE TESTED
1345  003760  110177  175404                    MOVB    R1,@DQREG       ;LOAD DATA INTO MEMORY EXTENSION BITS
1346  003764  005077  175402                    CLR     @DQSEC          ;REFERENCE SEL 6 (DQSEC)
1347  003770  062703  000002                    ADD     #2,R3           ;NEXT LOCATION TO BE TESTED
1348  003774  005300                            DEC     R0              ;CONTINUE IF ALL LOCATIONS NOT LOADED
1349  003776  001357                            BNE     1$              ;
1350  004000  012700  000004                    MOV     #4,R0           ;4 LOCATIONS TO BE TESTED
1351  004004  005003                            CLR     R3              ;FIRST LOCATION
1352  004006  142777  000037  175354  2$:      BICB    #37,@DQREG      ;REINITIALIZE  ADDRESS
1353  004014  150377  175350                    BISB    R3,@DQREG       ;RESELECT SECONDARY ADDRESS
1354  004020  117704  175344                    MOVB    @DQREG,R4       ;READ DQREG
1355  004024  142704  000037                    BICB    #37,R4          ;CLEAR UNWANTED BITS
1356  004030  020504                            CMP     R5,R4           ;IS DATA CORRECT?
1357  004032  001401                            BEQ     3$              ;BR IF OK.
1358  004034  104003                            HLT     3               ;BUS ADDRESS EXTENSION MEMORY ERROR
1359  004036  104401                    3$:     SCOP1                   ;FREEZE ON CURRENT DATA? (SW09=1)
1360  004040  062703  000002                    ADD     #2,R3           ;UPDATE POINTER TO NEXT ADDRESS
1361  004044  005300                            DEC     R0              ;ALL ADDRESS DONE?
1362  004046  001357                            BNE     2$              ;BR IF NOT DONE.
1363  004050  104400                    4$:     SCOPE                   ;SCOPE THE TEST.
1364
1365                                            ;BUS ADDRESS MEMORY EXTENSION DATA TEST
1366                                            ;LOAD EACH LOCATION IN BUS ADDRESS EXTENTION MEMORY WITH 100
1367                                            ;VERIFY THAT EACH ADDRESS WAS LOADED WITH THE CORRECT DATA
1368
1369                                    ;  TEST 12
1370                                    ;*************************************************************************
1371  004052  012737  000012  001226    TST12:  MOV     #12,TSTNO
1372  004060  012737  004222  001216            MOV     #TST13,NEXT
1373  004066  012737  004156  001220            MOV     #2$,LOCK
1374  004074  012705  000100                    MOV     #100,R5
1375  004100  012700  000004                    MOV     #4,R0           ;4 LOCATIONS WILL BE ADDRESSED
1376  004104  005003                            CLR     R3              ;FIRST MEMORY ADDRESS =0
1377                                                                    ;(RECEIVER PRIMARY BUS ADDRESS)
1378  004106  010501                    1$:     MOV     R5,R1           ;LOAD 'DATA''
1379  004110  050301                            BIS     R3,R1           ;LOAD SECONDARY ADDRESS
1380  004112  052701  000020                    BIS     #BIT4,R1        ;SET WRITE ENABLE.
1381  004116  112777  000020  175244            MOVB    #BIT4,@DQREG    ;SET MEMORY EXTENSION WRITE ENABLE
1382  004124  150377  175240                    BISB    R3,@DQREG       ;SELECT LOCATION TO BE TESTED
1383  004130  110177  175234                    MOVB    R1,@DQREG       ;LOAD DATA INTO MEMORY EXTENSION BITS
1384  004134  005077  175232                    CLR     @DQSEC          ;REFERENCE SEL 6 (DQSEC)
1385  004140  062703  000002                    ADD     #2,R3           ;NEXT LOCATION TO BE TESTED
```

```
1386  004144  005300                          DEC     R0              ;CONTINUE IF ALL LOCATIONS NOT LOADED
1387  004146  001357                          BNE     1$              ;
1388  004150  012700  000004                  MOV     #4,R0           ;4 LOCATIONS TO BE TESTED
1389  004154  005003                          CLR     R3              ;FIRST LOCATION
1390  004156  142777  000037  175204  2$:     BICB    #37,@DQREG      ;REINITIALIZE  ADDRESS
1391  004164  150377  175200                  BISB    R3,@DQREG       ;RESELECT SECONDARY ADDRESS
1392  004170  117704  175174                  MOVB    @DQREG,R4       ;READ DQREG
1393  004174  142704  000037                  BICB    #37,R4          ;CLEAR UNWANTED BITS
1394  004200  020504                          CMP     R5,R4           ;IS DATA CORRECT?
1395  004202  001401                          BEQ     3$              ;BR IF OK.
1396  004204  104003                          HLT     3               ;BUS ADDRESS EXTENSION MEMORY ERROR
1397  004206  104401                  3$:     SCOP1                   ;FREEZE ON CURRENT DATA? (SW09=1)
1398  004210  062703  000002                  ADD     #2,R3           ;UPDATE POINTER TO NEXT ADDRESS
1399  004214  005300                          DEC     R0              ;ALL ADDRESS DONE?
1400  004216  001357                          BNE     2$              ;BR IF NOT DONE.
1401  004220  104400                  4$:     SCOPE                   ;SCOPE THE TEST.
1402
1403                                          ;BUS ADDRESS MEMORY EXTENSION DATA TEST
1404                                          ;LOAD EACH LOCATION IN BUS ADDRESS EXTENTION MEMORY WITH 140
1405                                          ;VERIFY THAT EACH ADDRESS WAS LOADED WITH THE CORRECT DATA
1406
1407                                  ;  TEST 13
1408                                  ;*************************************************************************
1409  004222  012737  000013  001226  TST13:  MOV     #13,TSTNO
1410  004230  012737  004372  001216          MOV     #CHKBB1,NEXT
1411  004236  012737  004326  001220          MOV     #2$,LOCK
1412  004244  012705  000140                  MOV     #140,R5
1413  004250  012700  000004                  MOV     #4,R0           ;4 LOCATIONS WILL BE ADDRESSED
1414  004254  005003                          CLR     R3              ;FIRST MEMORY ADDRESS =0
1415                                                                  ;(RECEIVER PRIMARY BUS ADDRESS)
1416  004256  010501                  1$:     MOV     R5,R1           ;LOAD 'DATA''
1417  004260  050301                          BIS     R3,R1           ;LOAD SECONDARY ADDRESS
1418  004262  052701  000020                  BIS     #BIT4,R1        ;SET WRITE ENABLE.
1419  004266  112777  000020  175074          MOVB    #BIT4,@DQREG    ;SET MEMORY EXTENSION WRITE ENABLE
1420  004274  150377  175070                  BISB    R3,@DQREG       ;SELECT LOCATION TO BE TESTED
1421  004300  110177  175064                  MOVB    R1,@DQREG       ;LOAD DATA INTO MEMORY EXTENSION BITS
1422  004304  005077  175062                  CLR     @DQSEC          ;REFERENCE SEL 6 (DQSEC)
1423  004310  062703  000002                  ADD     #2,R3           ;NEXT LOCATION TO BE TESTED
1424  004314  005300                          DEC     R0              ;CONTINUE IF ALL LOCATIONS NOT LOADED
1425  004316  001357                          BNE     1$              ;
1426  004320  012700  000004                  MOV     #4,R0           ;4 LOCATIONS TO BE TESTED
1427  004324  005003                          CLR     R3              ;FIRST LOCATION
1428  004326  142777  000037  175034  2$:     BICB    #37,@DQREG      ;REINITIALIZE  ADDRESS
1429  004334  150377  175030                  BISB    R3,@DQREG       ;RESELECT SECONDARY ADDRESS
1430  004340  117704  175024                  MOVB    @DQREG,R4       ;READ DQREG
1431  004344  142704  000037                  BICB    #37,R4          ;CLEAR UNWANTED BITS
1432  004350  020504                          CMP     R5,R4           ;IS DATA CORRECT?
1433  004352  001401                          BEQ     3$              ;BR IF OK.
1434  004354  104003                          HLT     3               ;BUS ADDRESS EXTENSION MEMORY ERROR
1435  004356  104401                  3$:     SCOP1                   ;FREEZE ON CURRENT DATA? (SW09=1)
1436  004360  062703  000002                  ADD     #2,R3           ;UPDATE POINTER TO NEXT ADDRESS
1437  004364  005300                          DEC     R0              ;ALL ADDRESS DONE?
1438  004366  001357                          BNE     2$              ;BR IF NOT DONE.
1439  004370  104400                  4$:     SCOPE                   ;SCOPE THE TEST.
1440
1441                                          ;IF CHARACTER DETECT OPTION IS INSTALLED
```

```
                                        ;TESTS 14 THRU 25 WILL BE EXECUTED
                                        ;
004372  032737  020000  001510  CHKBB1: BIT     #BBBIT,DQSTAT
004400  001002                        BNE     TST14
004402  000137  006206                JMP     TST26
                                        ;
                                        ;CHARACTER MEMORY ADDRESSING TEST
                                        ;LOAD EACH LOCATION IN CHARACTER MEMORY WITH ITS
                                        ;ADDRESS (DUPLICATED EVERY 4 BITS)
                                        ;VERIFY THAT EACH LOCATION WAS ADDRESSED
                                        ;
                                    ;  TEST 14
                                    ;*************************************************************************
004406  012737  000014  001226  TST14:  MOV     #14,TSTNO
004414  012737  004546  001216          MOV     #TST15,NEXT
004422  012737  004512  001220          MOV     #2$,LOCK
004430  005005                        CLR     R5                      ;CLEAR DATA TO BE LOADED
004432  012700  000020                MOV     #16.,R0                 ;16(DECIMAL) REGISTERS WILL BE TESTED
004436  005002                        CLR     R2                      ;FIRST ADDRESS =0
004440  012703  000010                MOV     #10,R3                  ;CHARACTER MEMORY WILL BE SELECTED
004444  112777  000010  174716        MOVB    #10,@DQREG              ;SELECT CHARACTER MEMORY
004452  110277  174704          1$:   MOVB    R2,@DQRCSH              ;SELECT LOCATION IN CHARACTER MEMORY
004456  010577  174710                MOV     R5,@DQSEC               ;LOAD DATA
004462  005202                        INC     R2                      ;UPDATE ADDRESS
004464  062705  010421                ADD     #10421,R5               ;UPDATE DATA
004470  005300                        DEC     R0                      ;CONTINUE IF NOT DONE
004472  001367                        BNE     1$                      ;
004474  005005                        CLR     R5                      ;CLEAR EXPECTED DATA
004476  012700  000020                MOV     #16.,R0                 ;16 (DECIMAL) LOCATIONS WILL BE TESTED
004502  005002                        CLR     R2                      ;FIRST LOCATION =0
004504  112777  000010  174656        MOVB    #10,@DQREG              ;SELECT CHARACTER REGISTER
004512  110277  174644          2$:   MOVB    R2,@DQRCSH              ;SELECT ADDRESS TO BE TESTED
004516  017704  174650                MOV     @DQSEC,R4               ;READ ADDRESS TO BE TESTEC
004522  020504                        CMP     R5,R4                   ;ARE EXPECTED AND RECEIVED DATA THE SAME
004524  001401                        BEQ     3$                      ;BR IF GOOD.
004526  104004                        HLT     4                       ;CHARACTER MEMORY ADDRESS ERROR
004530  104401                  3$:   SCOP1                           ;CHECK FOR LOOP ON CUREENT ADDRESS
004532  005202                        INC     R2                      ;UPDATE ADDRESS TO BE TESTED
004534  062705  010421                ADD     #10421,R5               ;UPDATE EXPECTED DATA
004540  005300                        DEC     R0                      ;CONTINUE IF NOT DONE
004542  001363                        BNE     2$                      ;
004544  104400                  4$:   SCOPE                           ;CHECK FOR ITERATIONS, LOOP
                                        ;
                                        ;SEQUENCE MEMORY ADDRESSING TEST
                                        ;LOAD EACH LOCATION IN SEQUENCE MEMORY WITH ITS
                                        ;ADDRESS (DUPLICATED EVERY 4 BITS)
                                        ;VERIFY THAT EACH LOCATION WAS ADDRESSED
                                        ;
                                    ;  TEST 15
                                    ;*************************************************************************
004546  012737  000015  001226  TST15:  MOV     #15,TSTNO
004554  012737  004706  001216          MOV     #TST16,NEXT
004562  012737  004652  001220          MOV     #2$,LOCK
004570  005005                        CLR     R5                      ;CLEAR DATA TO BE LOADED
004572  012700  000020                MOV     #16.,R0                 ;16(DECIMAL) REGISTERS WILL BE TESTED
004576  005002                        CLR     R2                      ;FIRST ADDRESS =0
```

