

IDENTIFICATION

PRODUCT CODE: AC-8622E-MC
PRODUCT NAME: CZDQEE0 MISC & BCC TST
DATE: JUNE 1978
MAINTAINER: DIAGNOSTIC GROUP

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

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1974, 1978 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

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 INTERRUPT 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 DISCUSSED LATER.
1. CZDQO [REV] ONLINE TEST. (ITEP OVERLAY)

AND A PARAMETER INPUT PROGRAM IS AVAILABLE

1. CZDQG [REV] DQ11 TRIAL PROGRAM (PARAMETER INPUT) REQUIREMENTS

2.1 EQUIPMENT

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

2.2 STORAGE

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

3. LOADING PROCEEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND

ARE LOADED USING THE ABSOLUTE LOADER.

ABSOLUTE LOADER STARTING ADDRESS *500

MEMORY *
SIZE

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

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

3.1.2 THEN START

4. STARTING PROCEEDURE

A. LOAD LOC. 200

B. SET SWR TO ZERO FOR "AUTO SIZING" OR LEAVE
LEAVE SWR BIT 7=1 TO USE EXISTING PARAMETERS SET UP
BY DQ11 TRIAL PROGRAM OR A PREVIOUSLY RUN DQ11 DIAGNOSTIC
THAT USED THE "AUTO SIZING".

****REFER TO SECTION 4.1 FOR SOFTWARE SWITCH REGISTER OPERATION
AND OPTIONS.****

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

C.THEN START

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

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

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

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

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

4.1 CONTROL SWITCH SETTINGS

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

CONTROL:

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

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

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

4.1.2 SWITCH REGISTER RESTRICTIONS

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

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

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

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

4.1.3 SWITCH REGISTER PRIORITYS

ERROR SWITCHES

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

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

SCOPE SWITCHES

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

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

4.2 STARTING ADDRESS

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

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

5. OPERATING PROCEDURE

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

AND PROGRAM WILL BEGIN RUNNING THE
DIAGNOSTIC

5.2 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

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

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

6. ERRORS

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

6.2 ERROR RECOVERY

IF FOR SOME REASON THE DQ11 SHOULD
"HANG THE BUS" (GAIN CONTROL OF BUS SO THAT
CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT
OR POWER DOWN/UP IS NECESSARY FOR OPERATOR
TO REGAIN CONTROL OF CPU.
IF THIS SHOULD HAPPEN; LOOK IN LOCATION
'TSTNO' (ADDRESS 1226) FOR THE NUMBER OF THE TEST THAT
WAS RUNNING AT THE TIME OF THE CATASTROPHIC
ERROR.
IN THIS WAY THE OPERATOR WILL HAVE AN IDEA AS TO
WHAT THE DQ11 WAS DOING AT THE TIME OF THE ERROR.

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

IF THE SOFTWARE SWITCH REGISTER IS TO BE CHANGED AFTER A HALT
THE 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-8622E-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 PERFORMED.

TSTNO (1226) CONTAINS THE NUMBER OF THE TEST NOW
BEING PERFORMED.

RUN (1304) THE BIT IN "RUN" ALWAYS POINTS ONE
PAST THE DQ11 CURRENTLY BEING TESTED.

EXAMPLE:

(RUN) 1304/0000000001000000

MEANS THAT DQ11 NO.05 IS THE DQ11 NOW
RUNNING.

DQCROO-DQCR17
DQSTOO-DQST17
(1400)-(1476)

THESE LOCATIONS CONTAIN THE INFORMATION
NEEDED TO TEST UP TO 16 (DECIMAL) DQ11S
SEQUENTIALLY. THEY CONTAIN THE CSR, VECTOR
AND STATUS CONCERNING THE CONFIGURATION
OF EACH DQ11.

DQACTV (1500) EACH BIT SET IN THIS LOCATION INDICATES
THAT THE ASSOCIATED DQ11 WILL BE TESTED
IN TURN.

EXAMPLE:

(DQACTV) 1500/000000000011111

MEANS THAT DQ11 NO. 00,01,02,03,04
WILL BE TESTED.

EXAMPLE:

(DQACTV) 1500/000000000010001

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 APPRIQATE 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 APPRIQATE DQSTXX: AFTER AUTC SIZING

8.5.8 SET FOR ODD OR EVEN PARITY?

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

8.5.9 FINDING THE VECTOR.

THE PROGRAM SETS 'PRIMARY DONE', 'SECONDARY DONE', AND 'INTERUPT ENABLE' AND LOOKS FOR AN INTERUPT. IF IT INTERUPTS IT IS PICKED UP AND STORED AWAY. IF NO INTERUPT OCCURS 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

```
522 .ENABLE AMA
523
524 ;CZDQEE0/<377>/ MISC & BCC TST
525 ;COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
526
527 ;REVISED 16-DEC-76 BY R. BLACK
528 ; A)SUPPORTS SOFTWARE SWITCH REGISTER
529 ; B)SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER
530 ; BY <^G>.
531 ;STARTING PROCEDURE
532 ;LOAD PROGRAM
533 ;LOAD ADDRESS 000200
534 ;PRESS START
535 ;PROGRAM WILL TYPE 'CZDQEE0/<377>/ MISC & BCC TST'
536 ;PROGRAM WILL TYPE 'R' TO INDICATE THAT TESTING HAS STARTED
537 ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
538 ;AND THEN RESUME TESTING
539
540
541 ;SWITCH REGISTER OPTIONS
542
543 100000 SW15=100000 :=1,HALT ON ERROR
544 040000 SW14=40000 :=1,LOOP ON CURRENT TEST
545 020000 SW13=20000 :=1,INHIBIT ERROR TYPEOUT
546 010000 SW12=10000 :=1,DELETE TYPEOUT/BELL ON ERROR.
547 004000 SW11=4000 :=1,INHIBIT ITERATIONS
548 002000 SW10=2000 :=1,ESCAPE TO NEXT TEST ON ERROR
549 001000 SW09=1000 :=1,LOOP WITH CURRENT DATA
550 000400 SW08=400 :=1,LOOP ON ERROR
551 000100 SW06=100
552 000040 SW05=40
553 000020 SW04=20
554 000010 SW03=10
555 000004 SW02=4 ;LOCK ON TEST SELECT
556 000002 SW01=2 ;RESTART PROGRAM AT SELECTED TEST
557 000001 SW00=1 ;RESELECT DQ11 DESIRED ACTIVE
558 ;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT
```

GENERAL DEFINATIONS AND EQUIVALENCIES

```
559
560
561      ;REGISTER DEFINITIONS
562
563      000000      R0=%0      ;GENERAL REGISTER
564      000001      R1=%1      ;GENERAL REGISTER
565      000002      R2=%2      ;GENERAL REGISTER
566      000003      R3=%3      ;GENERAL REGISTER
567      000004      R4=%4      ;GENERAL REGISTER
568      000005      R5=%5      ;GENERAL REGISTER
569      000006      SP=%6      ;PROCESSOR STACK POINTER
570      000007      PC=%7      ;PROGRAM COUNTER
571
572      ;LOCATION EQUIVALENCIES
573
574      177570      DSWR= 177570 ;HARDWARE SWITCH REGISTER LOC.
575      177570      DLIGHTS=177570 ;HARDWARE DISPLAY REGISTER LOC.
576      177776      PS=177776 ;PROCESSOR STATUS WORD
577      001200      STACK=1200 ;START OF PROCESSOR STACK
578
579      ;INSTRUCTION DEFINITIONS
580
581      005746      PUSH1SP=5746 ;DECREMENT PROCESSOR STACK 1 WORD
582      005726      POP1SP=5726 ;INCREMENT PROCESSOR STACK 1 WORD
583      010046      PUSHRO=10046 ;SAVE R0 ON STACK
584      012600      POPRO=12600 ;RESTORE R0 FROM STACK
585      024646      PUSH2SP=24646 ;DECREMENT STACK TWICE
586      022626      POP2SP=22626 ;INCREMENT STACK TWICE
587      .EQUIV EMT,HLT ;BASIC DEFINITION OF ERROR CALL
588
589
590      100000      BIT15=100000
591      040000      BIT14=40000
592      020000      BIT13=20000
593      010000      BIT12=10000
594      004000      BIT11=4000
595      002000      BIT10=2000
596      001000      BIT9=1000
597      000400      BIT8=400
598      000200      BIT7=200
599      000100      BIT6=100
600      000040      BIT5=40
601      000020      BIT4=20
602      000010      BIT3=10
603      000004      BIT2=4
604      000002      BIT1=2
605      000001      BIT0=1
606
607
608      ;DQ11 OPTIONAL DEFINITIONS
609
610      002000      ABBIT=2000
611      004000      ACTBIT=4000
612      010000      BABIT=10000
613      020000      BBBIT=20000
614      040000      JUMBIT=40000
```

GENERAL DEFINATIONS AND EQUIVALENCIES

615 001000 ODDBIT=1000
616 100000 SYNBIT=100000

:DQ11 SECONDARY REGISTER DEFINATIONS

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			

TRAPCATCHER FOR UNEXPECTED INTERRUPTS

```

640                                     :TRAPCATCAER FOR ILLEGAL INTERRUPTS
641         000000         .=0
642                                     :STANDARD INTERRUPT VECTORS
643
644         000024         000024         .=24
645         000024         014300         .PFAIL           :POWER FAIL HANDLER
646         000026         000340         340             :SERVICE AT LEVEL 7
647         000030         013750         .HLT             :ERROR HANDLER
648         000032         000340         340             :SERVICE AT LEVEL 7
649         000034         013716         .TRPSRV          :GENERAL HANDLER DISPATCH SERVICE
650         000036         000340         340             :SERVICE AT LEVEL 7
651
652         000046         012476         .=46
653                                     LOGICAL           :ACT HOOKS
654         000052         000000         .=52
655         .WORD 0
656         :THIS ROUTINE TRIES TO FORCE THE RECEIVER TO INTERRUPT
657         :TO ITS VECTOR WHERE IT WILL PICK UP THE STATUS LOCATION
658         :FOR ITS NEW PC; AND PICK UP AN IOT INSTRUCTION FOR ITS
659         :NEW PS. WHEN THE NEW PC IS FETCHED AN IOT INSTRUCTION IS
660         :EXECUTED, TRAPPING TO LOCATION 20 WHERE A ROUTINE IS EXECUTED
661         :TO TAKE THE PC FROM THE STACK AND US IT AS THE VECTOR ADDRESS
662         000056         .=56
663
664         000056         010120         VECMAP:
665         000060         012721         000004         1$:  MOV      R1,(R0)+      :START FILLING THE VECTOR AREA
666         000064         022021         001000         MOV      #4,(R1)+      :WITH +2: IOT (4)
667         000066         020127         001000         CMP      (R0)+,(R1)+  :UPDATE THE POINTERS
668         000072         101771         BLOS     1$           :IS ALL FLOATING VECTOR AREA DONE
669         000074         012737         000146         000020     MOV      #4$,@#20      :BR IF NOT ALL DONE
670         000102         013737         001500         001244     MOV      DQACTV,TEMP1  :SET FOR IOT TRAP BY DQ11
671         000110         006037         001244         2$:  ROR      TEMP1      :GET THE ACTIVE DQ11 S
672         000114         103023         BCC      5$           :ARE YOU ACTIVE.. DQ11
673         000116         005037         177776         CLR      PS           :IF CARRY CLEAR.. NO MORE DQ11S
674         000122         005722         TST      (R2)+        :CLEAR PS
675         000124         012772         000340         177776     MOV      #340,@-2(R2)  :PUT POINTER TO STATUS TABLE
676         000132         105200         INCB     R0           :TRY AND SET PRI/SEC DONE AND IE
677         000134         001376         BNE     -2            :DELAY.....
678         000136         112712         000300         MOVB     #300,(R2)    :.....DELAY
679         000142         005722         3$:  TST      (R2)+        :NO INTERRUPT ASSUME 300 FIX IN TEST C
680         000144         000761         BR       2$           :UPDATE POINTERS
681         000146         051612         4$:  BIS      (SP),(R2)    :GO DO IT AGAIN
682         000150         042712         000007         BIC      #7,(R2)     :ENTERD BY IOT TRAP BY DQ11
683         000154         022626         CMP      (SP)+,(SP)+  :CLEAR UNWANTED BITS
684         000156         012716         000142         MOV      #3$,(SP)    :POP IOT JUNK OFF STACK
685         000162         000002         RTI     PC           :SET RETURN PC ON STACK
686         000164         000207         5$:  RTS      PC           :GC HOME.
687                                     :ALL SIZING IS DONE
688
689         000174         000174         :****SOFTWARE SWITCH REGISTER****
690         000174         000000         .=174
691         000176         000000         DISPREG: 0           :SOFTWARE DISPLAY REGISTER
692                                     SWREG: 0           :SOFTWARE SWITCH REGISTER
693
694         :PROGRAM START
695         000200         .=200

```

```

696 000200 000137 001512          JMP      .START          ;GO TO START OF PROGRAM
697
698                                . =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$,R4             ;SET FOR TIME OUT TRAP--NO DEVICE--
707 000256 112761 000012 000005   1$:  MOVVB  #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   MOVVB  #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   MOVVB  #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

```

```

752 000534 005301          DEC      R1          ;DEC NUMBER OF DQ11S
753 000536 001371          BNE      3$         ;BR IF MORE TO GO
754 000540 012737 000006 000004  MOV      #6,@#4     ;RESET TIME OUT VECTOR
755 000546 013737 001500 001502  MOV      DQACTV,SAVACT ;SAVE ACTIVE
756 000554 012737 000340 000022  MOV      #340,@#22  ;SET IOT TRAP PRIO: TO 7
757 000562 012702 001400          MOV      #1400,R2   ;SET TABLE POINTER
758 000566 012700 000300          MOV      #300,R0    ;SET VECTOR START
759 000572 012701 000302          MOV      #302,R1    ;SET VECTOR+2 START
760 000576 000137 000056          JMP      VECMAP     ;GO FIND THE VECTORS
761 000602 104402          4$:      TYPE      ;TYPE MESSAGE
762 000604 014641          MERR2     ;I DIDN'T FIND ANY DQ11S. DON'T USE AUTO SIZE.
763 000606 005000          CLR      R0        ;
764 000610 000000          HALT     ;HOW CAN I TEST NO DQ11S
765 000612 000776          BR       .-2       ;DON'T LET OPR HIT CONT. SW
766 000614 012716 000466 5$:      MOV      #2$,(SP) ;ENTERED BY TIME OUT TRAP
767 000620 000002          RTI        ;GO HOME.
768
769
770          001000          .=1000
771 001000 005377 055103 050504  MTITLE: .ASCIZ <377><12>/CZDQEE0/<377>/ MISC & BCC TST/<377>
772 001006 042505 177460 046440
773 001014 051511 020103 020046
774 001022 041502 020103 051524
775 001030 177524      000
776
777          001200          .=1200
778          ;INDIRECT POINTERS
779
780 001200 177570          SWR:      177570     ;SWITCH REGISTER POINTER
781 001202 177570          LIGHTS: 177570    ;DISPLAY REGISTER POINTER
782 001204 177560          TKCSR:   177560    ;TELETYPE KEYBOARD CONTROL REGISTER
783 001206 177562          TKDBR:   177562    ;TELETYPE KEYBOARD DATA BUFFER
784 001210 177564          TPCSR:   177564    ;TELEPRINTER CONTROL REGISTER
785 001212 177566          TPDBR:   177566    ;TELEPRINTER DATA BUFFER
786
787          ;PROGRAM CONTROL PARAMETERS
788
789 001214 000000          RETURN: 0         ;SCOPE ADDRESS FOR LOOP ON TEST
790 001216 000000          NEXT:   0         ;ADDRESS OF NEXT TEST TO BE EXECUTED
791 001220 000000          LOCK:   0         ;ADDRESS FOR LOCK ON CURRENT DATA
792 001222 000003          ICOUNT: 3         ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
793 001224 000000          LPCNT: 0         ;NUMBER OF ITERATIONS COMPLETED
794 001226 000000          TSTNO:  0         ;NUMBER OF TEST IN PROGRESS
795 001230 000000          PASCNT: 0        ;NUMBER OF PASSES COMPLETED
796 001232 000000          ERRCNT: 0        ;TOTAL NUMBER OF ERRORS
797 001234 000000          LSTERR: 0        ;PC OF LAST ERROR CALL
798
799          ;PROGRAM VARIABLES
800
801 001236 000000          CHAR1:  0
802 001240 000000          CHAR2:  0
803 001242 000000          CHAR3:  0
804 001244 000000          TEMP1:  0        ;TEMPORARY STORAGE
805 001246 000000          TEMP2:  0        ;TEMPORARY STORAGE
806 001250 000000          TEMP3:  0        ;TEMPORARY STORAGE
807 001252 000000          TEMP4:  0        ;TEMPORARY STORAGE

```

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

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

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877

001310 000
001311 000
001312 000
001313 000
000000

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

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

:*****
:*****
.TRPTAB:
SCOPE=TRAP+0 ;CALL TO SCOPE LOOP AND ITERATION HANDLER
.SCOPE
SCOPI=TRAP+1 ;CALL TO LOOP ON CURRENT DATA HANDLER
.SCOPI
TYPE=TRAP+2 ;CALL TO TELETYPE OUTPUT ROUTINE
.TYPE
INSTR=TRAP+3 ;CALL TO ASCII STRING INPUT ROUTINE
.INSTR
INSTER=TRAP+4 ;CALL TO INPUT ERROR HANDLER
.INSTER
PARAM=TRAP+5 ;CALL TO NUMERICAL DATA INPUT ROUTINE
.PARAM
SAV05=TRAP+6 ;CALL TO REGISTER SAVE ROUTINE
.SAV05
RES05=TRAP+7 ;CALL TO REGISTER RESTORE ROUTINE
.RES05
CONVRT=TRAP+10 ;CALL TO DATA OUTPUT ROUTINE
.CONVRT
CNVRT=TRAP+11 ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
.CNVRT
MSTCLR=TRAP+12 ;CALL TO ISSUE MASTER CLEAR
.MSTCLR
MEMCLR=TRAP+13 ;CALL TO CLEAR ALL SCRATCH PAD MEMORIES
.MEMCLR
CKSWR=TRAP+14 ;CALL TO ALLOW SWREG TO BE LOADED FROM TTY
.CKSWR
CNTLU=TRAP+15 ;CALL TO ALLOW LOADING OF SWREG FROM TTY
.CNTLU

:*****
:*****

:DQ11 VECTOR AND REGISTER INDIRECT POINTERS

DQRVEC: 0 ;POINTER TO DQ11 RECEIVER INTERRUPT VECTOR
DQRLVL: 0 ;POINTER TO DQ11 RECEIVER INTERRUPT SERVICE PS
DQTVEC: 0 ;POINTER TO DQ11 TRANSMITTER INTERRUPT VECTOR
DQTLVL: 0 ;POINTER TO DQ11 TRANSMITTER INTERRUPT SERVICE PS
DQRCSR: 0 ;POINTER TO DQ11 RECEIVER CONTROL REGISTER
DQRCSH: 0 ;POINTER TO HIGH BYTE OF DQ11 RECEIVER CONTROL REGISTER