```
1498  004600  012703  000014                  MOV     #14,R3                  ;SEQUENCE MEMORY WILL BE SELECTED
1499  004604  112777  000014  174556          MOVB    #14,@DQREG              ;SELECT SEQUENCE MEMORY
1500  004612  110277  174544          1$:     MOVB    R2,@DQRCSH              ;SELECT LOCATION IN SEQUENCE MEMORY
1501  004616  010577  174550                  MOV     R5,@DQSEC               ;LOAD DATA
1502  004622  005202                          INC     R2                      ;UPDATE ADDRESS
1503  004624  062705  010421                  ADD     #10421,R5               ;UPDATE DATA
1504  004630  005300                          DEC     R0                      ;CONTINUE IF NOT DONE
1505  004632  001367                          BNE     1$                      ;
1506  004634  005005                          CLR     R5                      ;CLEAR EXPECTED DATA
1507  004636  012700  000020                  MOV     #16.,R0                 ;16 (DECIMAL) LOCATIONS WILL BE TESTED
1508  004642  005002                          CLR     R2                      ;FIRST LOCATION =0
1509  004644  112777  000014  174516          MOVB    #14,@DQREG              ;SELECT SEQUENCE REGISTER
1510  004652  110277  174504          2$:     MOVB    R2,@DQRCSH              ;SELECT ADDRESS TO BE TESTED
1511  004656  017704  174510                  MOV     @DQSEC,R4               ;READ ADDRESS TO BE TESTEC
1512  004662  020504                          CMP     R5,R4                   ;ARE EXPECTED AND RECEIVED DATA THE SAME
1513  004664  001401                          BEQ     3$                      ;BR IF GOOD.
1514  004666  104005                          HLT     5                       ;SEQUENCE MEMORY ADDRESS ERROR
1515  004670  104401                  3$:     SCOP1                           ;CHECK FOR LOOP ON CUREENT ADDRESS
1516  004672  005202                          INC     R2                      ;UPDATE ADDRESS TO BE TESTED
1517  004674  062705  010421                  ADD     #10421,R5               ;UPDATE EXPECTED DATA
1518  004700  005300                          DEC     R0                      ;CONTINUE IF NOT DONE
1519  004702  001363                          BNE     2$                      ;
1520  004704  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1521
1522                                          ;CHARACTER MEMORY TEST
1523                                          ;LOAD EACH LOCATION IN CHARACTER MEMORY WITH 000000
1524                                          ;VERIFY THAT EACH LOCATION IN CHARACTER MEMORY WAS
1525                                          ;LOADED WITH 000000
1526
1527                                  ;  TEST 16
1528                                  ;*********************************************************************************
1529  004706  012737  000016  001226  TST16:  MOV     #16,TSTNO
1530  004714  012737  005036  001216          MOV     #TST17,NEXT
1531  004722  012737  005006  001220          MOV,    #2$,LOCK
1532  004730  012705  000000                  MOV     #000000,R5              ;LOAD THE DATA INTO R5
1533  004734  012700  000020                  MOV     #16.,R0                 ;SET COUNT FOR 16
1534  004740  005002                          CLR     R2                      ;CLEAR REGISTER POINTER
1535  004742  012703  000010                  MOV     #10,R3                  ;GET REGISTER IN R3
1536  004746  112777  000010  174414          MOVB    #10,@DQREG              ;SELECT THE REGISTER
1537  004754  110277  174402          1$:     MOVB    R2,@DQRCSH              ;SELECT THE CHARACTER DET REG
1538  004760  010577  174406                  MOV     R5,@DQSEC               ;LOAD THE DATA
1539  004764  005202                          INC     R2                      ;UPDATE THE POINTER
1540  004766  005300                          DEC     R0                      ;ALL DONE YET?
1541  004770  001371                          BNE     1$                      ;BR IF NOT DONE
1542  004772  012700  000020                  MOV     #16.,R0                 ;SET FOR 16 REGISTERS TO BE TESTED
1543  004776  005002                          CLR     R2                      ;ZERO REG POINTER
1544  005000  112777  000010  174362          MOVB    #10,@DQREG              ;SELECT THE REGISTER
1545  005006  110277  174350          2$:     MOVB    R2,@DQRCSH              ;SELECT THE CHARACTER DET REGISTER
1546  005012  017704  174354                  MOV     @DQSEC,R4               ;READ THE DATA
1547  005016  020504                          CMP     R5,R4                   ;WAS IT LOADED CORRECTLY?
1548  005020  001401                          BEQ     3$                      ;BR IF DATA OK
1549  005022  104006                          HLT     6                       ;INCORRECT DATA REPORT ERROR
1550  005024  104401                  3$:     SCOP1                           ;CHECK FOR FREEZE ON DATA
1551  005026  005202                          INC     R2                      ;UPDATE POINTER
1552  005030  005300                          DEC     R0                      ;CHEACK FOR ALL DONE
1553  005032  001365                          BNE     2$                      ;BR IF NOT DONE
```

```
1554  005034  104400                    4$:     SCOPE                   ;SCOPE THIS TEST
1555
1556                                              ;CHARACTER MEMORY TEST
1557                                              ;LOAD EACH LOCATION IN CHARACTER MEMORY WITH 177777
1558                                              ;VERIFY THAT EACH LOCATION IN CHARACTER MEMORY WAS
1559                                              ;LOADED WITH 177777
1560
1561                                    ;  TEST 17
1562                                    ;*************************************************************************
1563  005036  012737  000017  001226    TST17:  MOV     #17,TSTNO
1564  005044  012737  005166  001216            MOV     #TST20,NEXT
1565  005052  012737  005136  001220            MOV     #2$,LOCK
1566  005060  012705  177777                    MOV     #177777,R5      ;LOAD THE DATA INTO R5
1567  005064  012700  000020                    MOV     #16.,R0         ;SET COUNT FOR 16
1568  005070  005002                            CLR     R2              ;CLEAR REGISTER POINTER
1569  005072  012703  000010                    MOV     #10,R3          ;GET REGISTER IN R3
1570  005076  112777  000010  174264            MOVB    #10,@DQREG      ;SELECT THE REGISTER
1571  005104  110277  174252            1$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1572  005110  010577  174256                    MOV     R5,@DQSEC       ;LOAD THE DATA
1573  005114  005202                            INC     R2              ;UPDATE THE POINTER
1574  005116  005300                            DEC     R0              ;ALL DONE YET?
1575  005120  001371                            BNE     1$              ;BR IF NOT DONE
1576  005122  012700  000020                    MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1577  005126  005002                            CLR     R2              ;ZERO REG POINTER
1578  005130  112777  000010  174232            MOVB    #10,@DQREG      ;SELECT THE REGISTER
1579  005136  110277  174220            2$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1580  005142  017704  174224                    MOV     @DQSEC,R4       ;READ THE DATA
1581  005146  020504                            CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1582  005150  001401                            BEQ     3$              ;BR IF DATA OK
1583  005152  104006                            HLT     6               ;INCORRECT DATA REPORT ERROR
1584  005154  104401                    3$:     SCOP1                   ;CHECK FOR FREEZE ON DATA
1585  005156  005202                            INC     R2              ;UPDATE POINTER
1586  005160  005300                            DEC     R0              ;CHEACK FOR ALL DONE
1587  005162  001365                            BNE     2$              ;BR IF NOT DONE
1588  005164  104400                    4$:     SCOPE                   ;SCOPE THIS TEST
1589
1590                                              ;CHARACTER MEMORY TEST
1591                                              ;LOAD EACH LOCATION IN CHARACTER MEMORY WITH 125252
1592                                              ;VERIFY THAT EACH LOCATION IN CHARACTER MEMORY WAS
1593                                              ;LOADED WITH 125252
1594
1595                                    ;  TEST 20
1596                                    ;*************************************************************************
1597  005166  012737  000020  001226    TST20:  MOV     #20,TSTNO
1598  005174  012737  005316  001216            MOV     #TST21,NEXT
1599  005202  012737  005266  001220            MOV     #2$,LOCK
1600  005210  012705  125252                    MOV     #125252,R5      ;LOAD THE DATA INTO R5
1601  005214  012700  000020                    MOV     #16.,R0         ;SET COUNT FOR 16
1602  005220  005002                            CLR     R2              ;CLEAR REGISTER POINTER
1603  005222  012703  000010                    MOV     #10,R3          ;GET REGISTER IN R3
1604  005226  112777  000010  174134            MOVB    #10,@DQREG      ;SELECT THE REGISTER
1605  005234  110277  174122            1$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1606  005240  010577  174126                    MOV     R5,@DQSEC       ;LOAD THE DATA
1607  005244  005202                            INC     R2              ;UPDATE THE POINTER
1608  005246  005300                            DEC     R0              ;ALL DONE YET?
1609  005250  001371                            BNE     1$              ;BR IF NOT DONE
```

```
1610  005252  012700  000020            MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1611  005256  005002                    CLR     R2              ;ZERO REG POINTER
1612  005260  112777  000010  174102    MOVB    #10,@DQREG      ;SELECT THE REGISTER
1613  005266  110277  174070        2$: MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1614  005272  017704  174074            MOV     @DQSEC,R4       ;READ THE DATA
1615  005276  020504                    CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1616  005300  001401                    BEQ     3$              ;BR IF DATA OK
1617  005302  104006                    HLT     6               ;INCORRECT DATA REPORT ERROR
1618  005304  104401                3$: SCOP1                   ;CHECK FOR FREEZE ON DATA
1619  005306  005202                    INC     R2              ;UPDATE POINTER
1620  005310  005300                    DEC     R0              ;CHEACK FOR ALL DONE
1621  005312  001365                    BNE     2$              ;BR IF NOT DONE
1622  005314  104400                4$: SCOPE                   ;SCOPE THIS TEST
1623
1624                                    ;CHARACTER MEMORY TEST
1625                                    ;LOAD EACH LOCATION IN CHARACTER MEMORY WITH 52525
1626                                    ;VERIFY THAT EACH LOCATION IN CHARACTER MEMORY WAS
1627                                    ;LOADED WITH 52525
1628
1629                                ;  TEST 21
1630                                ;*****************************************************************************
1631  005316  012737  000021  001226 TST21: MOV     #21,TSTNO
1632  005324  012737  005446  001216        MOV     #TST22,NEXT
1633  005332  012737  005416  001220        MOV     #2$,LOCK
1634  005340  012705  052525                MOV     #52525,R5       ;LOAD THE DATA INTO R5
1635  005344  012700  000020                MOV     #16.,R0         ;SET COUNT FOR 16
1636  005350  005002                        CLR     R2              ;CLEAR REGISTER POINTER
1637  005352  012703  000010                MOV     #10,R3          ;GET REGISTER IN R3
1638  005356  112777  000010  174004        MOVB    #10,@DQREG      ;SELECT THE REGISTER
1639  005364  110277  173772            1$: MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1640  005370  010577  173776                MOV     R5,@DQSEC       ;LOAD THE DATA
1641  005374  005202                        INC     R2              ;UPDATE THE POINTER
1642  005376  005300                        DEC     R0              ;ALL DONE YET?
1643  005400  001371                        BNE     1$              ;BR IF NOT DONE
1644  005402  012700  000020                MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1645  005406  005002                        CLR     R2              ;ZERO REG POINTER
1646  005410  112777  000010  173752        MOVB    #10,@DQREG      ;SELECT THE REGISTER
1647  005416  110277  173740            2$: MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1648  005422  017704  173744                MOV     @DQSEC,R4       ;READ THE DATA
1649  005426  020504                        CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1650  005430  001401                        BEQ     3$              ;BR IF DATA OK
1651  005432  104006                        HLT     6               ;INCORRECT DATA REPORT ERROR
1652  005434  104401                    3$: SCOP1                   ;CHECK FOR FREEZE ON DATA
1653  005436  005202                        INC     R2              ;UPDATE POINTER
1654  005440  005300                        DEC     R0              ;CHEACK FOR ALL DONE
1655  005442  001365                        BNE     2$              ;BR IF NOT DONE
1656  005444  104400                    4$: SCOPE                   ;SCOPE THIS TEST
1657
1658                                    ;SEQUENCE MEMORY TEST
1659                                    ;LOAD EACH LOCATION IN SEQUENCE MEMORY WITH 000000
1660                                    ;VERIFY THAT EACH LOCATION IN SEQUENCE MEMORY WAS
1661                                    ;LOADED WITH 000000
1662
1663                                ;  TEST 22
1664                                ;*****************************************************************************
1665  005446  012737  000022  001226 TST22: MOV     #22,TSTNO
```

```
1666  005454  012737  005576  001216          MOV     #TST23,NEXT
1667  005462  012737  005546  001220          MOV     #2$,LOCK
1668  005470  012705  000000                  MOV     #000000,R5      ;LOAD THE DATA INTO R5
1669  005474  012700  000020                  MOV     #16.,R0         ;SET COUNT FOR 16
1670  005500  005002                          CLR     R2              ;CLEAR REGISTER POINTER
1671  005502  012703  000014                  MOV     #14,R3          ;GET REGISTER IN R3
1672  005506  112777  000014  173654          MOVB    #14,@DQREG      ;SELECT THE REGISTER
1673  005514  110277  173642          1$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1674  005520  010577  173646                  MOV     R5,@DQSEC       ;LOAD THE DATA
1675  005524  005202                          INC     R2              ;UPDATE THE POINTER
1676  005526  005300                          DEC     R0              ;ALL DONE YET?
1677  005530  001371                          BNE     1$              ;BR IF NOT DONE
1678  005532  012700  000020                  MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1679  005536  005002                          CLR     R2              ;ZERO REG POINTER
1680  005540  112777  000014  173622          MOVB    #14,@DQREG      ;SELECT THE REGISTER
1681  005546  110277  173610          2$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1682  005552  017704  173614                  MOV     @DQSEC,R4       ;READ THE DATA
1683  005556  020504                          CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1684  005560  001401                          BEQ     3$              ;BR IF DATA OK
1685  005562  104007                          HLT     7               ;INCORRECT DATA REPORT ERROR
1686  005564  104401                  3$:     SCOP1                   ;CHECK FOR FREEZE ON DATA
1687  005566  005202                          INC     R2              ;UPDATE POINTER
1688  005570  005300                          DEC     R0              ;CHEACK FOR ALL DONE
1689  005572  001365                          BNE     2$              ;BR IF NOT DONE
1690  005574  104400                  4$:     SCOPE                   ;SCOPE THIS TEST
1691
1692                                          ;SEQUENCE MEMORY TEST
1693                                          ;LOAD EACH LOCATION IN SEQUENCE MEMORY WITH 177777
1694                                          ;VERIFY THAT EACH LOCATION IN SEQUENCE MEMORY WAS
1695                                          ;LOADED WITH 177777
1696
1697                                  ;  TEST 23
1698                                  ;*************************************************************************
1699  005576  012737  000023  001226  TST23:  MOV     #23,TSTNO
1700  005604  012737  005726  001216          MOV     #TST24,NEXT
1701  005612  012737  005676  001220          MOV     #2$,LOCK
1702  005620  012705  177777                  MOV     #177777,R5      ;LOAD THE DATA INTO R5
1703  005624  012700  000020                  MOV     #16.,R0         ;SET COUNT FOR 16
1704  005630  005002                          CLR     R2              ;CLEAR REGISTER POINTER
1705  005632  012703  000014                  MOV     #14,R3          ;GET REGISTER IN R3
1706  005636  112777  000014  173524          MOVB    #14,@DQREG      ;SELECT THE REGISTER
1707  005644  110277  173512          1$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1708  005650  010577  173516                  MOV     R5,@DQSEC       ;LOAD THE DATA
1709  005654  005202                          INC     R2              ;UPDATE THE POINTER
1710  005656  005300                          DEC     R0              ;ALL DONE YET?
1711  005660  001371                          BNE     1$              ;BR IF NOT DONE
1712  005662  012700  000020                  MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1713  005666  005002                          CLR     R2              ;ZERO REG POINTER
1714  005670  112777  000014  173472          MOVB    #14,@DQREG      ;SELECT THE REGISTER
1715  005676  110277  173460          2$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1716  005702  017704  173464                  MOV     @DQSEC,R4       ;READ THE DATA
1717  005706  020504                          CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1718  005710  001401                          BEQ     3$              ;BR IF DATA OK
1719  005712  104007                          HLT     7               ;INCORRECT DATA REPORT ERROR
1720  005714  104401                  3$:     SCOP1                   ;CHECK FOR FREEZE ON DATA
1721  005716  005202                          INC     R2              ;UPDATE POINTER
```

```
1722  005720  005300                        DEC     R0              ;CHEACK FOR ALL DONE
1723  005722  001365                        BNE     2$              ;BR IF NOT DONE
1724  005724  104400                4$:     SCOPE                   ;SCOPE THIS TEST
1725
1726                                        ;SEQUENCE MEMORY TEST
1727                                        ;LOAD EACH LOCATION IN SEQUENCE MEMORY WITH 125252
1728                                        ;VERIFY THAT EACH LOCATION IN SEQUENCE MEMORY WAS
1729                                        ;LOADED WITH 125252
1730
1731                                ;  TEST 24
1732                                ;*************************************************************************
1733  005726  012737  000024  001226 TST24:  MOV     #24,TSTNO
1734  005734  012737  006056  001216         MOV     #TST25,NEXT
1735  005742  012737  006026  001220         MOV     #2$,LOCK
1736  005750  012705  125252                 MOV     #125252,R5      ;LOAD THE DATA INTO R5
1737  005754  012700  000020                 MOV     #16.,R0         ;SET COUNT FOR 16
1738  005760  005002                         CLR     R2              ;CLEAR REGISTER POINTER
1739  005762  012703  000014                 MOV     #14,R3          ;GET REGISTER IN R3
1740  005766  112777  000014  173374         MOVB    #14,@DQREG      ;SELECT THE REGISTER
1741  005774  110277  173362          1$:    MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1742  006000  010577  173366                 MOV     R5,@DQSEC       ;LOAD THE DATA
1743  006004  005202                         INC     R2              ;UPDATE THE POINTER
1744  006006  005300                         DEC     R0              ;ALL DONE YET?
1745  006010  001371                         BNE     1$              ;BR IF NOT DONE
1746  006012  012700  000020                 MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1747  006016  005002                         CLR     R2              ;ZERO REG POINTER
1748  006020  112777  000014  173342         MOVB    #14,@DQREG      ;SELECT THE REGISTER
1749  006026  110277  173330          2$:    MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1750  006032  017704  173334                 MOV     @DQSEC,R4       ;READ THE DATA
1751  006036  020504                         CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1752  006040  001401                         BEQ     3$              ;BR IF DATA OK
1753  006042  104007                         HLT     7               ;INCORRECT DATA REPORT ERROR
1754  006044  104401                3$:     SCOP1                   ;CHECK FOR FREEZE ON DATA
1755  006046  005202                         INC     R2              ;UPDATE POINTER
1756  006050  005300                         DEC     R0              ;CHEACK FOR ALL DONE
1757  006052  001365                         BNE     2$              ;BR IF NOT DONE
1758  006054  104400                4$:     SCOPE                   ;SCOPE THIS TEST
1759
1760                                        ;SEQUENCE MEMORY TEST
1761                                        ;LOAD EACH LOCATION IN SEQUENCE MEMORY WITH 52525
1762                                        ;VERIFY THAT EACH LOCATION IN SEQUENCE MEMORY WAS
1763                                        ;LOADED WITH 52525
1764
1765                                ;  TEST 25
1766                                ;*************************************************************************
1767  006056  012737  000025  001226 TST25:  MOV     #25,TSTNO
1768  006064  012737  006206  001216         MOV     #TST26,NEXT
1769  006072  012737  006156  001220         MOV     #2$,LOCK
1770  006100  012705  052525                 MOV     #52525,R5       ;LOAD THE DATA INTO R5
1771  006104  012700  000020                 MOV     #16.,R0         ;SET COUNT FOR 16
1772  006110  005002                         CLR     R2              ;CLEAR REGISTER POINTER
1773  006112  012703  000014                 MOV     #14,R3          ;GET REGISTER IN R3
1774  006116  112777  000014  173244         MOVB    #14,@DQREG      ;SELECT THE REGISTER
1775  006124  110277  173232          1$:    MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REG
1776  006130  010577  173236                 MOV     R5,@DQSEC       ;LOAD THE DATA
1777  006134  005202                         INC     R2              ;UPDATE THE POINTER
```

```
1778  006136  005300                          DEC     R0              ;ALL DONE YET?
1779  006140  001371                          BNE     1$              ;BR IF NOT DONE
1780  006142  012700  000020                  MOV     #16.,R0         ;SET FOR 16 REGISTERS TO BE TESTED
1781  006146  005002                          CLR     R2              ;ZERO REG POINTER
1782  006150  112777  000014  173212          MOVB    #14,@DQREG      ;SELECT THE REGISTER
1783  006156  110277  173200          2$:     MOVB    R2,@DQRCSH      ;SELECT THE CHARACTER DET REGISTER
1784  006162  017704  173204                  MOV     @DQSEC,R4       ;READ THE DATA
1785  006166  020504                          CMP     R5,R4           ;WAS IT LOADED CORRECTLY?
1786  006170  001401                          BEQ     3$              ;BR IF DATA OK
1787  006172  104007                          HLT     7               ;INCORRECT DATA REPORT ERROR
1788  006174  104401                  3$:     SCOP1                   ;CHECK FOR FREEZE ON DATA
1789  006176  005202                          INC     R2              ;UPDATE POINTER
1790  006200  005300                          DEC     R0              ;CHEACK FOR ALL DONE
1791  006202  001365                          BNE     2$              ;BR IF NOT DONE
1792  006204  104400                  4$:     SCOPE                   ;SCOPE THIS TEST
1793
1794                                          ;RECEIVER CONTROL REGISTER MASTER CLEAR TEST
1795                                          ;SET ALL READ/WRITE BITS IN RECEIVER CONTROL REGISTER
1796                                          ;ISSUE MASTER CLEAR
1797                                          ;VERIFY THAT RECEIVER CONTROL WAS CLEARED
1798
1799                                  ;  TEST 26
1800                                  ;*************************************************************************
1801  006206  012737  000026  001226  TST26:  MOV     #26,TSTNO
1802  006214  012737  006272  001216          MOV     #TST27,NEXT
1803  006222  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1804  006230  013703  001360                  MOV     DQRCSR,R3               ;BUS ADDRESS OF
1805                                                                          ;RECEIVER CONTROL REGISTER
1806  006234  012777  170372  173116          MOV     #170372,@DQRCSR         ;LOAD RECEIVER CONTROL REGISTER
1807  006242  112777  000012  173120          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1808  006250  052777  000040  173114          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1809  006256  005005                          CLR     R5                      ;(R5)=EXPECTED 170372 IN
1810                                                                          ;RECEIVER CONTROL REGISTER, 0
1811  006260  017704  173074                  MOV     @DQRCSR,R4              ;(R4)=ACTUAL 170372 IN
1812                                                                          ;RECEIVER CONTROL REGISTER
1813  006264  001401                          BEQ     1$                      ;BR IF ALL OK.
1814  006266  104000                          HLT                             ;PRIMARY REGISTER MASTER CLEAR ERROR
1815  006270  104400                  1$:     SCOPE
1816
1817                                          ;RECEIVER CONTROL REGISTER MASTER CLEAR TEST
1818                                          ;SET ALL READ/WRITE BITS IN RECEIVER CONTROL REGISTER
1819                                          ;ISSUE MASTER CLEAR
1820                                          ;VERIFY THAT RECEIVER CONTROL WAS CLEARED
1821
1822                                  ;  TEST 27
1823                                  ;*************************************************************************
1824  006272  012737  000027  001226  TST27:  MOV     #27,TSTNO
1825  006300  012737  006356  001216          MOV     #CHKBA1,NEXT
1826  006306  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1827  006314  013703  001360                  MOV     DQRCSR,R3               ;BUS ADDRESS OF
1828                                                                          ;RECEIVER CONTROL REGISTER
1829  006320  012777  007400  173032          MOV     #07400,@DQRCSR          ;LOAD RECEIVER CONTROL REGISTER
1830  006326  112777  000012  173034          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1831  006334  052777  000040  173030          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1832  006342  005005                          CLR     R5                      ;(R5)=EXPECTED 07400 IN
1833                                                                          ;RECEIVER CONTROL REGISTER, 0
```