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

878	001364	000000	DQTCR: 0	: POINTER TO DQ11 TRANSMITTER CONTROL REGISTER
879	001366	000000	DQERR: 0	: POINTER TO DQ11 ERROR REGISTER
880	001370	000000	DQREG: 0	: POINTER TO HIGH BYTE OF ERROR REGISTER
881	001372	000000	DQSEC: 0	: POINTER TO DQ11 SECONDARY REGISTER
882	001374	000000	DQSECH: 0	: POINTER TO HIGH BYTE OF DQ11 SECONDARY REGISTER

883
884
885
886
887

:DQ11 STATUS TABLE AND ADDRESS ASSIGNMENTS

888		001400	.=1400	
889	001400	000001	DQCR00: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 00
890	001402	000001	DQST00: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 00
891	001404	000001	DQCR01: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 01
892	001406	000001	DQST01: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 01
893	001410	000001	DQCR02: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 02
894	001412	000001	DQST02: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 02
895	001414	000001	DQCR03: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 03
896	001416	000001	DQST03: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 03
897	001420	000001	DQCR04: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 04
898	001422	000001	DQST04: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 04
899	001424	000001	DQCR05: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 05
900	001426	000001	DQST05: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 05
901	001430	000001	DQCR06: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 06
902	001432	000001	DQST06: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 06
903	001434	000001	DQCR07: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 07
904	001436	000001	DQST07: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 07
905	001440	000001	DQCR10: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 10
906	001442	000001	DQST10: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 10
907	001444	000001	DQCR11: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 11
908	001446	000001	DQST11: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 11
909	001450	000001	DQCR12: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 12
910	001452	000001	DQST12: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 12
911	001454	000001	DQCR13: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 13
912	001456	000001	DQST13: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 13
913	001460	000001	DQCR14: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 14
914	001462	000001	DQST14: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 14
915	001464	000001	DQCR15: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 15
916	001466	000001	DQST15: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 15
917	001470	000001	DQCR16: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 16
918	001472	000001	DQST16: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 16
919	001474	000001	DQCR17: .BLKW 1	: CONTROL STATUS REGISTER FOR DEVICE NO: 17
920	001476	000001	DQST17: .BLKW 1	: VECTOR AND CONFIGURATION STATUS FOR DEVICE NO: 17
921	001500	000001	DQACTV: .BLKW 1	: HOLD ACTIVE BITS FOR TESTING
922	001502	000001	SAVACT: .BLKW 1	: SAVE NUMBER OF ACTIVE DQ11S
923	001504	000001	DQNUM: .BLKW 1	: OCTAL NUMBER OF TOTAL NUMBER OD DQ11S
924	001506	000001	DQCSR: .BLKW 1	: CSR OF DQ11 UNDER TEST
925	001510	000001	DQSTAT: .BLKW 1	: VECTOR AND CONFIGURATION STATUS OF DQ11 UNDER TEST

926
927
928
929
930
931
932
933

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

PROGRAM INITIALIZATION AND START UP.

```

934 001512 012737 000340 177776 .START: MOV #340,PS ;LOCK OUT INTERRUPTS
935 001520 012706 001200 MOV #STACK,SP ;SET UP STACK
936 001524 012737 014300 000024 MOV #.PFAIL,@#24 ;SET UP POWER FAIL VECTOR
937 001532 013737 001504 001276 MOV DQNUM,SAVNUM
938 001540 105037 001311 CLR STFLG ;CLEAR START FLAG
939 001544 005037 001230 CLR PASCNT ;CLEAR PASS COUNT
940 001550 105037 001312 CLR ERRFLG ;CLEAR ERROR FLAG
941 001554 005037 001302 CLR RUNFLG
942 001560 012737 001400 001300 MOV #1400,CREAM
943 001566 005037 001232 CLR ERRCNT ;CLEAR ERROR COUNT
944 001572 005037 001234 CLR LSTERR ;CLEAR LAST ERROR POINTER
945 001576 012737 000001 001226 MOV #1,TSTNO ;SET UP FOR TEST 1
946 001604 012737 001512 001214 MOV #.START,RETURN ;SET UP FOR POWER FAIL BEFORE
947 ;TESTING STARTS
948 001612 012737 177570 001200 MOV #DSWR,SWR ;MOV HARDWARE SWR TO SWR
949 001620 012737 177570 001202 MOV #DLIGHTS,LIGHTS ;MOV DISPLAY LIGHTS TO LIGHTS
950 001626 013746 000006 MOV @#6,-(SP) ;SAVE VECTORS
951 001632 013746 000004 MOV @#4,-(SP)
952 001636 012737 001656 000004 MOV #64$,@#4 ;SET UP FOR TIMEOUT
953 001644 022777 177777 177326 CMP #-1,@SWR ;REFERENCE HARDWARE SWITCH REGISTER
954 001652 001402 BEQ 65$
955 001654 000407 BR 66$
956 001656 022626 64$: CMP (SP)+,(SP)+ ;ADJUST STACK
957 001660 012737 000176 001200 65$: MOV #SWREG,SWR ;POINT TO SOFTWARE SWITCH REG
958 001666 012737 000174 001202 MOV #DISPREG,LIGHTS ;POINT TO SOFT DISPLAY REG
959 001674 012637 000004 66$: MOV (SP)+,@#4 ;RESTORE VECTORS
960 001700 012637 000006 MOV (SP)+,@#6
961 001704 005737 000042 TST @#42 ;UNDER MONITOR
962 001710 001014 BNE 67$
963 ;:*****THE NEXT 4 LINES OF CODE MOVED TO SOLVE PR#2757 (JUNE 78)*****
964 001712 105737 001310 TSTB INIFLG ;HAS INITIALIZATION BEEN PERFORMED?
965 001716 001035 BNE 12$ ;IF YES, BR
966 001720 104402 001000 TYPE ,MTITLE ;TYPE TITLE MESSAGE
967 001724 105137 001310 COMB INIFLG ;IF NOT SET FLAG AND INIT
968 001730 022737 000176 001200 CMP #SWREG,SWR ;IS SWREG USED
969 001736 001001 BNE 67$
970 001740 104415 CNTLU
971 001742 105777 177232 67$: TSTB @SWR
972 001746 100402 BMI .+6
973 001750 004737 000220 JSR PC,CSRMAP
974 001754 104402 015126 TYPE ,XHEAD
975 001760 012737 001400 001244 MOV #1400,TEMP1
976 001766 017737 177252 001246 MOV @TEMP1,TEMP2
977 001774 001406 BEQ .+16
978 001776 104410 CONVRT
979 002000 015154 XSTATQ
980 002002 062737 000002 001244 ADD #2,TEMP1
981 002010 000766 BR .-22
982 002012 032777 000001 177160 12$: BIT #SW00,@SWR
983 002020 001424 BEQ 1$
984 002022 104402 TYPE
985 002024 015047 MNEW
986 002026 005000 CLR R0
987 002030 000000 HALT
988 002032 104414 CKSWR
989 002034 027737 177140 001502 CMP @SWR,SAVACT

```

```

990 002042 101404      BLOS      11$
991 002044 104402      TYPE
992 002046 014710      MERR3
993 002050 000000      HALT
994 002052 000776      BR      -2
995 002054 017737 177120 001500 11$:  MOV      @SWR,DQACTV
996 002062 013700 001500      MOV      DQACTV,R0
997 002066 000000      HALT
998 002070 104414      CKSWR
999 002072 012700 000300      1$:  MOV      #300,R0
1000 002076 012701 000302      MOV      #302,R1
1001 002102 010120      2$:  MOV      R1,(R0)+
1002 002104 005021      CLR      (R1)+
1003 002106 022021      CMP      (R0)+,(R1)+
1004 002110 022700 001000      CMP      #1000,R0
1005 002114 001372      BNE      2$
1006
1007                      ;TEST START AND RESTART
1008
1009 002116 012737 000340 177776 .BEGIN: MOV      #340,PS          ;LOCK OUT INTERRUPTS
1010 002124 012706 001200      MOV      #STACK,SP      ;SET UP STACK
1011 002130 005737 000042      TST      @#42          ;IS PROGRAM UNDER MONITOR CONTROL
1012 002134 001040      BNE      3$
1013 002136 104414      CKSWR          ;CHECK FOR <^G>
1014 002140 032777 000004 177032  BIT      #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
1015 002146 001411      BEQ      1$
1016 002150 104402 014746      TYPE      ,MLOCK
1017 002154 012737 000240 012562  MOV      #NOP,TTST
1018 002162 012737 000240 012564  MOV      #NOP,TTST+2    ;SET UP TO LOCK
1019 002170 000406      BR      2$
1020 002172 013737 012660 012562 1$:  MOV      BRW,TTST
1021 002200 013737 012662 012564  MOV      BRX,TTST+2    ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
1022 002206 032777 000002 176764 2$:  BIT      #SW01,@SWR      ;IF SW01=1, GET STARTING PC
1023 002214 001410      BEQ      3$
1024 002216 104403      INSTR
1025 002220 014734      MTSTPC
1026 002222 104405      PARAM
1027 002224 002254      TST1
1028 002226 007472      TLAST
1029 002230 001214      #RETURN
1030 002232      001      .BYTE      1
1031 002233      001      .BYTE      1
1032 002234 000403      BR      4$
1033 002236 012737 002254 001214 3$:  MOV      #TST1,RETURN    ;START AT TEST 1
1034 002244 104402 014636 4$:  TYPE      ,MR          ;TYPE R
1035 002250 000177 176740      JMP      @RETURN        ;START TESTING
1036
1037                      : TEST 1
1038                      :*****
1038 002254 012737 000001 001226 TST1:  MOV      #1,TSTNO
1039 002262 012737 002646 001214      MOV      #TST2,RETURN
1040 002270 012737 002646 001216      MOV      #TST2,NEXT
1041 002276 105737 001302      TSTB      RUNFLG      ;IS THIS MY FIRST TIME HERE?
1042 002302 001010      BNE      1$          ;BR IF FLAG IS SET
1043 002304 012737 000001 001304      MOV      #BIT0,RUN      ;SET RUN POINTER.
1044 002312 012737 000020 001306      MOV      #15,RUNCNT    ;SET FOR MAX OF 16 DQ11'S PER SYSTEM
1045 002320 105137 001302      COMB      RUNFLG      ;SET RUN FLAG

```

```

1046 002324 033737 001304 001500 1$: BIT RUN,DQACTV ;FIND AN ACTIVE DQ11 TO TEST.
1047 002332 001032 BNE 3$ ;BR IF I FOUND ONE TO TEST.
1048 002334 005737 001500 TST DQACTV ;FIND OUT IF THERE ARE NO DQ11 ACTIVE.
1049 002340 001423 BEQ 2$ ;BR TO FATAL ERROR. WHY AM I HERE IF NO ACTIVE DQ11'S???
1050 002342 000257 CCC ;CLEAR ALL THE CONDITION CODES OF CPU
1051 002344 006137 001304 ROL RUN ;UPDATE RUN POINTER
1052 002350 062737 000004 001300 ADD #4,CREAM ;UPDATE ADDRESS POINTER.
1053 002356 005337 001306 DEC RUNCNT ;DEC NUMBER OF TIMES I LOOKED AT ACTIVE.
1054 002362 001360 BNE 1$ ;BR AND KEEP LOOKING.
1055 002364 012737 000020 001306 MOV #16,RUNCNT ;START RESTORING MY POINTERS.
1056 002372 012737 001400 001300 MOV #1400,CREAM ;RESTORE ADDRESS POINTER
1057 002400 012737 000001 001304 MOV #1,RUN ;RESTORE RUN POINTER.
1058 002406 000746 BR 1$ ;KEEP ON TESTING.
1059 002410 104402 2$: TYPE ;ALLERT OPERATOR OF FATAL ERROR
1060 002412 014641 MERR2 ;NO DQ11 ACTIVE. WHY AM I HERE???
1061 002414 000000 HALT ;YOU MUST RELOAD DQ11 DIAGNOSTIC!!
1062 002416 000776 BR -.2 ;STICK HERE ON CONT.
1063 002420 000257 3$: CCC ;CLEAR CPU COND. CODES
1064 002422 006137 001304 ROL RUN ;UPDATE RUN. ACTIVE DQ11 FOUND.
1065 002426 017737 176646 001506 MOV @CREAM,DQCSR ;PLACE ADDRESS OF DQ11 AT DQCSR
1066 002434 062737 000002 001300 ADD #2,CREAM ;UPDATE ADDRESS POINTER
1067 002442 017737 176632 001510 MOV @CREAM,DQSTAT ;PLACE STATUS OF DQ11 AT DQSTAT
1068 002450 062737 000002 001300 ADD #2,CREAM ;UPDATE ADDRESS POINTER
1069 002456 013737 001506 001360 MOV DQCSR,DQCSR
1070 002464 013737 001510 001350 MOV DQSTAT,DQVEC
1071 002472 042737 177007 001350 BIC #177007,DQVEC
1072 002500 013737 001350 001352 MOV DQVEC,DQRLVL ;GENERATE ADDRESS OF RECEIVER INTERRUPT SERVICE PS
1073 002506 062737 000002 001352 ADD #2,DQRLVL
1074 002514 013737 001352 001354 MOV DQRLVL,DQTEC ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT VECTOR
1075 002522 062737 000002 001354 ADD #2,DQTEC
1076 002530 013737 001354 001356 MOV DQTEC,DQTLVL ;GENERATE ADDRESS OF TRANSMITTER INTERRUPT SERVICE PS
1077 002536 062737 000002 001356 ADD #2,DQTLVL
1078 002544 013737 001360 001362 MOV DQCSR,DQCSH
1079 002552 005237 001362 INC DQCSH ;GENERATE ADDRESS OF HIGH BYTE
1080 002556 013737 001360 001364 MOV DQCSR,DQTCR ;GENERATE ADDRESS OF TRANSMITTER CONTROL REGISTER
1081 002564 062737 000002 001364 ADD #2,DQTCR
1082 002572 013737 001364 001366 MOV DQTCR,DQERR ;GENERATE ADDRESS OF ERROR REGISTER
1083 002600 062737 000002 001366 ADD #2,DQERR
1084 002606 013737 001366 001370 MOV DQERR,DQREG ;GENERATE ADDRESS OF HIGH BYTE OF ERROR REGISTER
1085 002614 005237 001370 INC DQREG
1086 002620 013737 001370 001372 MOV DQREG,DQSEC ;GENERATE ADDRESS OF SECONDARY REGISTER
1087 002626 005237 001372 INC DQSEC
1088 002632 013737 001372 001374 MOV DQSEC,DQSECH ;GENERATE ADDRESS OF HIGH BYTE
1089 002640 005237 001374 INC DQSECH
1090 002644 000240 NOP
1091
1092
1093
1094
1095

```

1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151

:
:CABLE TEST.
:TEST OF DATA REALIBILITY THROUGH
:CABLE AND LEVEL CONVERTERS.

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

: TEST 2
:*****

```
TST2:  MOV #2,TSTNO
      MOV #TST3,NEXT

CKSYN1: BIT #SYNBIT,DQSTAT ;ADJUST POINTER FOR NUMBER OF SYNC CHARS.
        BNE 1$ ;BR IF TWO SYNC CHARS SELECTED.
        MOV #377,SYNC ;SET SYNC TO MARK
        BR CKDN ;BR TO CONT.
1$:     MOV #26,SYNC ;RESET FIRST SYNC TO 26
CKDN:  NOP ;CONTINUE TEST.

        BIT #JUMBIT,DQSTAT ;DOES THE TEST JUMPER EXIST.
        BNE .+14 ;BR IF YES
        MOV NEXT,RETURN ;PREPARE TO DO NEXT TEST.
        JMP @RETURN ;GOTO NEXT TEST
        CLR R0 ;ZERO DATA POINTER.
1$:     MOV #TXBUFF,R4 ;SET BUFFER POINTER
        MOV R0,(R4)+ ;FILL TX BUFFER WITH BINARY COUNT PATTERN
        INCB R0 ;UPDATE CHAR.
        BNE 1$ ;BR IF MORE TO DO.
2$:     MEMCLR ;CLEAR DQ11 MEMORIES.
        CLR R0 ;ZERO COUNTER POINTER
        MOV #RXBUFF,R4 ;PREPARE TO ZERO ALL RX BUFFER.
3$:     CLRB (R4)+ ;START CLEARING.
        INCB R0 ;UPDATE
        BNE 3$ ;BR IF NOT ALL CLEARED.
        CLRB @DQREG ;SELECT RX BA PRI.
        MOV #RXBUFF,@DQSEC ;LOAD IT.
        INCB @DQREG ;SELECT RX WC PRI.
        MOV #-400,@DQSEC ;LOAD IT FOR 400(8) CHARS.
        INCB @DQREG ;SEL TX BA PRI.
        MOV #SYNC,@DQSEC ;LOAD IT.
        INCB @DQREG ;SEL TX WC PR.
        MOV #-402,@DQSEC ;SET 400(8) CHARS AND TWO SYNC.
        MOV #11,@DQREG ;SEL SYNC REGISTER
        MOV .SYNC,@DQSEC ;LOAD IT.
        INCB @DQREG ;GET MISC REGISTER
        MOV #4000,@DQSEC ;SET FOR EIGHTBITS.
        CLR TEMP1 ;SET DELAY.....
        MOV #20,TEMP2
        INC @DQCSR ;SET RX GO!!
        INC @DQTCR ;SET TX GO!!
4$:     TST @DQERR ;ANY ERRORS
        BPL 7$ ;BR IF NO ERRORS
```

::++E

```

1152 003114 017700 176240      MOV      @DQRCR,R0
1153 003120 017701 176240      MOV      @DQTCR,R1
1154 003124 017702 176236      MOV      @DQERR,R2
1155 003130 104007              HLT      7                ;THE DQ11 ERROR FLAG IS SET.
1156 003132 105777 176222      7$: TSTB   @DQRCR          ;IS RX PRI DONE SET?
1157 003136 100407              BMI      5$              ;BR IF YES
1158 003140 005237 001244      INC      TEMP1          ;DELAY.....
1159 003144 001360              BNE      4$              ;
1160 003146 005337 001246      DEC      TEMP2          ;
1161 003152 001355              BNE      4$              ;
1162 003154 104000              HLT      ;
1163 003156 005000              5$: CLR      R0          ;RX PRI. DONE FAILED TO SET.
1164 003160 005037 001252      CLR      TEMP4         ;ZERO COUNTER.
1165 003164 005037 001254      CLR      TEMP5         ;CLEAR STORAGE
1166 003170 012704 015376      MOV      #TXBUFF,R4    ;SAME..
1167 003174 012705 016000      MOV      #RXBUFF,R5    ;GET TX BUFFER AREA
1168 003200 112437 001254      6$: MOVB   (R4)+,TEMP5   ;GET RX BUFFER AREA
1169 003204 112537 001252      MOVB   (R5)+,TEMP4     ;LOAD FOR ERROR CALL
1170 003210 023737 001254 001252  CMP      TEMP5,TEMP4   ;
1171 003216 001401              BEQ      +4             ;DOES DATA CHECK OUT OK?
1172 003220 104004              HLT      4             ;BR IF GOOD DATA.
1173 003222 105200              INCB    R0             ;DATA COMPARISON ERROR.
1174 003224 001365              BNE     6$             ;UPDATE COUNTER
1175 003226 104400              SCOPE   6$            ;BR IF MORE DATA TO CHECK.
1176                                     ;SCOPE THE TEST.