```
1834  006344  017704  173010                  MOV     @DQRCSR,R4              ;(R4)=ACTUAL 07400 IN
1835                                                                          ;RECEIVER CONTROL REGISTER
1836  006350  001401                          BEQ     1$                      ;BR IF ALL OK.
1837  006352  104000                          HLT                             ;PRIMARY REGISTER MASTER CLEAR ERROR
1838  006354  104400                  1$:     SCOPE
1839
1840                                          ;IF DATASET CONTROL OPTION IS INSTALLED
1841                                          ;TEST 30 WILL BE EXECUTED
1842
1843  006356  032737  010000  001510  CHKBA1: BIT     #BABIT,DQSTAT
1844  006364  001436                          BEQ     TST31
1845
1846
1847                                          ;TRANSMITTER CONTROL REGISTER MASTER CLEAR TEST
1848                                          ;SET ALL READ/WRITE BITS IN TRANSMITTER CONTROL REGISTER
1849                                          ;ISSUE MASTER CLEAR
1850                                          ;VERIFY THAT TRANSMITTER CONTROL WAS CLEARED
1851
1852                                  ;  TEST 30
1853                                  ;*************************************************************************
1854  006366  012737  000030  001226  TST30:  MOV     #30,TSTNO
1855  006374  012737  006460  001216          MOV     #CHKBA2,NEXT
1856  006402  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1857  006410  013703  001364                  MOV     DQTCSR,R3               ;BUS ADDRESS OF
1858                                                                          ;TRANSMITTER CONTROL REGISTER
1859  006414  012777  101772  172742          MOV     #101772,@DQTCSR         ;LOAD TRANSMITTER CONTROL REGISTER
1860  006422  112777  000012  172740          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1861  006430  052777  000040  172734          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1862  006436  005005                          CLR     R5                      ;(R5)=EXPECTED 101772 IN
1863                                                                          ;TRANSMITTER CONTROL REGISTER, 0
1864  006440  017704  172720                  MOV     @DQTCSR,R4              ;(R4)=ACTUAL 101772 IN
1865                                                                          ;TRANSMITTER CONTROL REGISTER
1866  006444  042704  100000                  BIC     #100000,R4              ;100000 OFF UNWANTED BITS.
1867  006450  020504                          CMP     R5,R4                   ;EXPECTED=ACTUAL?
1868  006452  001401                          BEQ     1$                      ;BR IF ALL OK.
1869  006454  104000                          HLT                             ;PRIMARY REGISTER MASTER CLEAR ERROR
1870  006456  104400                  1$:     SCOPE
1871  006460  000435                  CHKBA2: BR      TST32
1872
1873                                          ;IF DATASET CONTROL OPTION IS NOT INSTALLED
1874                                          ;TEST 31 WILL BE EXECUTED
1875
1876
1877                                          ;TRANSMITTER CONTROL REGISTER MASTER CLEAR TEST
1878                                          ;SET ALL READ/WRITE BITS IN TRANSMITTER CONTROL REGISTER
1879                                          ;ISSUE MASTER CLEAR
1880                                          ;VERIFY THAT TRANSMITTER CONTROL WAS CLEARED
1881
1882                                  ;  TEST 31
1883                                  ;*************************************************************************
1884  006462  012737  000031  001226  TST31:  MOV     #31,TSTNO
1885  006470  012737  006554  001216          MOV     #TST32,NEXT
1886  006476  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1887  006504  013703  001364                  MOV     DQTCSR,R3               ;BUS ADDRESS OF
1888                                                                          ;TRANSMITTER CONTROL REGISTER
1889  006510  012777  101772  172646          MOV     #101772,@DQTCSR         ;LOAD TRANSMITTER CONTROL REGISTER
```

```
1890  006516  112777  000012  172644          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1891  006524  052777  000040  172640          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1892  006532  005005                          CLR     R5                      ;(R5)=EXPECTED 101772 IN
1893                                                                          ;TRANSMITTER CONTROL REGISTER, 0
1894  006534  017704  172624                  MOV     @DQTCSR,R4              ;(R4)=ACTUAL 101772 IN
1895                                                                          ;TRANSMITTER CONTROL REGISTER
1896  006540  042704  001400                  BIC     #1400,R4                ;1400 OFF UNWANTED BITS.
1897  006544  020504                          CMP     R5,R4                   ;EXPECTED=ACTUAL?
1898  006546  001401                          BEQ     1$                      ;BR IF ALL OK.
1899  006550  104000                          HLT                             ;PRIMARY REGISTER MASTER CLEAR ERROR
1900  006552  104400              1$:         SCOPE
1901
1902                                          ;ERROR REGISTER MASTER CLEAR TEST
1903                                          ;SET ALL READ/WRITE BITS IN ERROR REGISTER
1904                                          ;ISSUE MASTER CLEAR
1905                                          ;VERIFY THAT ERROR WAS CLEARED
1906
1907                                  ;  TEST 32
1908                                  ;*************************************************************************
1909  006554  012737  000032  001226  TST32:  MOV     #32,TSTNO
1910  006562  012737  006646  001216          MOV     #TST33,NEXT
1911  006570  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1912  006576  013703  001366                  MOV     DQERR,R3                ;BUS ADDRESS OF
1913                                                                          ;ERROR REGISTER
1914  006602  012777  107777  172556          MOV     #107777,@DQERR          ;LOAD ERROR REGISTER
1915  006610  112777  000012  172552          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1916  006616  052777  000040  172546          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1917  006624  005005                          CLR     R5                      ;(R5)=EXPECTED 107777 IN
1918                                                                          ;ERROR REGISTER, 0
1919  006626  017704  172534                  MOV     @DQERR,R4               ;(R4)=ACTUAL 107777 IN
1920                                                                          ;ERROR REGISTER
1921  006632  042704  060000                  BIC     #60000,R4               ;60000 OFF UNWANTED BITS.
1922  006636  020504                          CMP     R5,R4                   ;EXPECTED=ACTUAL?
1923  006640  001401                          BEQ     1$                      ;BR IF ALL OK.
1924  006642  104000                          HLT                             ;PRIMARY REGISTER MASTER CLEAR ERROR
1925  006644  104400              1$:         SCOPE
1926
1927                                          ;SYNC REGISTER MASTER CLEAR TEST
1928                                          ;SET ALL READ/WRITE BITS IN SYNC REGISTER
1929                                          ;ISSUE MASTER CLEAR
1930                                          ;VERIFY THAT SYNC WAS CLEARED
1931
1932                                  ;  TEST 33
1933                                  ;*************************************************************************
1934  006646  012737  000033  001226  TST33:  MOV     #33,TSTNO
1935  006654  012737  006742  001216          MOV     #TST34,NEXT
1936  006662  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1937  006670  012703  000011                  MOV     #11,R3                  ;GET ADDRESS OF SECONDARY
1938                                                                          ;REGISTER TO BE TESTED
1939  006674  110377  172470                  MOVB    R3,@DQREG               ;SELECT CEREG
1940                                                                          ;SECONDARY REGISTER
1941  006700  012777  177777  172464          MOV     #177777,@DQSEC          ;LOAD SYNC REGISTER
1942  006706  112777  000012  172454          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1943  006714  052777  000040  172450          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1944  006722  110377  172442                  MOVB    R3,@DQREG               ;SELECT SYNC SECONDARY REGISTER
1945  006726  005005                          CLR     R5                      ;(R5)=EXPECTED 177777 IN
```

```
1946
1947  006730  017704  172436                  MOV     @DQSEC,R4               ;SYNC REGISTER, 0
1948                                                                            ;(R4)=ACTUAL 177777 IN
1949  006734  001401                          BEQ     1$                      ;SYNC REGISTER
1950  006736  104000                          HLT                             ;BR IF ALL OK.
1951  006740  104400                  1$:     SCOPE                           ;SECONDARY REGISTER MASTER CLEAR ERROR
1952
1953                                          ;MISCELLANEAOUS REGISTER MASTER CLEAR TEST
1954                                          ;SET ALL READ/WRITE BITS IN MISCELLANEAOUS REGISTER
1955                                          ;ISSUE MASTER CLEAR
1956                                          ;VERIFY THAT MISCELLANEAOUS WAS CLEARED
1957
1958                                  ;  TEST 34
1959                                  ;*************************************************************************
1960  006742  012737  000034  001226  TST34:  MOV     #34,TSTNO
1961  006750  012737  007036  001216          MOV     #CHKAB1,NEXT
1962  006756  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1963  006764  012703  000012                  MOV     #12,R3                  ;GET ADDRESS OF SECONDARY
1964                                                                          ;REGISTER TO BE TESTED
1965  006770  110377  172374                  MOVB    R3,@DQREG               ;SELECT CEREG
1966                                                                          ;SECONDARY REGISTER
1967  006774  012777  007717  172370          MOV     #7717,@DQSEC            ;LOAD MISCELLANEAOUS REGISTER
1968  007002  112777  000012  172360          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
1969  007010  052777  000040  172354          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
1970  007016  110377  172346                  MOVB    R3,@DQREG               ;SELECT MISCELLANEAOUS SECONDARY REGISTER
1971  007022  005005                          CLR     R5                      ;(R5)=EXPECTED 7717 IN
1972                                                                          ;MISCELLANEAOUS REGISTER, 0
1973  007024  017704  172342                  MOV     @DQSEC,R4               ;(R4)=ACTUAL 7717 IN
1974                                                                          ;MISCELLANEAOUS REGISTER
1975  007030  001401                          BEQ     1$                      ;BR IF ALL OK.
1976  007032  104000                          HLT                             ;SECONDARY REGISTER MASTER CLEAR ERROR
1977  007034  104400                  1$:     SCOPE
1978
1979                                          ;IF BCC OPTION IS INSTALLED
1980                                          ;TEST 35 WILL BE EXECUTED
1981
1982  007036  032737  002000  001510  CHKAB1: BIT     #ABBIT,DQSTAT
1983  007044  001436                          BEQ     .EOP
1984
1985                                          ;BCC POLYNOMIAL REGISTER MASTER CLEAR TEST
1986                                          ;SET ALL READ/WRITE BITS IN BCC POLYNOMIAL REGISTER
1987                                          ;ISSUE MASTER CLEAR
1988                                          ;VERIFY THAT BCC POLYNOMIAL WAS CLEARED
1989
1990                                  ;  TEST 35
1991                                  ;*************************************************************************
1992  007046  012737  000035  001226  TST35:  MOV     #35,TSTNO
1993  007054  012737  007142  001216          MOV     #.EOP,NEXT
1994  007062  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1995  007070  012703  000017                  MOV     #17,R3                  ;GET ADDRESS OF SECONDARY
1996                                                                          ;REGISTER TO BE TESTED
1997  007074  110377  172270                  MOVB    R3,@DQREG               ;SELECT CEREG
1998                                                                          ;SECONDARY REGISTER
1999  007100  012777  177777  172264          MOV     #177777,@DQSEC          ;LOAD BCC POLYNOMIAL REGISTER
2000  007106  112777  000012  172254          MOVB    #12,@DQREG              ;SELECT MISCELLANEOUS REGISTER
2001  007114  052777  000040  172250          BIS     #BIT5,@DQSEC            ;ISSUE MASTER CLEAR
```

```
2002  007122  110377  172242              MOVB    R3,@DQREG          ;SELECT BCC POLYNOMIAL SECONDARY REGISTER
2003  007126  005005                      CLR     R5                 ;(R5)=EXPECTED 177777 IN
2004                                                                 ;BCC POLYNOMIAL REGISTER, 0
2005  007130  017704  172236              MOV     @DQSEC,R4          ;(R4)=ACTUAL 177777 IN
2006                                                                 ;BCC POLYNOMIAL REGISTER
2007  007134  001401                      BEQ     1$                 ;BR IF ALL OK.
2008  007136  104000                      HLT                        ;SECONDARY REGISTER MASTER CLEAR ERROR
2009  007140  104400              1$:     SCOPE
```

```
2010
2011                                        ;END OF PASS
2012                                        ;TYPE NAME OF TEST
2013                                        ;UPDATE PASS COUNT
2014                                        ;CHECK FOR EXIT TO ACT-11
2015                                        ;RESTART TEST
2016
2017  007142  005037  001234        .EOP:   CLR     LSTERR                  ;CLEAR LAST ERROR PC
2018  007146  005037  001312                CLR     ERRFLG                  ;CLEAR ERROR FLAG
2019  007152  005237  001230                INC     PASCNT                  ;UPDATE PASS COUNT
2020  007156  104402                        TYPE
2021  007160  011372                        MEPASS
2022  007162  104402                        TYPE
2023  007164  011553                        MCSRX
2024  007166  104411                        CNVRT
2025  007170  007300                        XCSR
2026  007172  104402                        TYPE
2027  007174  011561                        MVECX
2028  007176  104411                        CNVRT
2029  007200  007306                        XVEC
2030  007202  104402                        TYPE
2031  007204  011567                        MPASSX
2032  007206  104411                        CNVRT
2033  007210  007314                        XPASS
2034  007212  104402                        TYPE
2035  007214  011600                        MERRX
2036  007216  104411                        CNVRT
2037  007220  007322                        XERR
2038  007222  013777  001230  171752        MOV     PASCNT,@LIGHTS          ;DISPLAY PASS COUNT
2039  007230  005337  001276                DEC     SAVNUM
2040  007234  001013                        BNE     RESTRT
2041  007236  013737  001504  001276        MOV     DQNUM,SAVNUM
2042  007244  013701  000042                MOV     @#42,R1                 ;CHECK FOR ACT-11 OR DDP
2043  007250  001405                        BEQ     RESTRT                  ;IF NOT, CONTINUE TESTING
2044  007252  000005                        RESET
2045  007254                        LOGICAL:
2046  007254  004711                        JSR     PC,(R1)
2047  007256  000240                        NOP
2048  007260  000240                        NOP
2049  007262  000240                        NOP
2050  007264  104414                RESTRT: CKSWR
2051  007266  012737  002254  001214        MOV     #TST1,RETURN
2052  007274  000137  002254                JMP     TST1
2053  007300  000001                XCSR:   1
2054  007302     006     002                .BYTE   6,2
2055  007304  001360                        DQRCSR
2056  007306  000001                XVEC:   1
2057  007310     003     002                .BYTE   3,2
2058  007312  001350                        DQRVEC
2059  007314  000001                XPASS:  1
2060  007316     006     002                .BYTE   6,2
2061  007320  001230                        PASCNT
2062  007322  000001                XERR:   1
2063  007324     006     002                .BYTE   6,2
2064  007326  001232                        ERRCNT
2065
```

```
2066                                            ;SCOPE LOOP AND INTERATION HANDLER
2067
2068  007330  104414                .SCOPE: CKSWR
2069  007332  032777  040000  171640        BIT     #BIT14,@SWR
2070  007340  001407                TTST:   BEQ     1$
2071  007342  000432                        BR      3$
2072  007344  105777  171634                TSTB    @TKCSR
2073  007350  100027                        BPL     3$
2074  007352  017700  171630                MOV     @TKDBR,R0
2075  007356  000412                        BR      2$
2076  007360  032777  004000  171612  1$:   BIT     #SW11,@SWR
2077  007366  001006                        BNE     2$
2078  007370  005237  001224                INC     LPCNT
2079  007374  023737  001224  001222        CMP     LPCNT,ICOUNT
2080  007402  001012                        BNE     3$
2081  007404  105037  001312          2$:   CLRB    ERRFLG
2082  007410  005037  001224                CLR     LPCNT
2083  007414  012737  002000  001222        MOV     #2000,ICOUNT
2084  007422  013737  001216  001214        MOV     NEXT,RETURN
2085  007430  013716  001214          3$:   MOV     RETURN,(SP)
2086  007434  000002                        RTI
2087  007436  001407                BRW:    1407
2088  007440  000432                BRX:    432
2089
2090                                            ;CHECK FOR FREEZE ON CURRENT DATA
2091
2092  007442  104414                .SCOP1: CKSWR
2093  007444  032777  001000  171526        BIT     #SW09,@SWR
2094  007452  001402                        BEQ     1$
2095  007454  013716  001220                MOV     LOCK,(SP)
2096  007460  000002                1$:     RTI
2097
2098                                            ;TELETYPE OUTPUT ROUTINE
2099
2100  007462  010546                .TYPE:  MOV     R5,-(SP)
2101  007464  017605  000002                MOV     @2(SP),R5
2102  007470  062766  000002  000002        ADD     #2,2(SP)
2103  007476  005737  011152          1$:   TST     @#RDSW
2104  007502  001004                        BNE     300$
2105  007504  032777  010000  171466        BIT     #SW12,@SWR
2106  007512  001024                        BNE     3$
2107  007514  105715                300$:   TSTB    (R5)
2108  007516  100014                        BPL     2$
2109  007520  105777  171464                TSTB    @TPCSR
2110  007524  100375                        BPL     .-4
2111  007526  012777  000015  171456        MOV     #15,@TPDBR
2112  007534  105777  171450                TSTB    @TPCSR
2113  007540  100375                        BPL     .-4
2114  007542  012777  000012  171442        MOV     #12,@TPDBR
2115  007550  105777  171434          2$:   TSTB    @TPCSR
2116  007554  100375                        BPL     2$
2117  007556  112577  171430                MOVB    (R5)+,@TPDBR
2118  007562  001345                        BNE     1$
2119  007564  012605                3$:     MOV     (SP)+,R5
2120  007566  000002                        RTI
2121
```

```
                                        ;ASCII STRING INPUT ROUTINE
                                
007570  010346                .INSTR: MOV     R3,-(SP)
007572  010446                        MOV     R4,-(SP)
007574  017637  000004  007612        MOV     @4(SP),.MSG
007602  062766  000002  000004        ADD     #2,4(SP)
007610  104402              .INST1: TYPE
007612  000000              .MSG:   0
007614  012704  011744              MOV     #INBUF,R4
007620  012703  000007              MOV     #7,R3
007624  105777  171354      1$:     TSTB    @TKCSR
007630  100375                      BPL     1$
007632  117714  171350              MOVB    @TKDBR,(R4)
007636  142714  000200              BICB    #200,(R4)
007642  121427  000025              CMPB    (R4),#25                ;IS IT <^G>
007646  001003                      BNE     200$
007650  104402  011332              TYPE,MCRLF
007654  000755                      BR      .INST1
007656  122427  000015      200$:   CMPB    (R4)+,#15
007662  001423                      BEQ     INSTR2
007664  117777  171316  171320      MOVB    @TKDBR,@TPDBR
007672  105777  171312      2$:     TSTB    @TPCSR
007676  100375                      BPL     2$
007700  005303                      DEC     R3
007702  001350                      BNE     1$
007704  000402                      BR      .INSTG
007706  010346              .INSTE: MOV     R3,-(SP)
007710  010446                      MOV     R4,-(SP)
007712  104402              .!NSTG: TYPE
007714  011326                      MQM
007716  005737  011152              TST     @#RDSW
007722  001402                      BEQ     400$
007724  104402  011332              TYPE,MCRLF
007730  000727              400$:   BR      .INST1
007732  012604              INSTR2: MOV     (SP)+,R4
007734  012603                      MOV     (SP)+,R3
007736  000002                      RTI
                                
                                        ;CONVERT ASCII STRING TO OCTAL
                                
007740  010546              .PARAM: MOV     R5,-(SP)
007742  010446                      MOV     R4,-(SP)
007744  016605  000004              MOV     4(SP),R5
007750  012537  010144              MOV     (R5)+,LOLIM
007754  012537  010146              MOV     (R5)+,HILIM
007760  012537  010150              MOV     (R5)+,DEVADR
007764  112537  010152              MOVB    (R5)+,LOBITS
007770  112537  010153              MOVB    (R5)+,ADRCNT
007774  010566  000004              MOV     R5,4(SP)
010000  005005              PARAM1: CLR     R5
010002  012704  011744              MOV     #INBUF,R4
010006  122714  000015              CMPB    #15,(R4)
010012  001420                      BEQ     PARERR
010014  121427  000060      1$:     CMPB    (R4),#60
010020  002415                      BLT     PARERR
010022  121427  000067              CMPB    (R4),#67
```

```
2178  010026  003012                        BGT     PARERR
2179  010030  142714  000060                BICB    #60,(R4)
2180  010034  152405                        BISB    (R4)+,R5
2181  010036  122714  000015                CMPB    #15,(R4)
2182  010042  001414                        BEQ     LIMITS
2183  010044  006305                        ASL     R5
2184  010046  006305                        ASL     R5
2185  010050  006305                        ASL     R5
2186  010052  000760                        BR      1$
2187  010054  122714  000015        PARERR: CMPB    #15,(R4)                ;IS FIRST CHARACTER A <CR>
2188  010060  001003                        BNE     120$
2189  010062  005737  011152                TST     @#RDSW                  ;IS CKSWR ROUTINE BEING USED
2190  010066  001023                        BNE     PARTI
2191  010070  104404                120$:   INSTER
2192  010072  000742                        BR      PARAM1
2193
2194                                        ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
2195
2196  010074  020537  010146        LIMITS: CMP     R5,HILIM
2197  010100  101365                        BHI     PARERR
2198  010102  020537  010144                CMP     R5,LOLIM
2199  010106  103762                        BLO     PARERR
2200  010110  133705  010152                BITB    LOBITS,R5
2201  010114  001357                        BNE     PARERR
2202
2203                                        ;STORE NUMBER AT SPECIFIED ADDRESS
2204
2205  010116  013704  010150                MOV     DEVADR,R4
2206  010122  010524                1$:     MOV     R5,(R4)+
2207  010124  062705  000002                ADD     #2,R5
2208  010130  105337  010153                DECB    ADRCNT
2209  010134  001372                        BNE     1$
2210  010136  012604                PARTI:  MOV     (SP)+,R4
2211  010140  012605                        MOV     (SP)+,R5
2212  010142  000002                        RTI
2213  010144  000000                LOLIM:  0
2214  010146  000000                HILIM:  0
2215  010150  000000                DEVADR: 0
2216  010152  000000                LOBITS: 0
2217          010153                ADRCNT=LOBITS+1
2218
2219                                        ;SAVE PC OF TEST THAT FAILED AND R0-R5
2220
2221  010154  016637  000004  001274  .SAV05: MOV     4(SP),SAVPC
2222
2223                                        ;SAVE R0-R5
2224
2225  010162  010537  001270        SV05:   MOV     R5,SAVR5
2226  010166  010437  001266                MOV     R4,SAVR4
2227  010172  010337  001264                MOV     R3,SAVR3
2228  010176  010237  001262                MOV     R2,SAVR2
2229  010202  010137  001260                MOV     R1,SAVR1
2230  010206  010037  001256                MOV     R0,SAVR0
2231  010212  000002                        RTI
2232
2233                                        ;RESTORE R0-R5
```