```

```

;RECEIVER STRIP SYNC TEST.
;TEST THAT THE RECEIVER CAN STRIP SYNC.
;CHARACTERS FROM 000-025 WILL BE TRANSFERED
;WITH A TRAIL OF SYNC CHARS. FOLLOWING (026).
;THE TRANSMITTER AND RECEIVER BUFFERS ARE BOTH
;CLEARED BEFORE THE TEST IS EXECUTED.
;A TOTAL OF 400 CHARS. WILL BE TRANSMITTER INTO
;THE RECEIVER. WHEN RX PRI. DONE SETS;
;THE RECEIVER BUFFER IS CHECKED FOR ANY SYNC
;CHARACTERS (026). IF NONE ARE FOUND THEN THE
;RECEIVER DID INDEED STRIP SYNC.

```

: TEST 3

```

1192 :*****
1193 003230 012737 000003 001226  TST3: MOV      #3,TSTNO
1194 003236 012737 003522 001216  MOV      #TST4,NEXT
1195 003244 005000              CLR      R0              ;CLEAR POINTER
1196 003246 012704 016000      MOV      #RXBUFF,R4     ;SET THE RX BUFFER
1197 003252 105024              1$: CLRB   (R4)+         ;BEGIN TO CLEAR THE RX BUFFER
1198 003254 105200              INCB    R0              ;ALL DONE?
1199 003256 001375              BNE     1$             ;BR IF NO
1200 003260 005000              CLR      R0              ;RESET R0 TO ZERO
1201 003262 012704 015376      MOV      #TXBUFF,R4     ;GET TX BUFFER
1202 003266 105024              8$: CLRB   (R4)+         ;BEGIN TO CLEAR THE TX BUFFER
1203 003270 105200              INCB    R0              ;DONE YET?
1204 003272 001375              BNE     8$             ;BR IF NO
1205
1206 003274 005000              CLR      R0              ;RESET R0
1207 003276 012704 015376      MOV      #TXBUFF,R4     ;GET TX BUFFER

```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1208 003302 110024      7$:  MOVB  R0,(R4)+  ;START FILLING WITH CHARS.
1209 003304 105200      INCB  R0          ;UPDATE POINTER
1210 003306 022700 000026  CMP   #26,R0     ;SYNC YET?
1211 003312 001373      BNE   7$         ;BR IF NO
1212 003314 012702 000026  MOV   #26,R2     ;SET FOR SYNC CHAR.
1213 003320 110024      6$:  MOVB  R0,(R4)+  ;FILL TX BUFFER WITH SYNC CHARS.
1214 003322 105202      INCB  R2          ;MORE TO GO?
1215 003324 100375      BPL   6$        ;BR IF YES
1216
1217 003326 104413      MEMCLR          ;CLEAR ALL DQ11
1218 003330 105077 176034  CLRB  @DQREG     ;SEL RX BA PRI.
1219 003334 012777 016000 176030  MOV   #RXBUFF,@DQSEC ;SET BUFFER.
1220 003342 105277 176022      INCB  @DQREG     ;RX WC PRI.
1221 003346 012777 177400 176016  MOV   #-400,@DQSEC ;256. CHARS
1222 003354 105277 176010      INCB  @DQREG     ;TX BA PRI.
1223 003360 012777 015374 176004  MOV   #SYNC,@DQSEC ;SET TO XMIT SYNC CHARS.
1224 003366 105277 175776      INCB  @DQREG     ;TX WC PRI.
1225 003372 012777 177400 175772  MOV   #-400,@DQSEC ;256. CHARS
1226 003400 112777 000011 175762  MOVB  #11,@DQREG ;SYNC REGISTER
1227 003406 013777 015372 175756  MOV   .SYNC,@DQSEC ;LOAD SYNC REGISTER
1228 003414 105277 175750      INCB  @DQREG     ;MISC. REGISTER.
1229 003420 012777 004010 175744  MOV   #4010,@DQSEC ;EIGHT BITS AND TEST LOOP
1230 003426 012777 000003 175724  MOV   #00003,@DQRCR ;GO AND STRIP SYNC.
1231 003434 005277 175724      INC   @DQTCR    ;SET TX GO.
1232 003440 005000      CLR   R0        ;PREPARE TO DELAY.
1233 003442 005001      CLR   R1        ;SAME
1234 003444 105777 175710      2$:  TSTB  @DQRCR    ;RX DONE?
1235 003450 100406      BMI   9$        ;BR IF YES
1236 003452 062700 000001      ADD   #1,R0     ;D
1237 003456 001372      BNE   2$        ;E
1238 003460 105201      INCB  R1        ;L
1239 003462 100370      BPL   2$        ;A
1240
1241 003464 104000      HLT
1242 003466 005077 175666      9$:  CLR   @DQRCR    ;RX PRI DONE NOT SET.
1243 003472 005077 175666      CLR   @DQTCR    ;DISABLE RX
1244 003476 005000      CLR   R0        ;DISABLE TX
1245 003500 012704 016000      MOV   #RXBUFF,R4 ;SET COUNTER
1246 003504 122724 000026      3$:  CMPB  #26,(R4)+ ;GET RX BUFFER
1247 003510 001001      BNE   .+4       ;ANY SYNC CHARS?
1248 003512 104000      HLT            ;BR IF NONE
1249 003514 105200      INCB  R0        ;RECEIVER DID NOT STRIP SYNC.
1250 003516 001372      INCB  R0        ;UPDATE COUNTER
1251 003520 104400      BNE   3$        ;BR IF MORE TO GO.
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
    
```

```

;DQ11 MEMORY TRANSFER TESTS
;
;IF THE PROCESSOR HAS AT LEAST 8K OF MEMORY
;THIS TEST WILL BE EXECUTED.
;
;THE FIRST PART EXERCISES THE TRANSMITTER ALONE
;TRANSMITTING FROM ADD.20000 TO LAST MEMORY ADD.
    
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1264 : UNDER 28K MINUS 400 (SAVE ABL). THE DATA IS
 1265 : VERIFIED ONE CHAR AT A TIME. TWO EIGHT BIT
 1266 : CHARS ARE TRANSFERED AT ONE TIME.
 1267 :
 1268 : AFTER THE TX ALONE THEN THE TRANSMITTER AND RECEIVER
 1269 : ARE EXERCISED TOGETHER IN THE SAME
 1270 : MANNER AS DESCRIBED ABOVE.
 1271 :
 1272 :
 1273 :
 1274 :
 1275 :

: TEST 4

```

*****
TST4:  MOV    #4,TSTNO
      MOV    #TST5,NEXT
      MEMCLR
      MOV    @#4,R5          ;SAVE THE TIME OUT VECTOR
      MOV    #10$,@#4      ;LOAD TRAP VECTOR
      TST    @#20000       ;CHECK FOR 8K OF MEMORY.
      NOP
      MOV    #2$,@#4       ;CPU DOES HAVE AT LEAST 8K.
      MOV    #2000,R0      ;PREPARE TO SIZE MEMORY TO 28K
1$:   ADD    #20000,R0     ;CHECK MEMORY.
      TST    (R0)         ;EXIST?
      NOP
      CMP    #160000,R0   ;28K HIT YET.
      BNE   1$           ;BR IF NO
2$:   MOV    #STACK,SP   ;ADJUST STACK
      MOV    #4$,LOCK    ;SET FOR LOCK (SW09=1)
      MOV    R5,@#4      ;RESET TRAP VECTOR.
      SUB    #5000,R0    ;ALLOW ROOM FOR DDP2 MONITOR.
      MOV    R0,LIMIT.H1 ;SAVE LAST MEMORY ADDRESS
      CLR    R2          ;ZERO DATA CHAR POINTER
3$:   MOV    #20000,R0   ;PREPARE TO FILL MEMORY WITH BINARY COUNT
      MOVB  R2,(R0)+    ;START FILL
      INC   R2          ;UPDATE CHAR.
      CMPB #26,R2      ;DOES IT EQUAL THE SYNC CHAR?
      BNE  .+4         ;BR IF NO
      INC   R2          ;BUMP ONE HIGHER
      CMP  R0,LIMIT.HI ;IS ALL OF MEMORY FULL?
      BNE  3$         ;BR IF NO.
4$:   MOVB  #2,@DQREG   ;SEL TX BA PRI.
      MOV   #20000,@DQSEC ;SET TX BA TO FIRST ADD IN 8K
      MOV   #20000,R0    ;SET SOFTWARE POINTER.
      BR   5$         ;CONTINUE TEST
5$:   MOVB  #2,@DQREG   ;SEL TX BA PRI.
      SUB  #2,@DQSEC   ;GC BACKWARDS FOR SCOPE ROUTINE
      MOVB #2,@DQREG   ;SEL TX BA PRI.
      MOV  @DQSEC,TEMP4 ;SAVE THE TX BA PRI. ADDRESS
      MSTCLR ;DO INIT DQ11
      MOVB #2,@DQREG   ;SEL TX BA PRI.
      MOV  TEMP4,@DQSEC ;RELOAD TX BA PRI.
      INCB @DQREG      ;SEL TX WC PRI.
      MOV  #-2,@DQSEC  ;SET FOR A TWO EIGHT BIT XFER
      MOVB #MISC.,@DQREG ;SEL MISC REGISTER
      MOV  #4012,@DQSEC ;SET 8 BITS TEST LOOP AND AUTO/STEP
      INC  @DQTCR      ;SET TX GO.
  
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1320 004010 005777 175352      TST      @DQERR      ;ANY ERRORS?
1321 004014 100001             BPL      .+4         ;BR IF NO ERRORS
1322 004016 104000             HLT                      ;DQ11 ERROR FLAG SET.
1323 004020 112777 000013 175342  MOVB     #13,@DQREG   ;SEL TX MUX REG
1324 004026 017737 175340 001252  MOV      @DQSEC,TEMP4 ;READ TX MUX
1325 004034 011037 001254             MOV      (R0),TEMP5  ;READ SOFTWARE POINTER
1326 004040 023710 001252             CMP      TEMP4,(R0)  ;IS THE DATA CORRECT
1327 004044 001401             BEQ      .+4         ;BR IF GOOD
1328 004046 104005             HLT      5          ;DATA COMPARISON ERROR.
1329
1330 004050 104401             SCOP1                    ;LOCK ON CHARACTER (SW09=1)
1331 004052 005720             TST      (R0)+       ;UPDATE SOFTWARE POINTER.
1332 004054 020037 004456             CMP      R0,LIMIT.HI ;ALL DONE?
1333 004060 001321             BNE      5$         ;BR IF NO
1334
1335
1336 ;THE ABOVE WAS FOR THE TX ONLY
1337
1338 ;THE BELOW ROUTINE EXECISES BOTH THE
1339 ;TX AND RX TOGETHER.
1340
1341 ;NOTE THAT THE RX CA SHOULD BE
1342 ;400 (8) LOCATIONS HIGHER THAN THE TX CA.
1343
1344
1345
1346 004062 012737 004110 001220      MOV      #6$,LOCK     ;SET FOR LOCK ON CHAR(SW09=1)
1347 004070 012700 020000             MOV      #20000,R0   ;SET FIRT ADD IN 8K
1348 004074 112777 000006 175266      MOVB     #6,@DQREG   ;SEL TX BA SEC
1349 004102 010077 175264             MOV      R0,@DQSEC   ;LOAD TX BA SEC
1350 004106 000413             BR       7$         ;CONT TEST.
1351 004110 112777 000006 175252 6$:  MOVB     #6,@DQREG   ;SEL TX BA SEC.
1352 004116 162777 000002 175246      SUB      #2,@DQSEC   ;KILL LAST XFER
1353 004124 105077 175240             CLRB    @DQREG      ;SEL RX BA PRI.
1354 004130 162777 000002 175234      SUB      #2,@DQSEC   ;KILL LAST XFER
1355 004136 112777 000006 175224 7$:  MOVB     #6,@DQREG   ;SEL TX BA SEC
1356 004144 017737 175222 001252      MOV      @DQSEC,TEMP4 ;SAVE IT
1357 004152 104412             MSTCLR                    ;INIT DQ11
1358 004154 105077 175210             CLRB    @DQREG      ;SEL RX BA PRI
1359 004160 013777 001252 175204      MOV      TEMP4,@DQSEC ;LOAD IT
1360 004166 062777 000400 175176      ADD      #400,@DQSEC ;UPDATE IT
1361 004174 105277 175170             INCB    @DQREG      ;SEL RX WC PRI
1362 004200 012777 177776 175164      MOV      #-2,@DQSEC  ;SET FOR TWO CHARS.
1363 004206 105277 175156             INCB    @DQREG      ;SEL TX BA PRI
1364 004212 012777 015374 175152      MOV      #SYNC,@DQSEC ;TX SYNC ON PRI.
1365 004220 105277 175144             INCB    @DQREG      ;SEL TX WC PRI.
1366 004224 012777 177776 175140      MOV      #-2,@DQSEC  ;SET FOR TWO SYNC.
1367 004232 112777 000006 175130      MOVB     #6,@DQREG   ;LOAD TX BA SEC.
1368 004240 013777 001252 175124      MOV      TEMP4,@DQSEC ;LOAD IT
1369 004246 105277 175116             INCB    @DQREG      ;SEL TX WC SEC
1370 004252 012777 177776 175112      MOV      #-2,@DQSEC  ;SET FOR TWO CHAR XFER
1371 004260 112777 000011 175102      MOVB     #11,@DQREG  ;SEL THE SYNC REG
1372 004266 013777 015372 175076      MOV      .SYNC,@DQSEC ;LOAD IT
1373 004274 105277 175070             INCB    @DQREG      ;SEL MISC REG.
1374 004300 012777 004010 175064      MOV      #4010,@DQSEC ;EIGHT BITS TEST LOOP
1375 004306 005277 175046             INC      @DQRC5R     ;SET RX GO..
    
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1376 004312 005277 175046          INC      @DQTCR      ;SET TX GO..
1377 004316 005037 001244          CLR      TEMP1      ;DELAY COUNTER
1378 004322 012737 000005 001246    MOV      #5,TEMP2   ;.....
1379 004330 105777 175024    8$:     TSTB     @DQRCR      ;RX PRI DONE?
1380 004334 100407          BMI      9$         ;BR IF YES
1381 004336 005237 001244          INC      TEMP1      ;DELAY
1382 004342 001372          BNE      8$         ;.....
1383 004344 005337 001246          DEC      TEMP2      ;.....
1384 004350 001367          BNE      8$         ;.....
1385 004352 104000          HLT                     ;RX PRI DONE NOT SET.
1386 004354 005777 175006    9$:     TST      @DQERR   ;ANY ERRORS
1387 004360 100001          BPL      .+4        ;BR IF NO.
1388 004362 104000          HLT                     ;DQ11 ERROR FLAG SET.
1389 004364 011037 001254          MOV      (R0),TEMP5 ;SET EXPECTED
1390 004370 105077 174774          CLRB    @DQREG      ;SELECT RX BA PRI.
1391 004374 017701 174772          MOV      @DQSEC,R1  ;GET RX BA
1392 004400 162701 000002          SUB      #2,R1      ;GET LAST XFER
1393 004404 011137 001252          MOV      (R1),TEMP4 ;GET ACTUAL DATA
1394 004410 021037 001252          CMP      (R0),TEMP4 ;IS DATA OF?
1395 004414 001401          BEQ      .+4        ;BR IF GOOD
1396 004416 104006          HLT      6          ;DATA COMPARISON ERROR
1397
1398 004420 104401          SCOPI                     ;LOCK ON DATA (SW09=1)
1399 004422 005720          TST      (R0)+      ;UPDATE SOFTWARE POINTER
1400 004424 020037 004456          CMP      R0,LIMIT.HI ;ALL DONE?
1401 004430 001242          BNE      7$         ;BR IF NO
1402 004432 000410          BR      11$        ;END TEST
1403 004434 022626    10$:    POP2SP   ;ADJUST STACK POINTER
1404 004436 010537 000004          MOV      R5,@#4     ;RESET TRAP VECTOR
1405 004442 013737 001216 001214    MOV      NEXT,RETURN ;DO NEXT TEST
1406 004450 000177 174540          JMP      @RETURN    ;
1407 004454 104400    11$:    SCOPE                     ;SCOPE THIS TEST
1408
1409 004456          LIMIT.HI:
1410 004456 000000          0
1411

```

```

1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426 004460 012737 000005 001226
1427 004466 012737 005002 001216
1428 004474 032737 002000 001510
1429 004502 001005
1430 004504 012737 012364 001214
1431 004512 000177 174476
1432 004516 104412
1433 004520 104412
1434 004522 012737 000351 005000
1435 004530 112777 000017 174632
1436 004536 012777 000200 174626
1437 004544 012737 000011 015352
1438 004552 112777 000002 174610
1439 004560 012777 005000 174604
1440 004566 112777 000123 174574
1441 004574 012777 177777 174570
1442 004602 112777 000067 174560
1443 004610 005077 174556
1444 004614 112777 000012 174546
1445 004622 012777 004012 174542
1446 004630 005277 174530
1447 004634 027777 174524 174522
1448 004642 027777 174516 174514
1449 004650 027777 174510 174506
1450 004656 005277 174510
1451 004662 005377 174504
1452 004666 005277 174500
1453 004672 005377 174474
1454 004676 005337 015352
1455 004702 001371
1456 004704 112777 000016 174456
1457 004712 017705 174454
1458 004716 022705 000351
1459 004722 001401
1460 004724 104000
1461 004726 112777 000012 174434
1462 004734 012737 000010 015352
1463 004742 005277 174424
1464 004746 005377 174420
1465 004752 005337 015352
1466 004756 001371
1467 004760 112777 000016 174402

```

```

: TEST OF 'ENTER T' AND 'EXIT T'
: TRANSMITTER TRIGGERED
:
: TEST TO TRANSMITT ONE CHARACTER ENTERING T
: CHECKING THE BCC THEN OVERFLOWING
: CAUSING AN EXIT T THEN MAKING SURE
: THAT THE BCC WENT TO ZERO.
:
: NOTE: IF THE BCC DOES NOT EXIST THESE TESTS WILL NOT BE DONE