```
2234
2235  010214  013700  001256          .RES05: MOV     SAVR0,R0
2236  010220  013701  001260                  MOV     SAVR1,R1
2237  010224  013702  001262                  MOV     SAVR2,R2
2238  010230  013703  001264                  MOV     SAVR3,R3
2239  010234  013704  001266                  MOV     SAVR4,R4
2240  010240  013705  001270                  MOV     SAVR5,R5
2241  010244  000002                         RTI
2242
2243                                         ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
2244
2245  010246  104402                 .CONVR: TYPE
2246  010250  011332                         MCRLF
2247  010252  010046                 .CNVRT: MOV     R0,-(SP)
2248  010254  010146                         MOV     R1,-(SP)
2249  010256  010346                         MOV     R3,-(SP)
2250  010260  010446                         MOV     R4,-(SP)
2251  010262  010546                         MOV     R5,-(SP)
2252  010264  017601  000012                 MOV     @12(SP),R1
2253  010270  013737  012006  001250         MOV     TEMP,TEMP3
2254  010276  062766  000002  000012         ADD     #2,12(SP)
2255  010304  012137  010466                 MOV     (R1)+,WRDCNT
2256  010310  112137  010470          1$:    MOVB    (R1)+,CHRCNT
2257  010314  112137  010471                 MOVB    (R1)+,SPACNT
2258  010320  013137  010472                 MOV     @(R1)+,BINWRD
2259  010324  013704  010472          2$:    MOV     BINWRD,R4
2260  010330  113705  010470                 MOVB    CHRCNT,R5
2261  010334  012700  012006                 MOV     #TEMP,R0
2262  010340  010403                  3$:    MOV     R4,R3
2263  010342  042703  177770                 BIC     #177770,R3
2264  010346  062703  000060                 ADD     #060,R3
2265  010352  110320                         MOVB    R3,(R0)+
2266  010354  000241                         CLC
2267  010356  006004                         ROR     R4
2268  010360  000241                         CLC
2269  010362  006004                         ROR     R4
2270  010364  000241                         CLC
2271  010366  006004                         ROR     R4
2272  010370  005305                         DEC     R5
2273  010372  001362                         BNE     3$
2274  010374  012703  012050                 MOV     #MDATA,R3
2275  010400  114023                  4$:    MOVB    -(R0),(R3)+
2276  010402  105337  010470                 DECB    CHRCNT
2277  010406  001374                         BNE     4$
2278  010410  105737  010471                 TSTB    SPACNT
2279  010414  001405                         BEQ     6$
2280  010416  112723  000040          5$:    MOVB    #040,(R3)+
2281  010422  105337  010471                 DECB    SPACNT
2282  010426  001373                         BNE     5$
2283  010430  105013                  6$:    CLRB    (R3)
2284  010432  104402                         TYPE
2285  010434  012050                         MDATA
2286  010436  005337  010466                 DEC     WRDCNT
2287  010442  001322                         BNE     1$
2288  010444  013737  001250  012006         MOV     TEMP3,TEMP
2289  010452  012605                         MOV     (SP)+,R5
```

```
2290  010454  012604                        MOV     (SP)+,R4
2291  010456  012603                        MOV     (SP)+,R3
2292  010460  012601                        MOV     (SP)+,R1
2293  010462  012600                        MOV     (SP)+,R0
2294  010464  000002                        RTI
2295  010466  000000                WRDCNT: 0
2296  010470  000000                CHRCNT: 0
2297          010471                SPACNT=CHRCNT+1
2298  010472  000000                BINWRD: 0
2299                                        ;TRAP DISPATCH SERVICE
2300                                        ;ARGUMENT OF TRAP IS EXTRACTED
2301                                        ;AND USED AS OFFSET TO OBTAIN POINTER
2302                                        ;TO SELECTED SUBROUTINE
2303
2304  010474  011646                .TRPSR: MOV     (SP),-(SP)              ;GET PC OF RETURN
2305  010476  162716  000002                SUB     #2,(SP)                 ;=PC OF TRAP
2306  010502  017616  000000                MOV     @(SP),(SP)              ;GET TRP
2307  010506  006316                TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
2308  010510  042716  177001                BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
2309  010514  062716  001314                ADD     #.TRPTAB,(SP)           ;POINTER TO SUBROUTINE ADDRESS
2310  010520  017616  000000                MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
2311  010524  000136                        JMP     @(SP)+                  ;GO TO SUBROUTINE
2312
2313                                        ;ERROR HANDLER
2314
2315  010526  104414                .HLT:   CKSWR
2316  010530  032777  010000  170442        BIT     #SW12,@SWR
2317  010536  001406                        BEQ     XBX
2318  010540  105777  170444                TSTB    @TPCSR
2319  010544  100003                        BPL     XBX
2320  010546  112777  000207  170436        MOVB    #207,@TPDBR
2321  010554  032777  020000  170416 XBX:    BIT     #SW13,@SWR
2322  010562  001074                        BNE     HALTS
2323  010564  021637  001234                CMP     (SP),LSTERR
2324  010570  001404                        BEQ     1$
2325  010572  011637  001234                MOV     (SP),LSTERR
2326  010576  105037  001312                CLRB    ERRFLG
2327  010602  104406                1$:     SAV05
2328  010604  011605                        MOV     (SP),R5
2329  010606  162705  000002                SUB     #2,R5
2330  010612  011504                        MOV     (R5),R4
2331  010614  006304                        ASL     R4
2332  010616  061504                        ADD     (R5),R4
2333  010620  006304                        ASL     R4
2334  010622  042704  177001                BIC     #177001,R4
2335  010626  062704  012760                ADD     #.ERRTAB,R4
2336  010632  012437  010724                MOV     (R4)+,ERRMSG
2337  010636  012437  010736                MOV     (R4)+,DATAHD
2338  010642  011437  010750                MOV     (R4),DATABP
2339  010646  105737  001312                TSTB    ERRFLG
2340  010652  001403                        BEQ     TYPMSG
2341  010654  005737  010750                TST     DATABP
2342  010660  001027                        BNE     TYPDAT
2343  010662  104402                TYPMSG: TYPE
2344  010664  011611                        MTSTN
2345  010666  104411                        CNVRT
```

```
2346  010670  011050                          XTSTN
2347  010672  104402                          TYPE
2348  010674  011677                          MERRPC
2349  010676  104411                          CNVRT
2350  010700  011042                          ERTABO
2351  010702  104402                          TYPE
2352  010704  011332                          MCRLF
2353  010706  112737  177777  001312          MOVB    #-1,ERRFLG
2354  010714  005737  010724                  TST     ERRMSG
2355  010720  001402                          BEQ     WRKO.FM
2356  010722  104402                          TYPE
2357  010724  000000              ERRMSG:     0
2358  010726                      WRKO.FM:
2359  010726  005737  010736                  TST     DATAHD
2360  010732  001402                          BEQ     TYPDAT
2361  010734  104402                          TYPE
2362  010736  000000              DATAHD:     0
2363  010740  005737  010750      TYPDAT:     TST     DATABP
2364  010744  001402                          BEQ     RESREG
2365  010746  104410                          CONVRT
2366  010750  000000              DATABP:     0
2367  010752  104407              RESREG:     RES05
2368  010754  005777  170220      HALTS:      TST     @SWR
2369  010760  100005                          BPL     EXITER
2370  010762  010046                          PUSHR0
2371  010764  016600  000002                  MOV     2(SP),R0
2372  010770  000000                          HALT
2373  010772  012600                          POPR0
2374  010774  104414              EXITER:     CKSWR
2375  010776  005237  001232                  INC     ERRCNT
2376  011002  032777  000400  170170          BIT     #SW08,@SWR
2377  011010  001007                          BNE     1$
2378  011012  032777  002000  170160          BIT     #SW10,@SWR
2379  011020  001407                          BEQ     2$
2380  011022  013737  001216  001214          MOV     NEXT,RETURN
2381  011030  012706  001200      1$:         MOV     #STACK,SP
2382  011034  000177  170154                  JMP     @RETURN
2383  011040  000002              2$:         RTI
2384  011042  000001              ERTABO:     1
2385  011044     006     002                  .BYTE   6,2
2386  011046  001274                          SAVPC
2387  011050  000001              XTSTN:      1
2388  011052     003     002                  .BYTE   3,2
2389  011054  001226                          TSTNO
2390                                          ;ENTER HERE ON POWER FAILURE
2391
2392
2393  011056                      .PFAIL:
2394  011056  012737  011070  000024          MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
2395  011064  000000                          HALT                            ;HALT ON POWER DOWN NORMAL
2396  011066  000777                          BR      .
2397
2398                                          ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
2399
2400  011070                      RESTAR:
2401  011070  012737  011056  000024          MOV     #.PFAIL,24              ;SET UP FOR POWER FAILURE
```

```
2402  011076  012706  001200                  MOV     #STACK,SP
2403  011102  005037  012006                  CLR     TEMP
2404  011106  005237  012006                  INC     TEMP
2405  011112  001375                          BNE     .-4
2406  011114  104402                          TYPE
2407  011116  011334                          MPFAIL
2408  011120  104411                          CNVRT
2409  011122  011144                          PFTAB
2410  011124  005037  001312                  CLR     ERRFLG
2411  011130  005037  001234                  CLR     LSTERR
2412  011134  104412                          MSTCLR
2413  011136  104413                          MEMCLR
2414  011140  000177  170050                  JMP     @RETURN
2415  011144  000001                  PFTAB:  1
2416  011146     003     002          .BYTE   3,2
2417  011150  001226                          TSTNO
2418
2419
2420                                          ;CHECK SWITCH REGISTER ROUTINE. CHECKS FOR ^G TO ALLOW CHANGING
2421                                          ;OF LOC.176.
2422                                          ;LOCATIONS USED:
2423  011152  000000                  RDSW:   .WORD   0
2424
2425
2426  011154  005737  000042          .CKSWR: TST     @#42
2427  011160  001042                          BNE     OUT
2428  011162  022737  000176  001200          CMP     #SWREG,SWR      ;SOFTWARE SWITCH REGISTER PRESENT
2429  011170  001036                          BNE     OUT             ;NO, GET OUT
2430  011172  105777  170006                  TSTB    @TKCSR          ;YES,WAIT FOR
2431  011176  100033                          BPL     OUT             ;READY, GET CHARACTER
2432  011200  017737  170002  007612          MOV     @TKDBR,.MSG     ;AND STRIP OFF
2433  011206  042737  177600  007612          BIC     #177600,.MSG    ;THE GARBAGE
2434  011214  122737  000007  007612          CMPB    #7,.MSG         ;IS IT A <^G>
2435  011222  001021                          BNE     OUT
2436  011224  104402  011302                  TYPE,$CNTG
2437  011230  005137  011152          .CNTLU: COM     @#RDSW
2438  011234  104402  011306                  TYPE,$MSWR
2439  011240  104411  011274                  CNVRT,SWREGC
2440  011244  104403  011315                  INSTR,$MNEW
2441  011250  104405                          PARAM
2442  011252  000000                          0
2443  011254  177777                          177777
2444  011256  000176                          SWREG
2445  011260     000     001          .BYTE           0,1
2446  011262  104402  011332                  TYPE,MCRLF
2447  011266  005037  011152          OUT:    CLR     @#RDSW
2448  011272  000002                          RTI
2449  011274  000001                  SWREGC: 1
2450  011276     006     002          .BYTE   6,2
2451  011300  000176                          SWREG
2452  011302  057377  000107          $CNTG:  .ASCIZ  <377>/^G/
2453  011306  051777  051127  020075  $MSWR:  .ASCIZ  <377>/SWR= /
2454  011314     000
2455  011315     040  047040  053505  $MNEW:  .ASCIZ  /  NEW= /
2456  011322  020075     000
2457          011326                  .EVEN
```

```
2458  011326  020040  000077          MQM:    .ASCIZ  /  ?/
2459  011332  000377                  MCRLF:  .ASCIZ   <377>
2460  011334  050377  051127  043040  MPFAIL: .ASCIZ  <377>/PWR FAILED. RESTART AT TEST /
2461  011342  044501  042514  027104
2462  011350  051040  051505  040524
2463  011356  052122  040440  020124
2464  011364  042524  052123  000040
2465  011372  042777  042116  050040  MEPASS: .ASCIZ  <377>/END PASS CZDQB  /
2466  011400  051501  020123  055103
2467  011406  050504  020102  000040
2468  011414  051377     000          MR:     .ASCIZ  <377>/R/
2469  011417     377  051120  043517  MERR2:  .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
2470  011424  040522  020115  047111
2471  011432  044504  040503  042524
2472  011440  020123  047516  042040
2473  011446  053105  041511  051505
2474  011454  050040  042522  042523
2475  011462  052116  000056
2476  011466  044777  051516  043125  MERR3:  .ASCIZ  <377>/INSUFFICIENT DATA!/
2477  011474  044506  044503  047105
2478  011502  020124  040504  040524
2479  011510  000041
2480  011512  052377  051505  020124  MTSTPC: .ASCIZ  <377>/TEST PC-/
2481  011520  041520  000055
2482  011524  046377  041517  020113  MLOCK:  .ASCIZ  <377>/LOCK ON SELECTED TEST/
2483  011532  047117  051440  046105
2484  011540  041505  042524  020104
2485  011546  042524  052123     000
2486  011553     103  051123  020072  MCSRX:  .ASCIZ   /CSR: /
2487  011560     000
2488  011561     126  041505  020072  MVECX:  .ASCIZ  /VEC: /
2489  011566     000
2490  011567     120  051501  042523  MPASSX: .ASCIZ  /PASSES: /
2491  011574  035123  000040
2492  011600  051105  047522  051522  MERRX:  .ASCIZ  /ERRORS: /
2493  011606  020072     000
2494  011611     377  052377  051505  MTSTN:  .ASCIZ  <377><377> /TEST NO: /
2495  011616  020124  047516  020072
2496  011624     000
2497  011625     377  042523  020124  MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DQ11'S DESIRED ACTIVE./
2498  011632  053523  052111  044103
2499  011640  051040  043505  052040
2500  011646  020117  050504  030461
2501  011654  051447  042040  051505
2502  011662  051111  042105  040440
2503  011670  052103  053111  027105
2504  011676     000
2505  011677     120  035103  000040  MERRPC: .ASCIZ  /PC: /
2506  011704  046777  050101  047440  XHEAD:  .ASCIZ  <377>/MAP OF DQ11 STATUS/<377>
2507  011712  020106  050504  030461
2508  011720  051440  040524  052524
2509  011726  177523     000
2510          011732                  .EVEN
2511  011732  000002                  XSTATQ: 2
2512  011734     006     003                  .BYTE   6,3
2513  011736  001244                          TEMP1
```