```

: TEST 5

:*****

```

TST5:  MOV      #5,TSTNO
      MOV      #TST6,NEXT
      BIT      #ABBIT,DQSTAT
      BNE      .+14
      MOV      #.EOP,RETURN
      JMP      @RETURN
      MSTCLR
      MSTCLR
      MOV      #351,WORD      :ISSUE A MASTER CLEAR
      MOV      #17,@DQREG    :SET CHAR
      MOV      #200,@DQSEC   :SELECT POLY REGISTER
      MOV      #11,COUNT     :SET POLY FOR LRC 8
      MOV      #2,@DQREG     :SET COUNT TO 11
      MOV      #WORD,@DQSEC  :SET TX BA PRI
      MOV      #123,@DQREG   :LOAD TX BA
      MOV      #-1,@DQSEC    :SEL TX CC-WRITE EN ENTER T
      MOV      #67,@DQREG    :SEL TX CC TO -1
      CLR      @DQSEC        :SEL TX CC SEC-WRITE EN EXIT T
      MOV      #MISC.,@DQREG :SET TX CC TO ZERO
      INC      @DQSEC        :SEL MISC REG
      INC      @DQSEC        :EIGHT BITS TEST LOOP AUTO SET
      INC      @DQTCR        :SET TX GO
      CMP      @DQTCR,@DQTCR :WAIST TIME.
      CMP      @DQTCR,@DQTCR :WAIST TIME
      CMP      @DQTCR,@DQTCR :WAIST TIME
      INC      @DQSEC        :CLOCK UP
      DEC      @DQSEC        :CLOCK DOWN
      INC      @DQSEC        :START THE CHAR UP
      DEC      @DQSEC        :DOWN
      DEC      COUNT
      BNE      1$           :DONE YET?
      MOV      #16,@DQREG   :BR IF NO
      MOV      @DQSEC,R5    :SET TX BCC REG
      CMP      #351,R5     :STORE IT
      BEQ      .+4         :DID CHAR GET INTO BCC
      HLT
      MOV      #MISC.,@DQREG :TX BCC FAILED
      MOV      #10,COUNT    :SEL MISC REG
      INC      @DQSEC       :SET COUNT TO 10
      DEC      @DQSEC       :START CLOCKING BCC OUT UP
      DEC      COUNT       :DOWN
      BNE      2$
      MOV      #16,@DQREG   :DONE YET
                          :BR IF NO
                          :SEL TX BCC

```

1\$:

2\$:

1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579

005130 012737 000007 001226
005136 012737 005176 001214
005144 012737 005636 001216
005152 104413
005154 005000
005156 012704 015376
005162 110024
005164 105200
005166 001375
005170 012737 177777 015350
005176 104412
005200 104412
005202 005037 015344
005206 012737 000016 001254
005214 005037 015354
005220 112777 000022 174142
005226 012777 015376 174136
005234 112777 000123 174126
005242 012777 177400 174122
005250 112777 000067 174112
005256 005077 174110
005262 112777 000017 174100
005270 013777 015350 174074
005276 112777 000012 174064
005304 012777 004012 174060
005312 005277 174046
005316 027777 174042 174040
005324 027777 174034 174032
005332 027777 174026 174024
005340 005277 174026
005344 005377 174022
005350 005277 174016
005354 005377 174012
005360 005037 015352
005364 013737 015354 005406
005372 013737 015344 005410
005400 004537 012064
005404 000001
005406 000001
005410 000001
005412 112777 000012 173750
005420 005277 173746
005424 005377 173742
005430 112777 000016 173732

```
;DATA PATTERN IS RUN THROUGH  
;THE BCC WITH A SHIFT BY SHIFT  
;CHECK OF THE HARDWARE BY THE SOFTWARE.  
;AT THE END THE TRANSMITTER IS ALSO  
;CHECKED TO SEE IF THE BCC WAS SHIFTED  
;OUT AND THAT THE BCC WENT TO ZERO.  
;  
;NOTE: THERE IS A TWO SHIFT DELAY  
;BEFORE THE TX BCC STARTS.  
;  
: TEST ?  
:*****  
TST7: MOV #7,TSTNO  
MOV #2$,RETURN  
MOV #TST10,NEXT  
MEMCLR ;CLEAR ALL THE DQ11  
CLR R0 ;SET POINTER TO ZERO  
MOV #TXBUFF,R4 ;GET TX BUFFER  
1$: MOVB R0,(R4)+ ;START FILLING TX BUFFER  
INCB R0 ;WITH A BINARY  
BNE 1$ ;COUNT PATTERN  
MOV #177777,XPOLY ;SET POLYNOMIAL TO 177777  
2$: MSTCLR ;ISSUE MASTER CLEAR  
MSTCLR  
CLR CALBCC ;SET CALCULATED BCC TO ZERO  
MOV #16,TEMP5 ;SET TYPE OUT ERROR REG TO 16  
CLR DATA ;SET DATA OF BCC SIMULATOR TO ZERO  
MOVB #22,@DQREG ;WRITE EN TX BA PRI  
MOV #TXBUFF,@DQSEC ;SET TX BUFFER  
MOVB #123,@DQREG ;ENTER T WRITE EN ,TX CC PRI  
MOV #-400,@DQSEC ;SET FOR 400 CHARS  
MOVB #67,@DQREG ;EXIT 'T',WRITE EN,TX CC SEC  
CLR @DQSEC ;SET FOR ZERO CHARS ON SEC  
MOVB #17,@DQREG ;SEL POLYNOMIAL REGISTER  
MOV XPOLY,@DQSEC ;LOAD IT  
MOVB #MISC.,@DQREG ;SEL MISC REGISTER  
MOV #4012,@DQSEC ;EIGHT BITS,TEST LOOP,AUTO STEP  
INC @DQTCR ;SET TX GO  
CMP @DQTCR,@DQTCR ;WAIST TIME.  
CMP @DQTCR,@DQTCR ;WAIST TIME  
CMP @DQTCR,@DQTCR ;WAIST TIME  
INC @DQSEC ;CLOCK UP---  
DEC @DQSEC ;CLOCK DOWN---  
INC @DQSEC ;CLOCK UP---  
DEC @DQSEC ;CLOCK DOWN---  
3$: CLR COUNT ;SET COUNT TO 0  
MOV DATA,6$ ;SET DATA FOR SUBROUTINE  
4$: MOV CALBCC,7$ ;SET CALCULATED BCC FOR SUB ROUTINE  
JSR R5,SIMBCC ;GO TO BCC SIMULATOR ROUTINE  
5$: 1 ;THIS IS THE NUMBER OF SHIFTS FOR ROUTINE TO DO  
6$: .BLKW 1 ;THIS IS WHERE THE CHAR IS PLACED  
7$: .BLKW 1 ;THIS IS THE PREVIOUS BCC CALCULATED  
MOVB #MISC.,@DQREG ;RESELECT THE MISC REG  
INC @DQSEC ;CLOCK UP---  
DEC @DQSEC ;CLOCK DOWN---  
MOVB #16,@DQREG ;SEL THE TX BCC REGISTER
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1580 005436 017737 173730 001252      MOV    @DQSEC,TEMP4      ;SAVE IT IN TEMP4
1581 005444 023737 015344 001252      CMP    CALBCC,TEMP4     ;ARE THE CALCULATED AND RECEIVED RESULTS THE SAME??
1582 005452 001401                BEQ    .+4               ;BR IF GOOD (SAME)
1583 005454 104003                HLT    3                 ;BCC(S) ARE DIFFERNT..
1584 005456 000241                CLC                    ;CLEAR THE CARRY BIT OF PSW
1585 005460 006037 005406      ROR    6$               ;UPDATE MY DATA LOCATION
1586 005464 005237 015352      INC    COUNT            ;UPDATE THE COUNT OF BITS PER CHAR
1587 005470 023727 015352 000010  CMP    COUNT,#8.        ;IS THIS CHARACTER DONE YET??
1588 005476 001335                BNE    4$               ;BR IF CHAR NOT DONE
1589 005500 105237 015354      INCB   DATA            ;GET NEW CHAR
1590 005504 001325                BNE    3$               ;HAVE ALL CHARACTERS BEEN DONE
1591
1592 005506 005037 015352      CLR    COUNT            ;INIT COUNT
1593 005512 112777 000012 173650 8$:  MOVB   #MISC.,@DQREG    ;RESELECT THE MISC REG
1594 005520 000241                CLC                    ;CLEAR CARRY
1595 005522 006037 015344      ROR    CALBCC           ;SHIFT OUT CALCULATED BCC
1596 005526 005277 173640      INC    @DQSEC           ;CLOCK UP---
1597 005532 005377 173634      DEC    @DQSEC           ;CLOCK DOWN---
1598 005536 112777 000016 173624  MOVB   #16,@DQREG       ;SEL TX BCC REGISTER
1599 005544 017737 173622 001252  MOV    @DQSEC,TEMP4     ;SAVE IT IN TEMP4
1600 005552 023737 015344 001252  CMP    CALBCC,TEMP4     ;ARE THEY THE SAME??
1601 005560 001401                BEQ    .+4               ;BR IF SAME(GOOD)
1602 005562 104003                HLT    3                 ;BCC DIFFERENT
1603 005564 005237 015352      INC    COUNT            ;UPDATE COUNT
1604 005570 022737 000020 015352  CMP    #16.,COUNT     ;HAS ALL THE BCC BEEN SHIFED OUT
1605 005576 001345                BNE    8$               ;BR IF MORE TO DO
1606 005600 112777 000012 173562  MOVB   #MISC.,@DQREG    ;SELECT THE MISC REGISTER
1607 005606 005277 173560      INC    @DQSEC           ;GIVE ONE LAST CLOCK UP--
1608 005612 005377 173554      DEC    @DQSEC           ;AND CLOCK DOWN--
1609 005616 112777 000016 173544  MOVB   #16,@DQREG       ;SEL THE TX BCC REGISTER
1610 005624 005777 173542      TST    @DQSEC           ;DID THE TX BCC GO TO ZERO
1611 005630 001401                BEQ    .+4               ;BR IF GOOD
1612 005632 104000                HLT                    ;TX BCC NOT ZERO
1613 005634 104400                SCOPE                    ;SCOPE TEST
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632

```

```

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

```

```

: TEST 10
:*****
TST10: MOV    #10,TSTNO
        MOV    #2$,RETURN
        MOV    #TST11,NEXT

```

```

1633 005636 012737 000010 001226
1634 005644 012737 005676 001214
1635 005652 012737 006662 001216

```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

1636	005660	104413			1\$:	MEMCLR		:CLEAR THE DEVICE
1637	005662	012737	177777	015350		MOV	#177777,XPOLY	:SELXPOLY TO 177777
1638	005670	012737	000015	001254		MOV	#15,TEMP5	:SET TYPE OUT REG TO 15
1639	005676	104412			2\$:	MSTCLR		:ISSUE A MASTER CLEAR
1640	005700	104412				MSTCLR		
1641	005702	005037	015344			CLR	CALBCC	:SET CALBCC TO ZERO
1642	005706	005037	015354			CLR	DATA	:SET DATA TO ZERO
1643	005712	112777	000020	173450		MOV	#20,@DQREG	:WRITE EN,RX BA PRI
1644	005720	012777	016000	173444		MOV	#RXBUFF,@DQSEC	:LOAD THE RX BA
1645	005726	112777	000121	173434		MOV	#121,@DQREG	:ENTER 'T',WRITE EN,RX CC PRI
1646	005734	012777	177400	173430		MOV	#-400,@DQSEC	:SET FOR FOUR HUNDRED CHARS
1647	005742	112777	000024	173420		MOV	#24,@DQREG	:SEL THE RX BA SEC
1648	005750	012777	016000	173414		MOV	#RXBUFF,@DQSEC	:SET FOR THE PAD CHAR.
1649	005756	112777	000065	173404		MOV	#65,@DQREG	:EXIT 'T',WRITE EN,RX CC SEC
1650	005764	012777	177777	173400		MOV	#-1,@DQSEC	:SEL RX CC SEC FOR ONE PAD CHAR.
1651	005772	112777	000017	173370		MOV	#17,@DQREG	:SEL THE POLYNO REGISTER
1652	006000	013777	015350	173364		MOV	XPOLY,@DQSEC	:LOAD IT WITH THE POLY
1653	006006	112777	000012	173354		MOV	#MISC.,@DQREG	:SEL THE MISC REGISTER
1654	006014	012777	004012	173350		MOV	#4012,@DQSEC	:EIGHT BITS TEST LOOP AND AUTO STEP
1655	006022	012777	010001	173330		MOV	#10001,@DQRCR	:SET RX ACTIVE AND GO!!
1656	006030	012737	000010	015352	3\$:	MOV	#8,COUNT	:SET FOR ONE CHAR TIME DELAY
1657	006036	013737	015354	015340		MOV	DATA,TMPDAT	:SAVE THE DATA
1658	006044	005137	015340			COM	TMPDAT	:COMPLIMENT IT FOR BIT WINDOW USE.
1659	006050	005037	001250		4\$:	CLR	TEMP3	:INIT LOC
1660	006054	006037	015340			ROR	TMPDAT	:SHIFT OUT ONE BIT OF DATA
1661	006060	106037	001250			RORB	TEMP3	:BRING IT IN FROM CARRY
1662	006064	042777	000200	173300		BIC	#BIT7,@DQSEC	:CLEAR THE BIT WINDOW
1663	006072	053777	001250	173272		BIS	TEMP3,@DQSEC	:PLACE DATA ON BIT WINDOW
1664	006100	005277	173266			INC	@DQSEC	:CLOCK UP---
1665	006104	005377	173262			DEC	@DQSEC	:CLOCK DN---
1666	006110	005337	015352			DEC	COUNT	:CHAR DONE??
1667	006114	001355				BNE	4\$:BR IF NOT DONE
1668	006116	105237	015354			INCB	DATA	:UPDATE DATA
1669	006122	005037	015352		5\$:	CLR	COUNT	:INIT COUNT
1670	006126	013737	015354	006166		MOV	DATA,8\$:MOV DATA TO SUB ROUTINE USE AREA
1671	006134	005337	006166			DEC	8\$:SET SUBROUTINE TO ONE LESS THAT RX GETS
1672	006140	013737	015354	015340		MOV	DATA,TMPDAT	:SAVE DATA
1673	006146	005137	015340			COM	TMPDAT	:COMPLIMENT DATA FOR BIT WINDOW USE
1674	006152	013737	015344	006170	6\$:	MOV	CALBCC,9\$:MOV CALCULATED BCC TO SUB ROUTINE USE
1675	006160	004537	012064			JSR	R5,SIMBCC	:GO AND CALCULATE BCC (SOFTWARE)
1676	006164	000001			7\$:	1		:THIS IS NUMBER OF SHIFTS TO BE DONE
1677	006166	000001			8\$:	.BLKW	1	:THIS IS WHERE THE DATA IS PLACED
1678	006170	000001			9\$:	.BLKW	1	:THIS IS WHERE THE PREVIOUS BCC IS PLACED
1679	006172	112777	000012	173170		MOV	#MISC.,@DQREG	:RESELECT THE MISC REGISTER
1680	006200	005037	001250			CLR	TEMP3	:INIT LOC
1681	006204	006037	015340			ROR	TMPDAT	:SHIFT OUT DATA BIT
1682	006210	106037	001250			RORB	TEMP3	:CATCH IT IN TEMP3
1683	006214	042777	000200	173150		BIC	#BIT7,@DQSEC	:CLEAR THE BIT WINDOW
1684	006222	053777	001250	173142		BIS	TEMP3,@DQSEC	:LOAD THE DATA
1685	006230	005277	173136			INC	@DQSEC	:CLOCK UP---
1686	006234	005377	173132			DEC	@DQSEC	:CLOCK DN---
1687	006240	112777	000015	173122		MOV	#15,@DQREG	:SEL RX BCC REGISTER
1688	006246	017737	173120	001252		MOV	@DQSEC,TEMP4	:SAVE THE BCC
1689	006254	023737	015344	001252		CMF	CALBCC,TEMP4	:IS IT CORRECT??
1690	006262	001401				BEQ		:BR IF GOOD
1691	006264	104003				HLT	3	:BCC NOT WHAT EXPECTED

1692	006266	000241				CLC			:CLEAR THE CARRY BIT
1693	006270	006037	006166			ROR	8\$:SHIFT THE DATA
1694	006274	005237	015352			INC	COUNT		:UPDATE THE COUNT
1695	006300	022737	000010	015352		CMP	#8.,COUNT		:IS THE CHARACTER DONE??
1696	006306	001321				BNE	6\$:BR IF CHAR NOT DONE
1697	006310	105237	015354			INCB	DATA		:UPDATA DATA
1698	006314	001302				BNE	5\$:BR IF NOT ALL CHARS DONE.
1699	006316	012737	000003	015370	10\$:	MOV	#3,LOC1		:POINTER****
1700	006324	013737	015344	006350		MOV	CALBCC,21\$:SAVE CALBCC
1701	006332	013737	015344	015366		MOV	CALBCC,STORE1		:
1702	006340	004537	012064			JSR	R5,SIMBCC		:GO FINISH THE BCC
1703	006344	000010				8.			:SHIFTS REQUIRED
1704	006346	000377				377			:DATA CHARACTER
1705	006350	000001			21\$:	.BLKW	1		:PREVIOUS BCC
1706	006352	013737	015344	015356		MOV	CALBCC,SAVBCC		:SAVE THE BCC
1707	006360	013737	015366	015344		MOV	STORE1,CALBCC		:RESTORE THE BCC
1708	006366	012737	000377	015354		MOV	#377,DATA		:DATA =377
1709	006374	013737	015356	015340		MOV	SAVBCC,TMPDAT		:PUSH IN THE LOW BYTE OF THE BCC
1710	006402	005137	015340			COM	TMPDAT		:INTO THE RX
1711	006406	000421				BR	12\$:
1712	006410	013737	015356	015354	11\$:	MOV	SAVBCC,DATA		:MOVE THE CALBCC TO DATA FOR SUBROUTINE
1713	006416	113737	015357	015340		MOVB	SAVBCC+1,TMPDAT		:MOVE THE HIGH BYTE OF CALBCC TO PLACED INTO THE RECEIVE
1714	006424	005137	015340			COM	TMPDAT		:PREPARE IT FOR THE BIT WINDOW
1715	006430	000410				BR	12\$:GO TO MAIN PART OF TEST
1716	006432	113737	015357	015354	16\$:	MOVB	SAVBCC+1,DATA		:MOVE THE HIGH BYTE OF THE CALBCC TO DATA
1717	006440	012737	000377	015340		MOV	#377,TMPDAT		:PAD CHAR FOR RX
1718	006446	005137	015340			COM	TMPDAT		:PREPARE IT FOR THE BIT WINDOW
1719	006452	005037	015352		12\$:	CLR	COUNT		:INIT COUNT (THIS IS FOR NUMBER OF BITS PER CHAR)
1720	006456	113737	015354	006500		MOVB	DATA,13\$:LOAD THE CHAR FOR SUBROUTINE
1721	006464	013737	015344	006502	15\$:	MOV	CALBCC,14\$:LOAD THE CALBCC FOR THE SUBROUTINE
1722	006472	004537	012064			JSR	R5,SIMBCC		:GO TO THE SUBROUTINE
1723	006476	000001				1			:THIS IS THE NUMBER OF SHIFTS TO BE DONE
1724	006500	000001			13\$:	.BLKW	1		:THIS IS THE CHAR FOR THE SUBROUTINE
1725	006502	000001			14\$:	.BLKW	1		:THIS IS THE PREVIOUS BCC
1726	006504	112777	000012	172656		MOVB	#MISC.,@DQREG		:SEL THE MISC REGISTER
1727	006512	005037	001250			CLR	TEMP3		:INIT LOC
1728	006516	006037	015340			ROR	TMPDAT		:SHIFT OUT A BIT OF DATA
1729	006522	106037	001250			RORB	TEMP3		:BRING IT FROM CARRY INTO TEMP3
1730	006526	042777	000200	172636		BIC	#BIT7,@DQSEC		:CLEAR THE BIT WINDOW
1731	006534	053777	001250	172630		BIS	TEMP3,@DQSEC		:PLACE DATA ON THE BIT WINDOW
1732	006542	005277	172624			INC	@DQSEC		:CLOCK UP---
1733	006546	005377	172620			DEC	@DQSEC		:CLOCK DN---
1734	006552	112777	000015	172610		MOVB	#15,@DQREG		:SEL THE RX BCC REGISTER
1735	006560	017737	172606	001252		MOV	@DQSEC,TEMP4		:SAVE IT IN TEMP4
1736	006566	023737	015344	001252		CMP	CALBCC,TEMP4		:IS THE BCC CORRECT??
1737	006574	001401				BEQ	,+4		:BR IF GOOD
1738	006576	104003				HLT	3		:BCC ERROR IN RX
1739	006600	000241				CLC			:CLEAR THE CARRY BIT
1740	006602	006037	006500			ROR	13\$:SHIFT THE CHARACTER TO THE RIGHT
1741	006606	005237	015352			INC	COUNT		:UPDATE THE COUNT
1742	006612	022737	000010	015352		CMP	#8.,COUNT		:IS THIS CHARACTER DONE
1743	006620	001321				BNE	15\$:BR IF NOT DONE
1744	006622	005337	015370			DEC	LOC1		:ALTER THE RETURN POINTER
1745	006626	022737	000002	015370		CMP	#2,LOC1		:WHERE SHOULD I GO??
1746	006634	001665				BEQ	1:\$:IF LOC1=2 GOTO 11\$
1747	006636	022737	000001	015370		CMP	#1,LOC1 ;		

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1748 006644 001672          BEQ      16$          :IF LOC1=1 GOTO 16$
1749 006646 017705 172514  MOV      @DQERR,R5    :SAVE THE ERROR REGISTER
1750 006652 005705          TST      R5          :DID AN ERROR OCCUR??
1751 006654 100001          BPL      .+4         :BR IF NO ERROR
1752 006656 104001          HLT      1          :DQ11 ERROR FLAG SET
1753 006660 104400          SCOPE                    :SCOPE THIS TEST
1754
1755

```

```

1756          :TEST OF TRANSMITTER AND RECEIVER
1757          :BCC WITH A POLYNOMIAL OF
1758          :CRC 16: X16+X15+X2+1
1759          :NOTE: IN THIS TEST IT IS UP TO
1760          :      THE HARDWARE TO DISCOVER
1761          :      AN ERROR IF ONE OCCURS.
1762

```

: TEST 11

```

1764          :*****
1765 006662 012737 000011 001226 TST11: MOV      #11,TSTNO
1766 006670 012737 006726 001216      MOV      #TST12,NEXT
1767 006676 104412          MSTCLR                    :CLEAR DQ11 WITH A MASTER CLEAR
1768 006700 012737 120001 015350      MOV      #120001,XPOLY :LOAD SELECTED POLYNOMIAL INTO XPOLY.
1769 006706 004737 010202          JSR      PC,STBCC       :TRANSFER CHARACTERS.
1770 006712 017705 172450          MOV      @DQERR,R5    :SAVE THE ERROR REGISTER
1771 006716 005705          TST      R5          :DID AN ERROR OCCUR??
1772 006720 100001          BPL      .+4         :BR IF NO ERROR
1773 006722 104001          HLT      1          :AN ERROR OCCURED
1774 006724 104400          SCOPE
1775
1776
1777
1778
1779

```

```

1780          :
1781          :TEST OF TRANSMITTER AND RECEIVER
1782          :BCC WITH A POLYNOMIAL OF
1783          :CRC 12: X12+X11+X3+X2+X+1
1784          :NOTE: IN THIS TEST IT IS UP TO
1785          :      THE HARDWARE TO DISCOVER
1786          :      AN ERROR IF ONE OCCURS
1787

```

: TEST 12

```

1788          :*****
1789 006726 012737 000012 001226 TST12: MOV      #12,TSTNO
1790 006734 012737 006772 001216      MOV      #TST13,NEXT
1791 006742 104412          MSTCLR                    :CLEAR DQ11 WITH A MASTER CLEAR
1792 006744 012737 007401 015350      MOV      #7401,XPOLY  :LOAD SELECTED POLYNOMIAL INTO XPOLY.
1793 006752 004737 010202          JSR      PC,STBCC       :TRANSFER CHARACTERS.
1794 006756 017705 172404          MOV      @DQERR,R5    :SAVE THE ERROR REGISTER
1795 006762 005705          TST      R5          :DID AN ERROR OCCUR??
1796 006764 100001          BPL      .+4         :BR IF NO ERROR
1797 006766 104001          HLT      1          :AN ERROR OCCURED
1798 006770 104400          SCOPE
1799
1800
1801
1802
1803

```

```

1804 ;TEST OF TRANSMITTER AND RECEIVER
1805 ;BCC WITH A POLYNOMIAL OF
1806 ;CRC/CCITT: X16+X12+X2+X+1
1807 ;NOTE: IN THIS TEST IT IS UP TO
1808 ;THE HARDWARE TO DISCOVER
1809 ;AN ERROR IF ONE OCCURES
1810

```

: TEST 13

```

1811 :*****
1812 TST13: MOV #13,TSTNO
1813 006772 012737 000013 001226 MOV #TST14,NEXT
1814 007000 012737 007036 001216 MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR
1815 007006 104412 JSR #102010,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
1816 007010 012737 102010 015350 JSR PC,STBCC ;TRANSFER CHARACTERS.
1817 007016 004737 010202 MOV @DQERR,R5 ;SAVE THE ERROR REGISTER
1818 007022 017705 172340 TST R5 ;DID AN ERROR OCCUR??
1819 007026 005705 BPL +4 ;BR IF NO ERROR
1820 007030 100001 HLT 1 ;AN ERROR OCCURED
1821 007032 104001 SCOPE
1822 007034 104400
1823
1824
1825
1826
1827

```

```

1828 ;TEST OF TRANSMITTER AND RECEIVER
1829 ;BCC WITH A POLYNOMIAL OF
1830 ;LRC 8: X8+1
1831 ;NOTE: IN THIS TEST IT IS UP TO
1832 ;THE HARDWARE TO DISCOVER
1833 ;AN ERROR IF ONE OCCURES.
1834

```

: TEST 14

```

1835 :*****
1836 TST14: MOV #14,TSTNO
1837 007036 012737 000014 001226 MOV #TST15,NEXT
1838 007044 012737 007102 001216 MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR
1839 007052 104412 JSR #200,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
1840 007054 012737 000200 015350 JSR PC,STBCC ;TRANSFER CHARACTERS.
1841 007062 004737 010202 MOV @DQERR,R5 ;SAVE THE ERROR REGISTER
1842 007066 017705 172274 TST R5 ;DID AN ERROR OCCUR??
1843 007072 005705 BPL +4 ;BR IF NO ERROR
1844 007074 100001 HLT 1 ;AN ERROR OCCURED
1845 007076 104001 SCOPE
1846 007100 104400
1847
1848
1849
1850

```

```

1851 ;TEST OF TRANSMITTER AND RECEIVER
1852 ;BCC WITH A POLYNOMIAL OF
1853 ;LRC 16: X16+1
1854 ;NOTE: IN THIS TEST IT IS UP TO
1855 ;THE HARDWARE TO DISCOVER
1856 ;AN ERROR IF ONE OCCURES
1857

```

: TEST 15

1859

```

1860
1861 007102 012737 000015 001226 TST15: MOV #15,TSTNO
1862 007110 012737 007146 001216 MOV #TST16,NEXT
1863 007116 104412 MSTCLR ;CLEAR DQ11 WITH A MASTER CLEAR
1864 007120 012737 100000 015350 MOV #100000,XPOLY ;LOAD SELECTED POLYNOMIAL INTO XPOLY.
1865 007126 004737 010202 JSR PC,STBCC ;TRANSFER CHARACTERS.
1866 007132 017705 172230 MOV @DQERR,R5 ;SAVE THE ERROR REGISTER
1867 007136 005705 TST R5 ;DID AN ERROR OCCUR??
1868 007140 100001 BPL .+4 ;BR IF NO ERROR
1869 007142 104001 HLT 1 ;AN ERROR OCCURED
1870 007144 104400 SCOPE

```

```

1871
1872
1873
1874
1875
1876 ;
1877 ;TEST OF RECEIVER AND TRANSMITTER
1878 ;BCC USING CRC 16 FOR POLYNOMIAL.
1879 ;
1880 ;THIS TEST USES IDLE MODE TO
1881 ;GET INTO TRANSPARENCY
1882 ;AND IF AN ERROR SHOULD OCCUR
1883 ;IT MUST BE REPORTED BY THE HARDWARE.
1884 ;

```

```

1885 ; TEST 16
1886 ;*****
1887 007146 012737 000016 001226 TST16: MOV #16,TSTNO
1888 007154 012737 007206 001214 MOV #2$,RETURN
1889 007162 012737 007242 001216 MOV #TST17,NEXT
1890 007170 104413 MEMCLR ;CLEAR THE DEVICE
1891 007172 005000 CLR R0 ;SET R0 TO ZERO
1892 007174 012704 015376 MOV #TXBUFF,R4 ;SET POINTER FOR BUFFER
1893 007200 110024 1$: MOVB R0,(R4)+ ;START FILLING THE BUFFER
1894 007202 105200 INCB R0 ;UPDATE THE DATA
1895 007204 001375 BNE 1$ ;HAS THE BUFFER BEEN FILLED
1896 007206 104412 2$: MSTCLR
1897 007210 012737 120001 015350 MOV #120001,XPOLY ;SELECT CRC 16 FOR POLYNOMIAL
1898 007216 004537 010570 JSR R5,SYNBCC ;GO PRIM THE DQ11
1899 007222 177400 -400 ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER
1900 007224 177400 -400 ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER
1901 007226 017705 172134 MOV @DQERR,R5 ;SAVE THE DQ ERROR REGISTER
1902 007232 005705 TST R5 ;DID AN ERROR OCCUR??
1903 007234 100001 BPL .+4 ;BR IF THE ERROR DID NOT OCCUR
1904 007236 104001 HLT 1 ;HALT THE DQ ERROR BIT IS SET
1905 007240 104400 SCOPE ;SCOPE THIS TEST
1906
1907

```

```

1908
1909
1910 ;
1911 ;TEST OF TRANSMITTER AND RECEIVER
1912 ;BCC WITH A POLYNOMIAL OF 177777
1913 ;
1914 ;THIS TEST USES IDLE MODE TO
1915 ;GET INTO TRANSPARENCY. IF AN
;ERROR SHOULD HAPPEN, THE HARDWARE

```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```
1916                                     ;MUST FLAG IT.
1917                                     .
1918                                     ; TEST 17
1919                                     ;*****
1920 007242 012737 000017 001226 TST17: MOV #17,TSTNO
1921 007250 012737 007302 001214      MOV #2$,RETURN
1922 007256 012737 007336 001216      MOV #TST20,NEXT
1923 007264 104413      MEMCLR      ;CLEAR THE DEVICE
1924 007266 005000      CLR R0      ;SET R0 TO ZERO
1925 007270 012704 015376      MOV #TXBUFF,R4 ;SET POINTER FOR BUFFER
1926 007274 110024 1$: MOVB R0,(R4)+ ;START FILLING THE BUFFER
1927 007276 105200      INCB R0      ;UPDATE THE DATA
1928 007300 001375      BNE 1$      ;HAS THE BUFFER BEEN FILLED
1929 007302 104412 2$: MSTCLR      ;ISSUE MASTER CLEAR
1930 007304 012737 177777 015350      MOV #177777,XPOLY ;SELECT 177777 FOR POLYNOMIAL
1931 007312 004537 010570      JSR R5,SYNBCC ;GO PRIM THE DQ11
1932 007316 177400      -400      ;THIS IS THE CHARACTER COUNT FOR THE RECEIVER
1933 007320 177400      -400      ;THIS IS THE CHARACTER COUNT FOR THE TRANSMITTER
1934 007322 017705 172040      MOV @DQERR,R5 ;SAVE THE DQ ERROR REGISTER
1935 007326 005705      TST R5      ;DID AN ERROR OCCUR??
1936 007330 100001      BPL +4      ;BR IF THE ERROR DID NOT OCCUR
1937 007332 104001      HLT 1      ;HALT THE DQ ERROR BIT IS SET
1938 007334 104400      SCOPE      ;SCOPE THIS TEST
1939
1940                                     .
1941                                     ;TEST OF TRANSMITTER AND RECEIVER
1942                                     ;BCC WITH ALL POLYNOMIALS
1943                                     ;BETWEEN 000000 AND 177777
1944                                     ;SENDING ONE CHARACTER (351)
1945                                     ;AND CALCULATING WHAT THE BCC
1946                                     ;SHOULD BE AND COMPARING IT
1947                                     ;WITH THE TRANSMITTER BCC AND RECEIVER BCC
1948                                     .
1949                                     ;NOTE: SW 09=1 WILL FREEZE THE POLYNOMIAL
1950                                     ;WHEN SW 09 IS ASSERTED.
1951
1952                                     ; TEST 20
1953                                     ;*****
1954 007336 012737 000020 001226 TST20: MOV #20,TSTNO
1955 007344 012737 000003 001222      MOV #3,ICOUNT
1956 007352 012737 007472 001216      MOV #TST21,NEXT
1957 007360 012737 007372 001220      MOV #1$,LOCK
1958 007366 005037 015350      CLR XPOLY      ;INIT SET XPOLY TO ZERO
1959 007372 104412 1$: MSTCLR      ;GO AND PRIM THE DQ11
1960 007374 004737 011110      JSR PC,TYBCC ;GO AND CALCULATE WHAT THE BCC SHOULD BE
1961 007400 004537 012064      JSR R5,SIMBCC ;THIS IS FOR EIGHT BITS PER CHAR (NUMBER OF SHIFTS)
1962 007404 000010      8.          ;THIS IS THE DATA CHARACTER
1963 007406 000351      351        ;THIS IS THE PREVIOUS BCC
1964 007410 000000      0          ;SEL THE RX BCC REGISTER
1965 007412 112777 000015 171750      MOVB #15,@DQREG ;SAVE THE REGISTER
1966 007420 012705 000015      MOV #15,R5      ;SAVE THE BCC
1967 007424 017701 171742      MOV @DQSEC,R1 ;SAVE THE BCC
1968 007430 023701 015344      CMP CALBCC,R1 ;IS THE BCC RIGHT??
1969 007434 001401      BEQ +4      ;BR IF YES
1970 007436 104002      HLT 2      ;BCC ERROR
1971 007440 012705 000016      MOV #16,R5      ;SEL THE TX BCC REGISTER
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

1972 007444 013701 015362      MOV      SEC16,R1      ;THIS IS WHERE THE TX BCC WAS STORED
1973 007450 023701 015344      CMP      CALBCC,R1    ;IS IT RIGHT??
1974 007454 001401              BEQ      .+4           ;BR IF GOOD
1975 007456 104002              HLT      2             ;TX BCC ERROR
1976 007460 104401              SCOP1    ;DOES THE USER HAVE SW09=1??
1977 007462 005237 015350      INC      XPOLY        ;UPDATE THE POLYNO
1978 007466 001341              BNE      1$           ;BE IF NOT ALL POLYNO HAVE BEEN DONE
1979 007470 104400              SCOPE    ;SCOPE THIS TEST

```

```

1980
1981
1982
1983      ;
1984      ;TEST OF BIT 06 OF MISC REGISTER
1985      ;POLYNOMIAL 16-24
1986      ;
1987      ;TEST WILL SEND ONE CHARACTER AT
1988      ;A TIME CHECKING THAT THE BCC
1989      ;OF RECEIVER AND TRANSMITTER
1990      ;ARE CORRECT.
1991      ;CHARACTER SENT: 000-377
1992
1993
1994

```

```

: TEST 21
:*****

```

```

1995 007472 012737 000021 001226  TST21: MOV      #21,TSTNO
1996 007500 012737 012364 001216  MOV      #.EOP,NEXT
1997      ;PART 1      READ/WRITE TEST OF POLY 16-24
1998      ;TEST OF ALL READ WRITE BITS IN POLY 16-24
1999      ;BY RUNNING A BINARY COUNT PATTERN TROUGH
2000      ;THE REGISTER.

```

```

2003 007506 012737 007520 001220      MOV      #1$,LOCK    ;SET FOR LOCK ON TEST(SW09=1)
2004 007514 005037 001254              CLR      TEMP5       ;ZERO POINTER
2005 007520 104412              MSTCLR   ;INIT DQ11
2006 007522 112777 000012 171640 1$:  MOVB    #MISC.,@DQREG ;SEL MISC REG
2007 007530 012777 000100 171634  MOV     #BIT6,@DQSEC ;SEL POLY 16-24
2008 007536 112777 000017 171624  MOVB    #17,@DQREG  ;SEL POLY REGISTER
2009 007544 053777 001254 171620  BIS     TEMP5,@DQSEC ;LOAD WITH CHAR.
2010 007552 017737 171614 001252  MOV     @DQSEC,TEMP4 ;READ CHAR BACK.
2011 007560 023737 001254 001252  CMP     TEMP5,TEMP4 ;IS IT CORRECT?
2012 007566 001401              BEQ      .+4         ;BR IF YES.
2013 007570 104006              HLT      6           ;POLY READ/WRITE ERROR.
2014 007572 104401              SCOP1    ;LOCK ON CHAR (SW09=1)
2015 007574 105237 001254      INCB    TEMP5       ;UPDATE CHAR.
2016 007600 001347              BNE      1$         ;BR IF MORE TO GO

```

;PART 2 RX AND TX BCC TESTS

```

2020 007602 005037 015344              CLR      CALBCC      ;ZERO EXPECTED BCC
2021 007606 012737 007614 001220  MOV     #2$,LOCK    ;SET FOR SW09=1
2022 007614 104412              MSTCLR   ;INIT DQ11
2023 007616 012737 000200 015350  MOV     #200,XPOLY  ;SEL 'LRC 24''
2024 007624 004737 011456              JSR     PC,TXBCC    ;GOTO SUBROUTINE
2025 007630 012705 000015              MOV     #15,R5      ;SEL BCC REG
2026 007634 112777 000015 171526  MOVB    #15,@DQREG  ;
2027 007642 017701 171524      MOV     @DQSEC,R1   ;READ BCC REG.