```
2514  011740  006     002                 .BYTE   6,2
2515  011742  001246                      TEMP2
2516                              .EVEN
2517
2518                                      ;BUFFERS FOR INPUT-OUTPUT
2519
2520  011744  000000                      INBUF:  0
2521          012006                      .=.+40
2522  012006  000000                      TEMP:   0
2523          012050                      .=.+40
2524  012050  000000                      MDATA:  0
2525          012112                      .=.+40
```

```
2526
2527  012112  000002                    .MEMCLR:        RTI
2528  012114  000002                    .MSTCLR:        RTI
2529                                              ;TABLE OF ERROR MESSAGES AND DATA HEADERS
2530
2531  012116  042515  047515  054522    EM0:    .ASCIZ  /MEMORY EXTENSION READ-WRITE ERROR/
      012160  042515  047515  054522    EM1:    .ASCIZ  /MEMORY EXTENSION ADDRESS ERROR/
      012217    102   051525  040440    EM2:    .ASCIZ  /BUS ADR-CHAR COUNT MEMORY DATA ERROR/
      012264  042515  047515  054522    EM3:    .ASCIZ  /MEMORY EXTENSION DATA ERROR/
      012320  044103  051101  041501    EM4:    .ASCIZ  /CHARACTER MEMORY ADDRESS ERROR/
      012357    123   050505  042525    EM5:    .ASCIZ  /SEQUENCE MEMORY ADDRESS ERROR/
      012415    103   040510  040522    EM6:    .ASCIZ  /CHARACTER MEMORY DATA ERROR/
      012451    123   050505  042525    EM7:    .ASCIZ  /SEQUENCE MEMORY DATA ERROR/
      012504  051120  046511  051101    EM10:   .ASCIZ  /PRIMARY REGISTER MASTER CLEAR ERROR/
      012550  042523  047503  042116    EM11:   .ASCIZ  /SECONDARY REGISTER MASTER CLEAR ERROR/
      012616  051127  052111  020105    EM12:   .ASCIZ  /WRITE ENABLE NOT CLEARED/

      012647    377   054105  042520    DH0:    .ASCIZ  <377>/EXPECTED  RECEIVED  SEC ADR   SEC REG/
      012716  042777  050130  041505    DH1:    .ASCIZ  <377>/EXPECTED  RECEIVED  REG ADDRESS/
              012760                    .EVEN
2532                                              ;TABLE OF ERROR POINTERS
2533
2534  012760  012116                    .ERRTAB:EM0
2535  012762  012716                            DH1
2536  012764  013126                            DT2
2537  012766  012160                            EM1
2538  012770  012716                            DH1
2539  012772  013126                            DT2
2540  012774  012217                            EM2
2541  012776  012647                            DH0
2542  013000  013104                            DT1
2543  013002  012264                            EM3
2544  013004  012647                            DH0
2545  013006  013062                            DT0
2546  013010  012320                            EM4
2547  013012  012647                            DH0
2548  013014  013104                            DT1
2549  013016  012357                            EM5
2550  013020  012647                            DH0
2551  013022  013104                            DT1
2552  013024  012415                            EM6
2553  013026  012647                            DH0
2554  013030  013104                            DT1
2555  013032  012451                            EM7
2556  013034  012647                            DH0
2557  013036  013104                            DT1
2558  013040  012504                            EM10
2559  013042  012716                            DH1
2560  013044  013126                            DT2
2561  013046  012550                            EM11
2562  013050  012647                            DH0
2563  013052  013104                            DT1
2564  013054  012616                            EM12
2565  013056  000000                            0
2566  013060  000000                            0
```

```
2567
2568                                        ;TABLE OF DATA POINTER FOR ERROR TYPEOUT
2569
2570  013062  000004                DT0:    4
2571  013064     003      007             .BYTE   3,7
2572  013066  001270                        SAVR5
2573  013070     003      007             .BYTE   3,7
2574  013072  001266                        SAVR4
2575  013074     006      004             .BYTE   6,4
2576  013076  001372                        DQSEC
2577  013100     002      000             .BYTE   2,0
2578  013102  001264                        SAVR3
2579  013104  000004                DT1:    4
2580  013106     006      004             .BYTE   6,4
2581  013110  001270                        SAVR5
2582  013112     006      004             .BYTE   6,4
2583  013114  001266                        SAVR4
2584  013116     006      004             .BYTE   6,4
2585  013120  001372                        DQSEC
2586  013122     002      000             .BYTE   2,0
2587  013124  001262                        SAVR2
2588  013126  000003                DT2:    3
2589  013130     006      004             .BYTE   6,4
2590  013132  001270                        SAVR5
2591  013134     006      004             .BYTE   6,4
2592  013136  001266                        SAVR4
2593  013140     006      004             .BYTE   6,4
2594  013142  001264                        SAVR3
2595          000001                    .END
```

```
ABBIT = 002000            610#    724    1982
ACTBIT= 004000            611#    732
ADRCNT= 010153           2169*   2208*   2217#
BABIT = 010000            612#    728    1843
BBBIT = 020000            613#    719    1444
BINWRD  010472           2258*   2259    2298#
BIT0  = 000001            605#   1045
BIT1  = 000002            604#
BIT10 = 002000            595#
BIT11 = 004000            594#
BIT12 = 010000            593#
BIT13 = 020000            592#
BIT14 = 040000            591#   2069
BIT15 = 100000            590#
BIT2  = 000004            603#   1016
BIT3  = 000010            602#
BIT4  = 000020            601#   1103   1107   1152   1169   1170   1342   1343   1380   1381   1418   1419
BIT5  = 000040            600#   1808   1831   1861   1891   1916   1943   1969   2001
BIT6  = 000100            599#
BIT7  = 000200            598#
BIT8  = 000400            597#
BIT9  = 001000            596#
BRW     007436           1022   2087#
BRX     007440           1023   2088#
CHARDT= 000010             630#
CHAR1   001236             803#
CHAR2   001240             804#
CHAR3   001242             805#
CHKAB1  007036            1961   1982#
CHKBA1  006356            1825   1843#
CHKBA2  006460            1855   1871#
CHKBB1  004372            1410   1444#
CHRCNT  010470           2256*   2260   2276*  2296#  2297
CKSWR = 104414             864#    990   1000   1015   2050   2068   2092   2315   2374
CNTLU = 104415             866#    972
CNVRT = 104411             858#   2024   2028   2032   2036   2345   2349   2408   2439
CONVRT= 104410             856#    980   2365
CREAM   001300             820#    944*  1054*  1058*  1067   1068*  1069   1070*
CSRMAP  000220             699#    975
DATABP  010750           2338*   2341   2363   2366#
DATAHD  010736           2337*   2359   2362#
DEVADR  010150           2167*   2205   2215#
DH0     012647           2531#   2541   2544   2547   2550   2553   2556   2562
DH1     012716           2531#   2535   2538   2559
DISPRE  000174             690#    960
DLIGHT= 177570             575#    951
DQACTV  001500             670     744*   750*   751*   755    923#   997*   998    1048   1050
DQCR00  001400             891#
DQCR01  001404             893#
DQCR02  001410             895#
DQCR03  001414             897#
DQCR04  001420             899#
DQCR05  001424             901#
DQCR06  001430             903#
DQCR07  001434             905#
DQCR10  001440             907#
```

```
DQCR11  001444          909#
DQCR12  001450          911#
DQCR13  001454          913#
DQCR14  001460          915#
DQCR15  001464          917#
DQCR16  001470          919#
DQCR17  001474          921#
DQCSR   001506          926#  1067*  1071
DQERR   001366          881#  1084*  1085*  1086   1912   1914*  1919
DQNUM   001504          703*   740*   745    747    925#   939    2041
DQRCSH  001362          879#  1080*  1081*  1463*  1473*  1500*  1510*  1537*  1545*  1571*  1579*  1605*  1613*
                       1639*  1647*  1673*  1681*  1707*  1715*  1741*  1749*  1775*  1783*
DQRCSR  001360          878#  1071*  1080   1082   1804   1806*  1811   1827   1829*  1834   2055
DQREG   001370          882#  1086*  1087*  1088   1103*  1107   1122   1123*  1124   1128*  1129   1146   1147*
                       1148   1153*  1155   1170*  1171*  1172*  1181*  1182*  1183   1210*  1217*  1243*  1250*
                       1276*  1283*  1309*  1316*  1343*  1344*  1345*  1352*  1353*  1354   1381*  1382*  1383*
                       1390*  1391*  1392   1419*  1420*  1421*  1428*  1429*  1430   1462*  1472*  1499*  1509*
                       1536*  1544*  1570*  1578*  1604*  1612*  1638*  1646*  1672*  1680*  1706*  1714*  1740*
                       1748*  1774*  1782*  1807*  1830*  1860*  1890*  1915*  1939*  1942*  1944*  1965*  1968*
                       1970*  1997*  2000*  2002*
DQRLVL  001352          875#  1074*  1075*  1076
DQRVEC  001350          874#  1072*  1073*  1074   2058
DQSEC   001372          883#  1088*  1089*  1090   1106*  1154*  1173*  1207   1211*  1218   1240   1244*  1251
                       1273   1277*  1284   1306   1310*  1317   1346*  1384*  1422*  1464*  1474   1501*  1511
                       1538*  1546   1572*  1580   1606*  1614   1640*  1648   1674*  1682   1708*  1716   1742*
                       1750   1776*  1784   1808*  1831*  1861*  1891*  1916*  1941*  1943*  1947   1967*  1969*
                       1973   1999*  2001*  2005   2576   2585
DQSECH  001374          884#  1090*  1091*
DQSTAT  001510          927#  1069*  1072   1444   1843   1982
DQST00  001402          892#
DQST01  001406          894#
DQST02  001412          896#
DQST03  001416          898#
DQST04  001422          900#
DQST05  001426          902#
DQST06  001432          904#
DQST07  001436          906#
DQST10  001442          908#
DQST11  001446          910#
DQST12  001452          912#
DQST13  001456          914#
DQST14  001462          916#
DQST15  001466          918#
DQST16  001472          920#
DQST17  001476          922#
DQTCSR  001364          880#  1082*  1083*  1084   1857   1859*  1864   1887   1889*  1894
DQTLVL  001356          877#  1078*  1079*
DQTVEC  001354          876#  1076*  1077*  1078
DSWR  = 177570          574#   950
DT0     013062         2545   2570#
DT1     013104         2542   2548   2551   2554   2557   2563   2579#
DT2     013126         2536   2539   2560   2588#
EM0     012116         2531#  2534
EM1     012160         2531#  2537
EM10    012504         2531#  2558
EM11    012550         2531#  2561
```