```

```

2028 007646 023701 015344      CMP      CALBCC,R1      ;IS BCC CORRECT?
2029 007652 001401              BEQ      .+4           ;BR IF GOOD.
2030 007654 104002              HLT      2            ;BCC ERROR.
2031 007656 012705 000016      MOV      #16,R5       ;SEL BCC REG.
2032 007662 013701 015362      MOV      SEC16,R1     ;GET SAVED BCC
2033 007666 023701 015344      CMP      CALBCC,R1     ;DID IT COMPARE?
2034 007672 001401              BEQ      .+4           ;BR IF GOOD.
2035 007674 104002              HLT      2            ;BCC ERROR
2036 007676 013704 015344      MOV      CALBCC,R4    ;SAVE GOOD BCC
2037 007702 005037 015344      CLR      CALBCC       ;ZERO SOFTWARE BCC
2038 007706 112777 000012 171454      MOVB     #MISC.,@DQREG ;SEL MISC REGISTER
2039 007714 042777 000100 171450      BIC      #BIT6,@DQSEC ;SEL POLY 0-15
2040 007722 012705 000015      MOV      #15,R5       ;SET FOR ERROR
2041 007726 110577 171436      MOVB     R5,@DQREG    ;SEL BCC REG
2042 007732 017701 171434      MOV      @DQSEC,R1    ;READ BCC
2043 007736 005701              TST      R1           ;IS IT 0
2044 007740 001401              BEQ      .+4           ;BR IF YES.
2045 007742 104002              HLT      2            ;BCC NOT 0
2046 007744 012705 000016      MOV      #16,R5       ;
2047 007750 013701 015364      MOV      SEC16X,R1    ;
2048 007754 005701              TST      R1           ;
2049 007756 001401              BEQ      .+4           ;
2050 007760 104002              HLT      2            ;
2051 007762 010437 015344      MOV      R4,CALBCC    ;
2052 007766 112777 000012 171374      MOVB     #MISC.,@DQREG ;
2053 007774 005277 171372      INC      @DQSEC       ;
2054 010000 005377 171366      DEC      @DQSEC       ;
2055 010004 012705 000016      MOV      #16,R5       ;
2056 010010 110577 171354      MOVB     R5,@DQREG    ;
2057 010014 017701 171352      MOV      @DQSEC,R1    ;
2058 010020 000301              SWAB     R1           ;
2059 010022 023701 015344      CMP      CALBCC,R1    ;
2060 010026 001401              BEQ      .+4           ;
2061 010030 104002              HLT      2            ;
2062 010032 012737 000007 015352      MOV      #7,COUNT     ;
2063 010040 112777 000012 171322      MOVB     #MISC.,@DQREG ;
2064 010046 005277 171320      INC      @DQSEC       ;
2065 010052 005377 171314      DEC      @DQSEC       ;
2066 010056 005337 015352      DEC      COUNT       ;
2067 010062 001371              BNE     3$           ;
2068 010064 012705 000015      MOV      #15,R5       ;
2069 010070 110577 171274      MOVB     R5,@DQREG    ;
2070 010074 017701 171272      MOV      @DQSEC,R1    ;
2071 010100 000301              SWAB     R1           ;
2072 010102 023701 015344      CMP      CALBCC,R1    ;
2073 010106 001401              BEQ      .+4           ;
2074 010110 104002              HLT      2            ;
2075 010112 104401              SCOP1    ;
2076 010114 105237 015344      INCB     CALBCC       ;
2077 010120 001235              BNE     2$           ;
2078 010122 104413              MEMCLR   ;
2079 010124 104413              MEMCLR   ;
2080 010126 104400              SCOPE    ;
2081 :
2082 ;USERS BCC RECEIVER TRANSMITTER TEST
2083 ;THIS TEST ALLOWS THE USER TO

```


MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2110 010202 011637 015360      STBCC: MOV      (SP),SAVEPC      ;SAVE PC OF ENTERING ROUTINE
2111 010206 104413              CLR      MEMCLR
2112 010210 005000              CLR      R0
2113 010212 012704 015376      1$: MOV      #TXBUFF,R4
2114 010216 110024              MOV      R0,(R4)+
2115 010220 105200              INCB     R0
2116 010222 001375              BNE      1$
2117 010224 105077 171140      CLR      @DQREG      ;SELECT THE RX BA PRI.
2118 010230 012777 016000 171134 MOV      #RXBUFF,@DQSEC ;LOAD THE RX BA
2119 010236 112777 000121 171124 MOV      #121,@DQREG    ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2120 010244 012777 177400 171120 MOV      #-400,@DQSEC  ;SET RX CC FOR A TRANSFER OF 400 CHARS.
2121 010252 112777 000022 171110 MOV      #22,@DQREG    ;WRITE ENABLE, TX BA PRI.
2122 010260 012777 015374 171104 MOV      #SYNC,@DQSEC  ;LOAD THE TX BA PRI.
2123 010266 112777 000023 171074 MOV      #23,@DQREG    ;ENTER T,WRITE ENABLE, TX CC PRI.
2124 010274 012777 177776 171070 MOV      #-2,@DQSEC    ;SET TX CC FOR A TRANSFER OF 2 CHARS.
2125 010302 112777 000024 171060 MOV      #24,@DQREG    ;WRITE ENABLE,RX BA SEC.
2126 010310 005077 171056      CLR      @DQSEC      ;CLEAR THE RX BA SEC
2127 010314 112777 000065 171046 MOV      #65,@DQREG    ;EXIT T,WRITE ENABLE,RX CC SEC.
2128 010322 012777 177777 171042 MOV      #-1,@DQSEC    ;SET THE RX CC SEC FOR ONE PAD CHAR.
2129 010330 112777 000026 171032 MOV      #26,@DQREG    ;WRITE ENABLE, TX BA SEC.
2130 010336 012777 015376 171026 MOV      #TXBUFF,@DQSEC
2131 010344 112777 000127 171016 MOV      #127,@DQREG   ;EXIT T,WRITE ENABLE, TX CC SEC.
2132 010352 012777 177400 171012 MOV      #-400,@DQSEC
2133 010360 112777 000011 171002 MOV      #11,@DQREG
2134 010366 013777 015372 170776 MOV      .SYNC,@DQSEC
2135 010374 112777 000017 170766 MOV      #17,@DQREG
2136 010402 013777 015350 170762 MOV      XPOLY,@DQSEC
2137 010410 112777 000012 170752 MOV      #MISC.,@DQREG
2138 010416 012777 004010 170746 MOV      #4010,@DQSEC
2139 010424 005037 001244      CLR      TEMP1
2140 010430 005037 001246      CLR      TEMP2
2141 010434 012777 010504 170712 MOV      #TXISR1,@DQTEC
2142 010442 005077 170710      CLR      @DQTLVL
2143 010446 012777 000001 170704 MOV      #1,@DQRCR
2144 010454 012777 000041 170702 MOV      #41,@DQTCR
2145 010462 005037 177776      CLR      PS
2146 010466 105237 001244      2$: INCB     TEMP1
2147 010472 001375              BNE      2$
2148 010474 105237 001246      INCB     TEMP2
2149 010500 001372              BNE      2$
2150 010502 104000              HLT
2151 010504 022626              4.7 TXISR1: CMP      (SP)+,(SP)+
2152 010506 042777 000040 170650 8.2      BIC      #BIT5,@DQTCR
2153 010514 112777 000063 170646 7.6      MOV      #63,@DQREG
2154 010522 005077 170644 6.1      CLR      @DQSEC
2155 010526 000240 1.5      NOP
2156 010530 005037 001244 3.7      CLR      TEMP1
2157 010534 005037 001246 3.7      CLR      TEMP2
2158 010540 032777 000100 170612 7.7 1$: BIT      #BIT6,@DQRCR
2159 010546 001007 2.6      BNE      ENDBCC
2160 010550 105237 001244 3.7      INCB     TEMP1
2161 010554 001371 2.6      BNE      1$
2162 010556 105237 001246 3.7      INCB     TEMP2
2163 010562 001366 2.6      BNE      1$
2164 010564 104000 9.3      HLT
2165 010566 000207 3.5 ENDBCC: RTS      PC      ;RX SECONDARY DONE NOT SET.
  
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2166
2167
2168
2169 010570 010537 015360      SYNBC: MOV      R5,SAVEPC      ;SAVE PC OF ENTERING ROUTINE
2170 010574 104412              MSTCLR           ;CLEAR THE DQ11
2171 010576 105077 170566      CLR            @DQREG        ;SELECT THE RX BA PRI.
2172 010602 012777 016000 170562  MOV          #RXBUFF,@DQSEC  ;LOAD THE RX BA
2173 010610 112777 000121 170552  MOVB        #121,@DQREG      ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2174 010616 012577 170550      MOV          (R5)+,@DQSEC    ;SET RX CC
2175 010622 112777 000022 170540  MOVB        #22,@DQREG      ;WRITE ENABLE, TX BA PRI.
2176 010630 012777 015376 170534  MOV          #TXBUFF,@DQSEC  ;LOAD THE TX BA PRI.
2177 010636 112777 000123 170524  MOVB        #123,@DQREG      ;ENTER T,WRITE ENABLE, TX CC PRI.
2178 010644 012577 170522      MOV          (R5)+,@DQSEC    ;SET TX CC
2179 010650 112777 000024 170512  MOVB        #24,@DQREG      ;WRITE ENABLE,RX BA SEC.
2180 010656 012777 016000 170506  MOV          #RXBUFF,@DQSEC  ;LOAD THE RX BA SEC.
2181 010664 112777 000065 170476  MOVB        #65,@DQREG      ;EXIT T,WRITE ENABLE,RX CC SEC.
2182 010672 012777 177777 170472  MOV          #-1,@DQSEC      ;SET FOR ONE PAD CHARACTER.
2183 010700 112777 000026 170462  MOVB        #26,@DQREG      ;WRITE ENABLE, TX BA SEC.
2184 010706 005077 170460      CLR            @DQSEC        ;CLEAR THE TX CC SEC
2185 010712 112777 000067 170450  MOVB        #67,@DQREG      ;EXIT T,WRITE ENABLE, TX CC SEC.
2186 010720 005077 170446      CLR            @DQSEC        ;CLEAR THE TX CC SEC
2187 010724 112777 000011 170436  MOVB        #11,@DQREG
2188 010732 013777 015372 170432  MOV          .SYNC,@DQSEC
2189 010740 112777 000017 170422  MOVB        #17,@DQREG
2190 010746 013777 015350 170416  MOV          XPOLY,@DQSEC
2191 010754 112777 000012 170406  MOVB        #MISC.,@DQREG
2192 010762 012777 004010 170402  MOV          #4010,@DQSEC
2193 010770 005037 001244      CLR            TEMP1
2194 010774 005037 001246      CLR            TEMP2
2195 011000 052777 000003 170352  BIS         #3,@DQRCSR
2196 011006 052777 000002 170350  BIS         #2,@DQTCSR
2197 011014 032777 020000 170350  2$: BIT      #BIT13,@DQSEC
2198 011022 001006              BNE          3$
2199 011024 005237 001244      INC          TEMP1
2200 011030 001371              BNE          2$
2201 011032 005237 001246      INC          TEMP2
2202 011036 001366              BNE          2$
2203 011040 012777 000001 170316  3$: MOV      #BIT0,@DQTCSR
2204 011046 005037 001244      CLR          TEMP1
2205 011052 012737 000005 001246  MOV      #5,TEMP2
2206 011060 032777 000100 170272  4$: BIT      #BIT6,@DQRCSR
2207 011066 001007              BNE          ENDSYN
2208 011070 005237 001244      INC          TEMP1
2209 011074 001371              BNE          4$
2210 011076 005337 001246      DEC          TEMP2
2211 011102 001366              BNE          4$
2212 011104 104000              HLT
2213 011106 000205      ENDSYN: RTS      R5      ;RECEIVER DONE SECONDARY NOT SET.
2214
2215
2216
2217 011110 011637 015360      TYBCC: MOV      (SP),SAVEPC   ;SAVE PC OF ENTERING ROUTINE
2218 011114 104412              MSTCLR           ;CLEAR THE DQ11
2219 011116 000240              NOP
2220 011120 104412              MSTCLR
2221 011122 012737 000351 015376  MOV      #351,TXBUFFER
  
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2222 011130 012737 000011 015352      MOV      #9.,COUNT
2223 011136 105077 170226      1$: CLR B  @DQREG      ;SELECT THE RX BA PRI.
2224 011142 012777 016000 170222      MOV      #RXBUFF,@DQSEC ;LOAD THE RX BA
2225 011150 112777 000121 170212      MOV B   #121,@DQREG     ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2226 011156 012777 177600 170206      MOV      #-200,@DQSEC   ;SET RX CC FOR A TRANSFER OF 1 CHARACTER.
2227 011164 112777 000022 170176      MOV B   #22,@DQREG     ;WRITE ENABLE, TX BA PRI.
2228 011172 012777 015376 170172      MOV      #TXBUFF,@DQSEC ;LOAD THE TX BA PRI.
2229 011200 112777 000123 170162      MOV B   #123,@DQREG    ;ENTER T,WRITE ENABLE, TX CC PRI.
2230 011206 012777 177600 170156      MOV      #-200,@DQSEC   ;SET TX CC FOR A TRANSFER OF 1 CHARACTER.
2231 011214 112777 000024 170146      MOV B   #24,@DQREG     ;WRITE ENABLE,RX BA SEC.
2232 011222 005077 170144      CLR      @DQSEC        ;CLEAR THE RX BA SEC
2233 011226 112777 000065 170134      MOV B   #65,@DQREG     ;EXIT T,WRITE ENABLE,RX CC SEC.
2234 011234 005077 170132      CLR      @DQSEC        ;CLEAR THE RX CC SEC.
2235 011240 112777 000026 170122      MOV B   #26,@DQREG     ;WRITE ENABLE, TX BA SEC.
2236 011246 005077 170120      CLR      @DQSEC        ;CLEAR THE TX CC SEC
2237 011252 112777 000067 170110      MOV B   #67,@DQREG     ;EXIT T,WRITE ENABLE, TX CC SEC.
2238 011260 005077 170106      CLR      @DQSEC        ;CLEAR THE TX CC SEC
2239 011264 112777 000017 170076      MOV B   #17,@DQREG
2240 011272 013777 015350 170072      MOV      XPOLY,@DQSEC
2241 011300 112777 000012 170062      MOV B   #MISC.,@DQREG
2242 011306 012777 004012 170056      MOV      #4012,@DQSEC
2243 011314 052777 000001 170042      BIS      #BIT0,@DQTCR  ;SET TRANSMITTER GO
2244 011322 027777 170036 170034      CMP      @DQTCR,@DQTCR ;WAIST TIME.
2245 011330 027777 170030 170026      CMP      @DQTCR,@DQTCR ;WAIST TIME
2246 011336 027777 170022 170020      CMP      @DQTCR,@DQTCR ;WAIST TIME
2247 011344 005277 170022      INC      @DQSEC        ;PRIM THE
2248 011350 005377 170016      DEC      @DQSEC        ;
2249 011354 042777 000200 170010      BIC      #BIT7,@DQSEC  ;CLEAR THE BIT WINDOW.
2250 011362 052777 010001 167770      BIS      #10001,@DQRCR
2251 011370 005277 167776      2$: INC      @DQSEC
2252 011374 005377 167772      DEC      @DQSEC
2253 011400 005337 015352      DEC      COUNT
2254 011404 001371      BNE      2$
2255 011406 112777 000016 167754      MOV B   #16,@DQREG
2256 011414 017737 167752 015362      MOV      @DQSEC,SEC16
2257 011422 112777 000012 167740      MOV B   #MISC.,@DQREG
2258 011430 012737 000007 015352      MOV      #7.,COUNT
2259 011436 005277 167730      3$: INC      @DQSEC
2260 011442 005377 167724      DEC      @DQSEC
2261 011446 005337 015352      DEC      COUNT
2262 011452 001371      BNE      3$
2263 011454 000207      ENDTY: RTS      PC
2264
2265
2266
2267
2268
2269 011456 011637 015360      TXBCC: MOV      (SP),SAVEPC ;SAVE PC OF ENTERING ROUTINE
2270 011462 104412      MSTCLR ;CLEAR THE DQ11
2271 011464 013737 015344 015376      MOV      CALBCC,TXBUFF
2272 011472 012737 000011 015352      MOV      #9.,COUNT
2273 011500 105077 167664      1$: CLR B  @DQREG      ;SELECT THE RX BA PRI.
2274 011504 012777 016000 167660      MOV      #RXBUFF,@DQSEC ;LOAD THE RX BA
2275 011512 112777 000121 167650      MOV B   #121,@DQREG    ;*ENTER T,WRITE ENABLE,RX CC,PRI.
2276 011520 012777 177777 167644      MOV      #-1,@DQSEC   ;SET RX CC FOR A TRANSFER OF 1 CHARACTER.
2277 011526 112777 000022 167634      MOV B   #22,@DQREG     ;WRITE ENABLE, TX BA PRI.

```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2278 011534 012777 015376 167630      MOV      #TXBUFF,@DQSEC      ;LOAD THE TX BA PRI.
2279 011542 112777 000123 167620      MOV      #123,@DQREG        ;ENTER T,WRITE ENABLE,TX CC PRI.
2280 011550 012777 177777 167614      MOV      #-1,@DQSEC         ;SET TX CC FOR A TRANSFER OF 1 CHARACTER.
2281 011556 112777 000024 167604      MOV      #24,@DQREG         ;WRITE ENABLE,RX BA SEC.
2282 011564 005077 167602      CLR      @DQSEC             ;CLEAR THE RX BA SEC
2283 011570 112777 000065 167572      MOV      #65,@DQREG         ;EXIT T,WRITE ENABLE,RX CC SEC.
2284 011576 005077 167570      CLR      @DQSEC             ;CLEAR THE RX CC SEC.
2285 011602 112777 000026 167560      MOV      #26,@DQREG         ;WRITE ENABLE,TX BA SEC.
2286 011610 005077 167556      CLR      @DQSEC             ;CLEAR THE TX CC SEC
2287 011614 112777 000067 167546      MOV      #67,@DQREG         ;EXIT T,WRITE ENABLE,TX CC SEC.
2288 011622 005077 167544      CLR      @DQSEC             ;CLEAR THE TX CC SEC
2289 011626 112777 000012 167534      MOV      #MISC.,@DQREG      ;
2290 011634 012777 004112 167530      MOV      #4112,@DQSEC      ;
2291 011642 112777 000017 167520      MOV      #17,@DQREG        ;
2292 011650 013777 015350 167514      MOV      XPOLY,@DQSEC      ;
2293 011656 112777 000012 167504      MOV      #MISC.,@DQREG      ;
2294 011664 052777 000001 167472      BIS      #BIT0,@DQTCR      ;SET TRANSMITTER GO
2295 011672 027777 167456 167464      CMP      @DQTCR,@DQTCR     ;WAIT TIME.
2296 011700 027777 167460 167456      CMP      @DQTCR,@DQTCR     ;WAIT TIME
2297 011706 027777 167452 167450      CMP      @DQTCR,@DQTCR     ;WAIT TIME
2298 011714 005277 167452      INC      @DQSEC             ;PRIM THE
2299 011720 005377 167446      DEC      @DQSEC             ; TRANSMITTER.
2300 011724 042777 000200 167440      BIC      #BIT7,@DQSEC      ;CLEAR THE BIT WINDOW.
2301 011732 052777 010001 167420      BIS      #10001,@DQRCR
2302 011740 005277 167426      2$: INC      @DQSEC
2303 011744 005377 167422      DEC      @DQSEC
2304 011750 005337 015352      DEC      COUNT
2305 011754 001371      BNE      2$
2306 011756 042777 000100 167406      BIC      #BIT6,@DQSEC
2307 011764 112777 000016 167376      MOV      #16,@DQREG
2308 011772 017737 167374 015364      MOV      @DQSEC,SEC16X
2309 012000 112777 000012 167362      MOV      #MISC.,@DQREG
2310 012006 052777 000100 167356      BIS      #BIT6,@DQSEC
2311 012014 112777 000016 167346      MOV      #16,@DQREG
2312 012022 017737 167344 015362      MOV      @DQSEC,SEC16
2313 012030 112777 000012 167332      MOV      #MISC.,@DQREG
2314 012036 012737 000007 015352      MOV      #7,COUNT
2315 012044 005277 167322      3$: INC      @DQSEC
2316 012050 005377 167316      DEC      @DQSEC
2317 012054 005337 015352      DEC      COUNT
2318 012060 001371      BNE      3$
2319 012062 000207      ENDTX: RTS      PC

```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2320
2321          ;BLOCK CHECK CHARACTER GENERATOR SIMULATOR.
2322          ;
2323 012064 012537 001244 SIMBCC: MOV (R5)+,TEMP1
2324 012070 012537 001246          MOV (R5)+,TEMP2
2325 012074 012537 001250          MOV (R5)+,TEMP3
2326 012100 005037 015346 1$: CLR BCCFBK
2327 012104 013700 001250          MOV TEMP3,R0
2328 012110 006037 001246          ROR TEMP2
2329 012114 005500          ADC R0
2330 012116 032700 000001          BIT #BIT0,R0
2331 012122 001402          BEQ 2$
2332 012124 005137 015346          COM BCCFBK
2333 012130 013700 015350 2$: MOV XPOLY,R0
2334 012134 005100          COM R0
2335 012136 040037 015346          BIC R0,BCCFBK
2336 012142 000241          CLC
2337 012144 006037 001250          ROR TEMP3
2338 012150 013700 015346          MOV BCCFBK,R0
2339 012154 013701 001250          MOV TEMP3,R1
2340 012160 010102          MOV R1,R2
2341 012162 040100          BIC R1,R0
2342 012164 043702 015346          BIC BCCFBK,R2
2343 012170 050200          BIS R2,R0
2344 012172 043737 015350 001250          BIC XPOLY,TEMP3
2345 012200 050037 001250          BIS R0,TEMP3
2346 012204 005337 001244          DEC TEMP1
2347 012210 001333          BNE 1$
2348 012212 013737 001250 015344          MOV TEMP3,CALBCC
2349 012220 000205          RTS R5
2350
2351          .MEMCLR:
2352 012222 005077 167132          CLR @DQRC SR
2353 012226 005077 167132          CLR @DQTC SR
2354 012232 005077 167130          CLR @DQERR
2355 012236 012705 000020          MOV #16.,R5
2356 012242 152777 000020 167120 1$: BISS #BIT4,@DQREG
2357 012250 142777 000140 167112          BICB #140,@DQREG
2358 012256 005077 167110          CLR @DQSEC
2359 012262 105277 167102          INCB @DQREG
2360 012266 005305          DEC R5
2361 012270 001364          BNE 1$
2362 012272 105077 167072          CLRB @DQREG
2363 012276 105077 167060          CLRB @DQRC SH
2364 012302 012705 000020          MOV #16.,R5
2365 012306 112777 000010 167054 2$: MOVB #10,@DQREG
2366 012314 005077 167052          CLR @DQSEC
2367 012320 112777 000014 167042          MOVB #14,@DQREG
2368 012326 005077 167040          CLR @DQSEC
2369 012332 105277 167024          INCB @DQRC SH
2370 012336 005305          DEC R5
2371 012340 001362          BNE 2$
2372 012342 105077 167014          CLRB @DQRC SH
2373 012346          .MSTCLR:
2374 012346 112777 000012 167014          MOVB #!!SC,@DQREG
2375 012354 012777 000040 167010          MOV #BIT5,@DQSEC
  
```

MISC. RECEIVER AND TRANSMITTER TESTS PLUS BCC TESTS.

```

2376 012362 000002          RTI
2377
2378
2379
2380
2381          :END OF PASS
2382          :TYPE NAME OF TEST
2383          :UPDATE PASS COUNT
2384          :CHECK FOR EXIT TO ACT-11
2385          :RESTART TEST
2386
2387 012364 005037 001234      .EOP: CLR      LSTERR      :CLEAR LAST ERROR PC
2388 012370 005037 001312      CLR      ERRFLG      :CLEAR ERROR FLAG
2389 012374 005237 001230      INC      PASCNT      :UPDATE PASS COUNT
2390 012400 104402          TYPE
2391 012402 014614          MEPASS
2392 012404 104402          TYPE
2393 012406 014775          MCSRX
2394 012410 104411          CNVRT
2395 012412 012522          XCSR
2396 012414 104402          TYPE
2397 012416 015003          MVECX
2398 012420 104411          CNVRT
2399 012422 012530          XVEC
2400 012424 104402          TYPE
2401 012426 015011          MPASSX
2402 012430 104411          CNVRT
2403 012432 012536          XPASS
2404 012434 104402          TYPE
2405 012436 015022          MERRX
2406 012440 104411          CNVRT
2407 012442 012544          XERR
2408 012444 013777 001230 166530  MOV      PASCNT,@LIGHTS :DISPLAY PASS COUNT
2409 012452 005337 001276      DEC      SAVNUM
2410 012456 001013          BNE      RESTRT
2411 012460 013737 001504 001276  MOV      DQNUM,SAVNUM
2412 012466 013701 000042      MOV      @#42,R1
2413 012472 001405          BEQ      RESTRT      :CHECK FOR ACT-11 OR DDP
2414 012474 000005          RESET      :IF NOT, CONTINUE TESTING
2415 012476          LOGICAL:
2416 012476 004711          JSR      PC,(R1)
2417 012500 000240          NOP
2418 012502 000240          NOP
2419 012504 000240          NOP
2420 012506 104414          RESTRT: CKSWR
2421 012510 012737 002254 001214  MOV      #TST1,RETURN
2422 012516 000137 002254          JMP      TST1
2423 012522 000001          XCSR:   1
2424 012524          .BYTE  6,2
2425 012526 001360          DQRCSR
2426 012530 000001          XVEC:   1
2427 012532 003          .BYTE  3,2
2428 012534 001350          DORVEC
2429 012536 000001          XPASS:  1
2430 012540          .BYTE  6,2
2431 012542 001230          PASCNT

```

```

2432 012544 000001          XERR: 1
2433 012546      006      002      .BYTE 6,2
2434 012550 001232          ERRCNT
2435
2436                          ;SCOPE LOOP AND INTERATION HANDLER
2437
2438 012552 104414          .SCOPE: CKSWR
2439 012554 032777 040000 166416  BIT      #BIT14,@SWR
2440 012562 001407          TTST:  BEQ      1$
2441 012564 000432          BR      3$
2442 012566 105777 166412  TSTB    @TKCSR
2443 012572 100027          BPL     3$
2444 012574 017700 166406  MOV     @TKDBR,R0
2445 012600 000412          BR      2$
2446 012602 032777 004000 166370 1$:  BIT     #SW11,@SWR
2447 012610 001006          BNE     2$
2448 012612 005237 001224  INC     LPCNT
2449 012616 023737 001224 001222  CMP     LPCNT,I COUNT
2450 012624 001012          BNE     3$
2451 012626 105037 001312  2$:  CLRB   ERRFLG
2452 012632 005037 001224  CLR     LPCNT
2453 012636 012737 000011 001222  MOV     #9,I COUNT
2454 012644 013737 001216 001214  MOV     NEXT,RETURN
2455 012652 013716 001214  3$:  MOV     RETURN,(SP)
2456 012656 000002          RTI
2457 012660 001407          BRW:  1407
2458 012662 000432          BRX:  432
2459
2460                          ;CHECK FOR FREEZE ON CURRENT DATA
2461
2462 012664 104414          .SCOPE1: CKSWR
2463 012666 032777 001000 166304  BIT     #SW09,@SWR
2464 012674 001402          BEQ     1$
2465 012676 013716 001220  MOV     LOCK,(SP)
2466 012702 000002          1$:  RTI
2467
2468                          ;TELETYPE OUTPUT ROUTINE
2469
2470 012704 010546          .TYPE:  MOV     R5,-(SP)
2471 012706 017605 000002  MOV     @2(SP),R5
2472 012712 062766 000002 000002  ADD     #2,2(SP)
2473 012720 005737 014374  1$:  TST     @#RDSW
2474 012724 001004          BNE     300$
2475 012726 032777 010000 166244  BIT     #SW12,@SWR
2476 012734 001024          BNE     3$
2477 012736 105715          300$: TSTB   (R5)
2478 012740 100014          BPL     2$
2479 012742 105777 166242  TSTB   @TPCSR
2480 012746 100375          BPL     #-4
2481 012750 012777 000015 166234  MOV     #15,@TPDBR
2482 012756 105777 166226  TSTB   @TPCSR
2483 012762 100375          BPL     #-4
2484 012764 012777 000012 166220  MOV     #12,@TPDBR
2485 012772 105777 166212  2$:  TSTB   @TPCSR
2486 012776 100375          BPL     2$
2487 013000 112577 166206  MOVB   (R5)+,@TPDBR
  
```

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

```

2488 013004 001345
2489 013006 012605
2490 013010 000002
2491
2492
2493
2494 013012 010346
2495 013014 010446
2496 013016 017637 000004 013034
2497 013024 062766 000002 000004
2498 013032 104402
2499 013034 000000
2500 013036 012704 015166
2501 013042 012703 000007
2502 013046 105777 166132
2503 013052 100375
2504 013054 117714 166126
2505 013060 142714 000200
2506 013064 121427 000025
2507 013070 001003
2508 013072 104402 014554
2509 013076 000755
2510 013100 122427 000015
2511 013104 001423
2512 013106 117777 166074 166076
2513 013114 105777 166070
2514 013120 100375
2515 013122 005303
2516 013124 001350
2517 013126 000402
2518 013130 010346
2519 013132 010446
2520 013134 104402
2521 013136 014550
2522 013140 005737 014374
2523 013144 001402
2524 013146 104402 014554
2525 013152 000727
2526 013154 012604
2527 013156 012603
2528 013160 000002
2529
2530
2531
2532 013162 010546
2533 013164 010446
2534 013166 016605 000004
2535 013172 012537 013366
2536 013176 012537 013370
2537 013202 012537 013372
2538 013206 112537 013374
2539 013212 112537 013375
2540 013216 010566 000004
2541 013222 005005
2542 013224 012704 015166
2543 013230 122714 000015

3$: BNE 1$
MOV (SP)+,R5
RTI

;ASCII STRING INPUT ROUTINE

.INSTR: MOV R3,-(SP)
MOV R4,-(SP)
MOV @4(SP),MSG
ADD #2,4(SP)

.INST1: TYPE
.MSG: 0
MOV #INBUF,R4
MOV #7,R3
1$: TSTB @TKCSR
BPL 1$
MOVB @TKDBR,(R4)
BICB #200,(R4)
CMPB (R4),#25 ;IS IT <^G>
BNE 200$
TYPE,MCRLF
BR .INST1
200$: CMPB (R4)+,#15
BEQ INSTR2
MOVB @TKDBR,@TPDBR
2$: TSTB @TPCSR
BPL 2$
DEC R3
BNE 1$
BR .INSTG
.INSTE: MOV R3,-(SP)
MOV R4,-(SP)
.INSTG: TYPE
MQM
TST @#RDSW
BEQ 400$
TYPE,MCRLF
400$: BR .INST1
INSTR2: MOV (SP)+,R4
MOV (SP)+,R3
RTI

;CONVERT ASCII STRING TO OCTAL

.PARAM: MOV R5,-(SP)
MOV R4,-(SP)
MOV 4(SP),R5
MOV (R5)+,LOLIM
MOV (R5)+,HILIM
MOV (R5)+,DEVADR
MOVB (R5)+,LOBITS
MOVB (R5)+,ADRCNT
MOV R5,4(SP)
PARAM1: CLR R5
MOV #INBUF,R4
CMPB #15,(R4)

```

```

2544 013234 001420          BEQ      PARERR
2545 013236 121427 000060  1$:     CMPB   (R4),#60
2546 013242 002415          BLT     PARERR
2547 013244 121427 000067           CMPB   (R4),#67
2548 013250 003012          BGT     PARERR
2549 013252 142714 000060           BICB   #60,(R4)
2550 013256 152405          BISB   (R4)+,R5
2551 013260 122714 000015           CMPB   #15,(R4)
2552 013264 001414          BEQ     LIMITS
2553 013266 006305          ASL    R5
2554 013270 006305          ASL    R5
2555 013272 006305          ASL    R5
2556 013274 000760          BR     1$
2557 013276 122714 000015  PARERR: CMPB   #15,(R4)          ;IS FIRST CHARACTER A <CR>
2558 013302 001003          BNE    120$
2559 013304 005737 014374  TST    @#RDSW          ;IS CKSWR ROUTINE BEING USED
2560 013310 001023          BNE    PARTI
2561 013312 104404          120$:  INSTER
2562 013314 000742          BR     PARAM1
2563
2564          ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
2565
2566 013316 020537 013370  LIMITS: CMP    R5,HILIM
2567 013322 101365          BHI    PARERR
2568 013324 020537 013366           CMP    R5,LOLIM
2569 013330 103762          BLO    PARERR
2570 013332 133705 013374           BITB  LOBITS,R5
2571 013336 001357          BNE    PARERR
2572
2573          ;STORE NUMBER AT SPECIFIED ADDRESS
2574
2575 013340 013704 013372  1$:     MOV    DEVADR,R4
2576 013344 010524          MOV    R5,(R4)+
2577 013346 062705 000002           ADD    #2,R5
2578 013352 105337 013375           DECB  ADRCNT
2579 013356 001372          BNE    1$
2580 013360 012604          PARTI: MOV    (SP)+,R4
2581 013362 012605          MOV    (SP)+,R5
2582 013364 000002          RTI
2583 013366 000000          LOLIM: 0
2584 013370 000000          HILIM: 0
2585 013372 000000          DEVADR: 0
2586 013374 000000          LOBITS: 0
2587          ADRCNT=LOBITS+1
2588
2589          ;SAVE PC OF TEST THAT FAILED AND R0-R5
2590
2591 013376 016637 000004 001274 .SAV05: MOV    4(SP),SAVPC
2592
2593          ;SAVE R0-R5
2594
2595 013404 010537 001270  SV05:  MOV    R5,SAVR5
2596 013410 010437 001266          MOV    R4,SAVR4
2597 013414 010337 001264          MOV    R3,SAVR3
2598 013420 010237 001262          MOV    R2,SAVR2
2599 013424 010137 001260          MOV    R1,SAVR1

```

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

2600	013430	010037	001256		MOV	R0,SAVR0
2601	013434	000002			RTI	
2602						
2603					;RESTORE R0-R5	
2604						
2605	013436	013700	001256	.RES05:	MOV	SAVR0,R0
2606	013442	013701	001260		MOV	SAVR1,R1
2607	013446	013702	001262		MOV	SAVR2,R2
2608	013452	013703	001264		MOV	SAVR3,R3
2609	013456	013704	001266		MOV	SAVR4,R4
2610	013462	013705	001270		MOV	SAVR5,R5
2611	013466	000002			RTI	
2612						
2613					;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER	
2614						
2615	013470	104402		.CONVR:	TYPE	
2616	013472	014554			MCRLF	
2617	013474	010046		.CNVRT:	MOV	R0,-(SP)
2618	013476	010146			MOV	R1,-(SP)
2619	013500	010346			MOV	R3,-(SP)
2620	013502	010446			MOV	R4,-(SP)
2621	013504	010546			MOV	R5,-(SP)
2622	013506	017601	000012		MOV	@12(SP),R1
2623	013512	013737	015230		MOV	TEMP,TEMP3
2624	013520	062766	000002	001250	ADD	#2,12(SP)
2625	013526	012137	013710	000012	MOV	(R1)+,WRDCNT
2626	013532	112137	013712		MOVB	(R1)+,CHRCNT
2627	013536	112137	013713	1\$:	MOVB	(R1)+,SPACNT
2628	013542	013137	013714		MOV	@(R1)+,BINWRD
2629	013546	013704	013714	2\$:	MOV	BINWRD,R4
2630	013552	113705	013712		MOVB	CHRCNT,R5
2631	013556	012700	015230		MOV	#TEMP,R0
2632	013562	010403		3\$:	MOV	R4,R3
2633	013564	042703	177770		BIC	#177770,R3
2634	013570	062703	000060		ADD	#060,R3
2635	013574	110320			MOVB	R3,(R0)+
2636	013576	000241			CLC	
2637	013600	006004			ROR	R4
2638	013602	000241			CLC	
2639	013604	006004			ROR	R4
2640	013606	000241			CLC	
2641	013610	006004			ROR	R4
2642	013612	005305			DEC	R5
2643	013614	001362			BNE	3\$
2644	013616	012703	015272		MOV	#MDATA,R3
2645	013622	114023		4\$:	MOVB	-(R0),(R3)+
2646	013624	105337	013712		DECB	CHRCNT
2647	013630	001374			BNE	4\$
2648	013632	105737	013713		TSTB	SPACNT
2649	013636	001405			BEQ	6\$
2650	013640	112723	000040	5\$:	MOVB	#040,(R3)+
2651	013644	105337	013713		DECB	SPACNT
2652	013650	001373			BNE	5\$
2653	013652	105013		6\$:	CLRB	(R3)
2654	013654	104402			TYPE	
2655	013656	015272			MDATA	

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

2656	013660	005337	013710		DEC	WRDCNT	
2657	013664	001322			BNE	1\$	
2658	013666	013737	001250	015230	MOV	TEMP3,TEMP	
2659	013674	012605			MOV	(SP)+,R5	
2660	013676	012604			MOV	(SP)+,R4	
2661	013700	012603			MOV	(SP)+,R3	
2662	013702	012601			MOV	(SP)+,R1	
2663	013704	012600			MOV	(SP)+,R0	
2664	013706	000002			RTI		
2665	013710	000000			WRDCNT:	0	
2666	013712	000000			CHRCNT:	0	
2667		013713			SPACNT=	CHRCNT+1	
2668	013714	000000			BINWRD:	0	
2669						:TRAP DISPATCH SERVICE	
2670						:ARGUMENT OF TRAP IS EXTRACTED	
2671						:AND USED AS OFFSET TO OBTAIN POINTER	
2672						:TO SELECTED SUBROUTINE	
2673							
2674	013716	011646			.TRPSR:	MOV (SP),-(SP)	:GET PC OF RETURN
2675	013720	162716	000002		SUB	#2,(SP)	:PC OF TRAP
2676	013724	017616	000000		MOV	@(SP),(SP)	:GET TRP
2677	013730	006316			TRPOK:	ASL (SP)	:MULTIPLY TRAP ARG BY 2
2678	013732	042716	177001		BIC	#177001,(SP)	:CLEAR UNWANTED BITS
2679	013736	062716	001314		ADD	#.TRPTAB,(SP)	:POINTER TO SUBROUTINE ADDRESS
2680	013742	017616	000000		MOV	@(SP),(SP)	:SUBROUTINE ADDRESS
2681	013746	000136			JMP	@(SP)+	:GO TO SUBROUTINE
2682							
2683						:ERROR HANDLER	
2684							
2685	013750	104414			.HLT:	CKSWR	
2686	013752	032777	010000	165220	BIT	#SW12,@SWR	
2687	013760	001406			BEQ	XBX	
2688	013762	105777	165222		TSTB	@TPCSR	
2689	013766	100003			BPL	XBX	
2690	013770	112777	000207	165214	MOVB	#207,@TPDBR	
2691	013776	032777	020000	165174	XBX:	BIT #SW13,@SWR	
2692	014004	001074			BNE	HALTS	
2693	014006	021637	001234		CMP	(SP),LSTERR	
2694	014012	001404			BEQ	1\$	
2695	014014	011637	001234		MOV	(SP),LSTERR	
2696	014020	105037	001312		CLRB	ERRFLG	
2697	014024	104406			1\$:	SAV05	
2698	014026	011605			MOV	(SP),R5	
2699	014030	162705	000002		SUB	#2,R5	
2700	014034	011504			MOV	(R5),R4	
2701	014036	006304			ASL	R4	
2702	014040	061504			ADD	(R5),R4	
2703	014042	006304			ASL	R4	
2704	014044	042704	177001		BIC	#177001,R4	
2705	014050	062704	016402		ADD	#.ERRTAB,R4	
2706	014054	012437	014146		MOV	(R4)+,ERRMSG	
2707	014060	012437	014160		MOV	(R4)+,DATAHD	
2708	014064	011437	014172		MOV	(R4),DATABP	
2709	014070	105737	001312		TSTB	ERRFLG	
2710	014074	001403			BEQ	T:PMMSG	
2711	014076	005737	014172		TST	DATABP	

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