```
EM12    012616          2531#   2564
EM2     012217          2531#   2540
EM3     012264          2531#   2543
EM4     012320          2531#   2546
EM5     012357          2531#   2549
EM6     012415          2531#   2552
EM7     012451          2531#   2555
ERRCNT  001232           798#    945*   2064    2375*
ERRFLG  001312           829#    942*   2018*   2081*   2326*   2339    2353*   2410*
ERRMSG  010724          2336*   2354    2357#
ERTABO  011042          2350    2384#
EXITER  010774          2369    2374#
HALTS   010754          2322    2368#
HILIM   010146          2166*   2196    2214#
ICOUNT  001222           794#   2079    2083*
INBUF   011744          2130    2172    2520#
INIFLG  001310           827#    966     969*
INSTER= 104404           848#   2191
INSTR = 104403           846#   1026    2440
INSTR2  007732          2141    2156#
JUMBIT= 040000           614#    731
LIGHTS  001202           783#    951*    960*    2038*
LIMITS  010074          2182    2196#
LOBITS  010152          2168*   2200    2216#   2217
LOCK    001220           793#   1205*   1238*   1271*   1304*   1335*   1373*   1411*   1457*   1494*   1531*   1565*   1599*
                        1633*   1667*   1701*   1735*   1769*   2095
LOGICA  007254           652    2045#
LOKFLG  001313           830#
LOLIM   010144          2165*   2198    2213#
LPCNT   001224           795#   2078*   2079    2082*
LSTERR  001234           799#    946*   2017*   2323    2325*   2411*
MCRLF   011332          2138    2154    2246    2352    2446    2459#
MCSRX   011553          2023    2486#
MDATA   012050          2274    2285    2524#
MEMCLR= 104413           862#   2413
MEPASS  011372          2021    2465#
MERRPC  011677          2348    2505#
MERRX   011600          2035    2492#
MERR2   011417           762    1062    2469#
MERR3   011466           994    2476#
MISC. = 000012           632#
MLOCK   011524          1018    2482#
MNEW    011625           987    2497#
MPASSX  011567          2031    2490#
MPFAIL  011334          2407    2460#
MQM     011326          2151    2458#
MR      011414          1036    2468#
MSTCLR= 104412           860#   2412
MTITLE  001000           771#    968
MTSTN   011611          2344    2494#
MTSTPC  011512          1027    2480#
MVECX   011561          2027    2488#
NEXT    001216           792#   1042*   1102*   1121*   1145*   1204*   1237*   1270*   1303*   1334*   1372*   1410*   1456*
                        1493*   1530*   1564*   1598*   1632*   1666*   1700*   1734*   1768*   1802*   1825*   1855*   1885*
                        1910*   1935*   1961*   1993*   2084    2380
ODDBIT= 001000           615#    733
```

```
OPT2X   002644          1098#
OUT     011266          2427    2429    2431    2435    2447#
PARAM = 104405           850#   1028    2441
PARAM1  010000          2171#   2192
PARERR  010054          2174    2176    2178    2187#   2197    2199    2201
PARTI   010136          2190    2210#
PASCNT  001230           797#    941*   2019*   2038    2061
PFTAB   011144          2409    2415#
POLY.  = 000017           637#
POPR0  = 012600           584#   2373
POP1SP= 005726            582#
POP2SP= 022626            586#
PS     = 177776           576#    673*    936*   1011*   1803*   1826*   1856*   1886*   1911*   1936*   1962*   1994*
PUSHR0= 010046            583#   2370
PUSH1S= 005746            581#
PUSH2S= 024646            585#
RDSW    011152          2103    2152    2189    2423#   2437*   2447*
RESREG  010752          2364    2367#
RESTAR  011070          2394    2400#
RESTRT  007264          2040    2043    2050#
RES05 = 104407           854#   2367
RETURN  001214           791#    948*   1031    1035*   1037    1041*   2051*   2084*   2085    2380*   2382    2414
RUN     001304           822#   1045*   1048    1053*   1059*   1066*
RUNCNT  001306           823#   1046*   1055*   1057*
RUNFLG  001302           821#    943*   1043    1047*
RXBA.P= 000000           621#
RXBA.S= 000004           625#
RXWC.P= 000001           622#
RXWC.S= 000005           626#
RX.BCC= 000015           635#
SAVACT  001502           755*    924#    991
SAVNUM  001276           748*    819#    939*   2039*   2041*
SAVPC   001274           818#   2221*   2386
SAVR0   001256           811#   2230*   2235
SAVR1   001260           812#   2229*   2236
SAVR2   001262           813#   2228*   2237    2587
SAVR3   001264           814#   2227*   2238    2578    2594
SAVR4   001266           815#   2226*   2239    2574    2583    2592
SAVR5   001270           816#   2225*   2240    2572    2581    2590
SAVSP   001272           817#
SAV05 = 104406           852#   2327
SCOPE = 104400           840#   1110    1134    1161    1193    1226    1259    1292    1325    1363    1401    1439    1483
                        1520    1554    1588    1622    1656    1690    1724    1758    1792    1815    1838    1870    1900
                        1925    1951    1977    2009
SCOP1 = 104401           842#   1188    1222    1255    1288    1321    1359    1397    1435    1478    1515    1550    1584
                        1618    1652    1686    1720    1754    1788
SEQ.   = 000014           634#
SPACNT= 010471          2257*   2278    2281*   2297#
STACK  = 001200           577#    937    1012    2381    2402
STFLG   001311            828#    940*
SV05    010162          2225#
SWR     001200           782#    950*    955     959*    970     973     984     991     997    1016    1024    2069    2076
                        2093    2105    2316    2321    2368    2376    2378    2428
SWREG   000176           691#    959     970    2428    2444    2451
SWREGC  011274          2439    2449#
SW00   = 000001           557#    984
```

```
SW01  = 000002          556#   1024
SW02  = 000004          555#
SW03  = 000010          554#
SW04  = 000020          553#
SW05  = 000040          552#
SW06  = 000100          551#
SW08  = 000400          550#   2376
SW09  = 001000          549#   2093
SW10  = 002000          548#   2378
SW11  = 004000          547#   2076
SW12  = 010000          546#   2105   2316
SW13  = 020000          545#   2321
SW14  = 040000          544#
SW15  = 100000          543#
SYNBIT= 100000          616#    713
SYNC. = 000011          631#
TEMP    012006         2253    2261   2288*  2403*  2404*  2522#
TEMP1   001244          670*    671*    806#   977*   978    982*  2513
TEMP2   001246          807#    978*   2515
TEMP3   001250          808#   2253*   2288
TEMP4   001252          809#
TEMP5   001254          810#
TKCSR   001204          784#   2072    2132   2430
TKDBR   001206          785#   2074    2134   2142   2432
TLAST = 007046         1030    2011#
TPCSR   001210          786#   2109    2112   2115   2143   2318
TPDBR   001212          787#   2111*   2114*  2117*  2142*  2320*
TRPOK   010506         2307#
TSTNO   001226          796#    947*   1040*  1101*  1120*  1144*  1203*  1236*  1269*  1302*  1333*  1371*  1409*
                       1455*   1492*  1529*  1563*  1597*  1631*  1665*  1699*  1733*  1767*  1801*  1824*  1854*
                       1884*   1909*  1934*  1960*  1992*  2389   2417
TST1    002254         1029    1035   1040#  2051   2052
TST10   003566         1270    1302#
TST11   003702         1303    1333#
TST12   004052         1334    1371#
TST13   004222         1372    1409#
TST14   004406         1445    1455#
TST15   004546         1456    1492#
TST16   004706         1493    1529#
TST17   005036         1530    1563#
TST2    002644         1041    1042   1101#
TST20   005166         1564    1597#
TST21   005316         1598    1631#
TST22   005446         1632    1665#
TST23   005576         1666    1699#
TST24   005726         1700    1733#
TST25   006056         1734    1767#
TST26   006206         1446    1768   1801#
TST27   006272         1802    1824#
TST3    002706         1102    1120#
TST30   006366         1854#
TST31   006462         1844    1884#
TST32   006554         1871    1885   1909#
TST33   006646         1910    1934#
TST34   006742         1935    1960#
TST35   007046         1992#   2011
```

CROSS REFERENCE TABLE -- USER SYMBOLS

```
TST36 = ****** U          1983    1993
TST4    002774            1121    1144#
TST5    003222            1145    1203#
TST6    003336            1204    1236#
TST7    003452            1237    1269#
TTST    007340            1019*   1020*   1022*   1023*   2070#
TXBA.P= 000002             623#
TXBA.S= 000006             627#
TXWC.P= 000003             624#
TXWC.S= 000007             628#
TX.BCC= 000016             636#
TX.MUX= 000013             633#
TYPDAT  010740            2342    2360    2363#
TYPE  = 104402             761     844#    968     976     986     993    1018    1036    1061    2020    2022    2026    2030
                          2034    2128    2138    2150    2154    2245    2284    2343    2347    2351    2356    2361    2406
                          2436    2438    2446
TYPMSG  010662            2340    2343#
VECMAP  000056             663#    760
WRDCNT  010466            2255*   2286*   2295#
WRKO.F  010726            2355    2358#
XBX     010554            2317    2319    2321#
XCSR    007300            2025    2053#
XERR    007322            2037    2062#
XHEAD   011704             976    2506#
XPASS   007314            2033    2059#
XSTATQ  011732             981    2511#
XTSTN   011050            2346    2387#
XVEC    007306            2029    2056#
$CNTG   011302            2436    2452#
$E    = 000037               1#    1042    1043#   1102    1103#   1121    1122#   1145    1146#   1204    1206#   1237    1239#
                          1270    1272#   1303    1305#   1334    1336#   1372    1374#   1412#   1456    1458#   1493    1495#
                          1530    1532#   1564    1566#   1598    1600#   1632    1634#   1666    1668#   1700    1702#   1734
                          1736#   1768    1770#   1802    1803#   1826#   1856#   1885    1886#   1910    1911#   1935    1936#
                          1962#   1993    1994#
$MNEW   011315            2440    2455#
$MSWR   011306            2438    2453#
$N    = 000035               1#    1038    1043#   1099    1103#   1118    1122#   1142    1146#   1201    1206#   1234    1239#
                          1267    1272#   1300    1305#   1331    1336#   1369    1374#   1407    1412#   1440    1453    1458#
                          1490    1495#   1527    1532#   1561    1566#   1595    1600#   1629    1634#   1663    1668#   1697
                          1702#   1731    1736#   1765    1770#   1799    1803#   1822    1826#   1839    1852    1856#   1871
                          1872    1882    1886#   1907    1911#   1932    1936#   1958    1962#   1978    1990    1994#   2011#
$N1   = 000035            1440#   1839#   1871#   1872#   1978#
$N2   = 000036            1440#   1839#   1978#
$N3   = 000026            1440#
$Y    = 000016               1#    831#    840     842#    844#    846#    848#    850#    852#    854#    856#    858#    860#
                           862#    864#    866#    868#
.     = 013144             641#    642     644#    651#    653#    661#    677     689#    695#    698#    702     718     723
                           727     730     765     770#    779#    820#    890#    891#    892#    893#    894#    895#    896#
                           897#    898#    899#    900#    901#    902#    903#    904#    905#    906#    907#    908#    909#
                           910#    911#    912#    913#    914#    915#    916#    917#    918#    919#    920#    921#    922#
                           923#    924#    925#    926#    927#    974     979     983     996     1064    2110    2113    2396
                          2405    2457#   2510#   2521#   2523#   2525#   2531#
.BEGIN  002116            1011#
.CKSWR  011154             865    2426#
.CNTLU  011230             867    2437#
.CNVRT  010252             859    2247#
```

```
.CONVR  010246            857    2245#
.EOP    007142           1983    1993    2017#
.ERRTA  012760           2335    2534#
.HLT    010526            647    2315#
.INSTE  007706            849    2148#
.INSTG  007712           2147    2150#
.INSTR  007570            847    2124#
.INST1  007610           2128#   2139    2155
.MEMCL  012112            863    2527#
.MSG    007612           2126*   2129#   2432*   2433*   2434
.MSTCL  012114            861    2528#
.PARAM  007740            851    2162#
.PFAIL  011056            645     938    2393#   2401
.RES05  010214            855    2235#
.SAV05  010154            853    2221#
.SCOPE  007330            841    2068#
.SCOP1  007442            843    2092#
.START  001512            696     936#    948
.TRPSR  010474            649    2304#
.TRPTA  001314           839#    2309
.TYPE   007462            845    2100#

. ABS.  013144     000

 ERRORS DETECTED:  0

 DSKZ:CZDQBD,DSKZ:CZDQBD.SEQ=DSKZ:CZDQXX.MAC,DSKZ:CZDQBD.P11
 RUN-TIME: 7 9 .7 SECONDS
 RUN-TIME RATIO: 51/18=2.8
 CORE USED:  20K  (39 PAGES)

 DOCUMENT PAGES:  59
```