```
2768 ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
2769
2770 014312 RESTAR:
2771 014312 012737 014300 000024 MOV #.PFAIL,24 ;SET UP FOR POWER FAILURE
2772 014320 012706 001200 MOV #STACK,SP
2773 014324 005037 015230 CLR TEMP
2774 014330 005237 015230 INC TEMP
2775 014334 001375 BNE .-4
2776 014336 104402 TYPE
2777 014340 014556 MPFAIL
2778 014342 104411 CNVRT
2779 014344 014366 PFTAB
2780 014346 005037 001312 CLR ERRFLG
2781 014352 005037 001234 CLR LSTERR
2782 014356 104412 MSTCLR
2783 014360 104413 MEMCLR
2784 014362 000177 164626 JMP @RETURN
2785 014366 000001 PFTAB: 1
2786 014370 003 002 .BYTE 3,2
2787 014372 001226 TSTNO
2788
2789
2790 ;CHECK SWITCH REGISTER ROUTINE. CHECKS FOR ^G TO ALLOW CHANGING
2791 ;OF LOC.176.
2792 ;LOCATIONS USED:
2793 014374 000000 RDSW: .WORD 0
2794
2795
2796 014376 005737 000042 .CKSWR: TST @#42
2797 014402 001042 BNE OUT
2798 014404 022737 000176 001200 CMP #SWREG,SWR ;SOFTWARE SWITCH REGISTER PRESENT
2799 014412 001036 BNE OUT ;NO, GET OUT
2800 014414 105777 164564 TSTB @TKCSR ;YES, WAIT FOR
2801 014420 100033 BPL OUT ;READY, GET CHARACTER
2802 014422 017737 164560 013034 MOV @TKDBR,.MSG ;AND STRIP OFF
2803 014430 042737 177600 013034 BIC #177600,.MSG ;THE GARBAGE
2804 014436 122737 000007 013034 CMPB #7,.MSG ;IS IT A <^G>
2805 014444 001021 BNE OUT
2806 014446 104402 014524 TYPE,$CNTG
2807 014452 005137 014374 .CNTLU: COM @#RDSW
2808 014456 104402 014530 TYPE,$MSWR
2809 014462 104411 014516 CNVRT,SWREGC
2810 014466 104403 014537 INSTR,$MNEW
2811 014472 104405 PARAM
2812 014474 000000 0
2813 014476 177777 177777
2814 014500 000176 SWREG
2815 014502 000 001 .BYTE 0,1
2816 014504 104402 014554 TYPE,MCRLF
2817 014510 005037 014374 OUT: CLR @#RDSW
2818 014514 000002 RTI
2819 014516 000001 SWREGC: 1
2820 014520 006 002 .BYTE 6,2
2821 014522 000176 SWREG
2822 014524 057377 000107 $CNTG: .ASCIZ <377>/^G/
2823 014530 051777 051127 020075 $MSWR: .ASCIZ <377>/SWR= /
```

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

2824	014536	000				
2825	014537	040	047040	053505	SMNEW: .ASCIZ / NEW= /	
2826	014544	020075	000			
2827		014550			.EVEN	
2828	014550	020040	000077		MQM: .ASCIZ / ?/	
2829	014554	000377			MCRLF: .ASCIZ <377>	
2830	014556	050377	051127	043040	MPFAIL: .ASCIZ <377>/PWR FAILED. RESTART AT TEST /	
2831	014564	044501	042514	027104		
2832	014572	051040	051505	040524		
2833	014600	052122	040440	020124		
2834	014606	042524	052123	000040		
2835	014614	042777	042116	050040	MEPASS: .ASCIZ <377>/END PASS CZDQE /	
2836	014622	051501	020123	055103		
2837	014630	050504	020105	000040		
2838	014636	051377	000		MR: .ASCIZ <377>/R/	
2839	014641	377	051120	043517	MERR2: .ASCIZ <377>/PROGRAM INDICATES NO DEVICES PRESENT./	
2840	014646	040522	020115	047111		
2841	014654	044504	040503	042524		
2842	014662	020123	047516	042040		
2843	014670	053105	041511	051505		
2844	014676	050040	042522	042523		
2845	014704	052116	000056			
2846	014710	044777	051516	043125	MERR3: .ASCIZ <377>/INSUFFICIENT DATA! /	
2847	014716	044506	044503	047105		
2848	014724	020124	040504	040524		
2849	014732	000041				
2850	014734	052377	051505	020124	MTSTPC: .ASCIZ <377>/TEST PC-/	
2851	014742	041520	000055			
2852	014746	046377	041517	020113	MLOCK: .ASCIZ <377>/LOCK ON SELECTED TEST/	
2853	014754	047117	051440	046105		
2854	014762	041505	042524	020104		
2855	014770	042524	052123	000		
2856	014775	103	051123	020072	MCSRX: .ASCIZ /CSR: /	
2857	015002	000				
2858	015003	126	041505	020072	MVECX: .ASCIZ /VEC: /	
2859	015010	000				
2860	015011	120	051501	042523	MPASSX: .ASCIZ /PASSES: /	
2861	015016	035123	000040			
2862	015022	051105	047522	051522	MERRX: .ASCIZ /ERRORS: /	
2863	015030	020072	000			
2864	015033	377	052377	051505	MTSTN: .ASCIZ <377><377> /TEST NO: /	
2865	015040	020124	047516	020072		
2866	015046	000				
2867	015047	377	042523	020124	MNEW: .ASCIZ <377>/SET SWITCH REG TO DQ11'S DESIRED ACTIVE./	
2868	015054	053523	052111	044103		
2869	015062	051040	043505	052040		
2870	015070	020117	050504	030461		
2871	015076	051447	042040	051505		
2872	015104	051111	042105	040440		
2873	015112	052103	053111	027105		
2874	015120	000				
2875	015121	120	035103	000040	MERRPC: .ASCIZ /PC: /	
2876	015126	046777	050101	047440	XHEAD: .ASCIZ <377>/MAP OF DQ11 STATUS/<377>	
2877	015134	020106	050504	030461		
2878	015142	051440	040524	052524		
2879	015150	177523	000			

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

2880		015154		.EVEN	
2881	015154	000002		XSTATQ: 2	
2882	015156	006	003	.BYTE 6,3	
2883	015160	001244		TEMP1	
2884	015162	006	002	.BYTE 6,2	
2885	015164	001246		TEMP2	
2886				.EVEN	
2887					
2888				:BUFFERS FOR INPUT-OUTPUT	
2889					
2890	015166	000000		INBUF: 0	
2891		015230		.=. +40	
2892	015230	000000		TEMP: 0	
2893		015272		.=. +40	
2894	015272	000000		MDATA: 0	
2895		015334		.=. +40	
2896					
2897					
2898	015334	000010		CHRLNG: 10	
2899	015336	000351		DATAIN: 351	
2900	015340	000000		TMPDAT: 0	
2901	015342	000000		BCCPRV: 0	
2902	015344	000000		CALBCC: 0	
2903	015346	000000		BCCFBK: 0	
2904	015350	000000		XPOLY: 0	
2905	015352	000000		COUNT: 0	
2906	015354	000000		DATA: 0	
2907	015356	000000		SAVBCC: 0	
2908	015360	000000		SAVEPC: 0	
2909	015362	000000		SEC16: 0	
2910	015364	000000		SEC16X: 0	
2911	015366	000000		STORE1: 0	
2912	015370	000000		LOC1: 0	
2913	015372	026	026	.SYNC: .BYTE 26,26	
2914	015374	026	026	SYNC: .BYTE 26,26	
2915	015376	000000		TXBUFF: 0	
2916		016000		.=. +400	
2917	016000	000000		RXBUFF: 0	
2918		016402		.=. +400	
2919	016402	000000		.ERRTA: 0	
2920	016404	000000		0	:HALT 0
2921	016406	000000		0	
2922	016410	000000		0	
2923	016412	017026		DH0	:HALT 1
2924	016414	017402		DT0	
2925	016416	000000		C	
2926	016420	017052		DH1	:HALT 2
2927	016422	017414		DT1	
2928	016424	016635		EM0	
2929	016426	017122		DH2	:HALT 3
2930	016430	017436		DT2	
2931	016432	016730		EM2	
2932	016434	017234		DH4	:HALT 4
2933	016436	017506		DT4	
2934	016440	016664		EM1	
2935	016442	017201		DH3	:HALT 5

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

2936	016444	017470			DT3		
2937	016446	016664			EM1		
2938	016450	017310			DH5	;HALT 6	
2939	016452	017520			DT5		
2940	016454	016745			EM3		
2941	016456	017352			DH6	;HALT 7	
2942	016460	017542			DT6		
2943	016462	016772			EM4		::++E
2944	016464	000000			0	;HALT 10	
2945	016466	000000			0		
2946	016470	050377	042514	051501	MPOLY:	.ASCII <377>/PLEASE SET SWITCH REGISTER TO POLYNOMIAL YOU DESIRE /	
	016555	377	047524	041040		.ASCIZ <377>/TO BE PLACED INTO POLYNOMIAL REGISTER OF DQ11./	
	016635	377	040503	041514	EM0:	.ASCIZ <377>/CALCULATED BCC ERROR./	
	016664	046777	046505	051117	EM1:	.ASCIZ <377>/MEMORY TRANSFER TEST *DATA ERROR*/	
	016730	041777	041101	042514	EM2:	.ASCIZ <377>/CABLE TEST /	
	016745	377	050504	030461	EM3:	.ASCIZ <377>/DQ11 ERROR FLAG SET/	
	016772	051377	041130	041503	EM4:	.ASCIZ <377>/RXBCC ERROR BIT6 NOT CLEAR/	::++E
	017026	050377	046117	047131	DH0:	.ASCIZ <377>/POLYNOMIAL DQERR/	
	017052	050377	046117	047131	DH1:	.ASCIZ <377>/POLYNO EXPECTED RECEIVED SEC REG/	
	017122	050377	046117	047131	DH2:	.ASCIZ <377>/POLYNO CHAR SHIFTS EXPECTED RECEIVED REG/	
	017201	377	042101	051104	DH3:	.ASCIZ <377>/ADDRESS EXPECTED FOUND /	
	017234	042377	052101	020101	DH4:	.ASCII <377>/DATA COMPARISON ERROR /	
	017263	377	054105	042520		.ASCIZ <377>/EXPECTED RECEIVED/	
	017310	052377	020130	042101	DH5:	.ASCIZ <377>/TX ADD RX ADD EXPECTED FOUND /	
	017352	042377	051121	051503	DH6:	.ASCIZ <377>/DQRCR DQTCR DQERR /	
	017402	000002			.EVEN		
2947	017404	006	007		DT0:	2	
2948	017406	015350			.BYTE	6.7	
2949	017410	006	001		.BYTE	XPOLY	
2950	017412	001270				6.1	
2951	017414	000004			DT1:	SAVR5	
2952	017416	006	003		.BYTE	4	
2953	017420	015350				6.3	
2954	017422	006	005		.BYTE	XPOLY	
2955	017424	015344				6.5	
2956	017426	006	005		.BYTE	CALBCC	
2957	017430	001260				6.5	
2958	017432	002	001		.BYTE	SAVR1	
2959	017434	001270				2.1	
2960	017436	000006			DT2:	SAVR5	
2961	017440	006	002		.BYTE	6	
2962	017442	015350				6.2	
2963	017444	003	003		.BYTE	XPOLY	
2964	017446	015354				3.3	
2965	017450	002	006		.BYTE	DATA	
2966	017452	015352				2.6	
2967	017454	006	004		.BYTE	COUNT	
2968	017456	015344				6.4	
2969	017460	006	004		.BYTE	CALBCC	
2970	017462	001252				6.4	
2971	017464	002	002		.BYTE	TEMP4	
2972	017466	001254				2.2	
2973	017470	000003			DT3:	TEMP5	
2974	017472	006	003			3	
2975	017474	001256			.BYTE	6.3	
					SAVR0		

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

2976	017476	006	004		.BYTE	6.4
2977	017500	001254			TEMP5	
2978	017502	006	002		.BYTE	6.2
2979	017504	001252			TEMP4	
2980	017506	000002		DT4:	2	
2981	017510	006	004	.BYTE	6.4	
2982	017512	001254			TEMP5	
2983	017514	006	002	.BYTE	6.2	
2984	017516	001252			TEMP4	
2985	017520	000004		DT5:	4	
2986	017522	006	002		.BYTE	6.2
2987	017524	001256			SAVR0	
2988	017526	006	002		.BYTE	6.2
2989	017530	001260			SAVR1	
2990	017532	006	004		.BYTE	6.4
2991	017534	001254			TEMP5	
2992	017536	006	002		.BYTE	6.2
2993	017540	001252			TEMP4	
2994	017542	000003		DT6:	3	
2995	017544	006	002		.BYTE	6.2
2996	017546	001256			SAVR0	
2997	017550	006	002		.BYTE	6.2
2998	017552	001260			SAVR1	
2999	017554	006	002		.BYTE	6.2
3000	017556	001262			SAVR2	
3001	000001			.END		

CZDQEE.P11 25-JUL-78 15:28 CROSS REFERENCE TABLE -- USER SYMBOLS

LOBITS	013374	2538*	2570	2586#	2587													
LOCK	001220	791#	1291*	1346*	1957*	2003*	2021*	2465										
LOC1	015370	1699*	1744*	1745	1747	2912#												
LOGICA	012476	652	2415#															
LOKFLG	001313	828#																
LOLIM	013366	2535*	2568	2583#														
LPCNT	001224	793#	2448*	2449	2452*													
LSTERR	001234	797#	944*	2387*	2693	2695*	2781*											
MCRLF	014554	2508	2524	2616	2722	2816	2829#											
MCSRX	014775	2393	2856#															
MDATA	015272	2644	2655	2894#														
MEMCLR=	104413	860#	1128	1217	1278	1494	1539	1636	1890	1923	2078	2079	2111	2783				
MEPASS	014614	2391	2835#															
MERRPC	015121	2718	2875#															
MERRX	015022	2405	2862#															
MERR2	014641	762	1060	2839#														
MERR3	014710	992	2846#															
MISC. =	000012	632#	1317	1444	1461	1559	1576	1593	1606	1653	1679	1726	2006	2038				
		2052	2063	2137	2191	2241	2257	2289	2293	2309	2313	2374						
MLOCK	014746	1016	2852#															
MNEW	015047	985	2867#															
MPASSX	015011	2401	2860#															
MPFAIL	014556	2777	2830#															
MPOLY	016470	2097	2946#															
MQM	014550	2521	2828#															
MR	014636	1034	2838#															
MSTCLR=	104412	858#	1312	1357	1432	1433	1500	1546	1547	1639	1640	1767	1791	1815				
		1839	1863	1896	1929	1959	2005	2022	2170	2218	2220	2270	2782					
MTITLE	001000	771#	966															
MTSTN	015033	2714	2864#															
MTSTPC	014734	1025	2850#															
MVECX	015003	2397	2858#															
NEXT	001216	790#	1040*	1110*	1121	1194*	1277*	1405	1427*	1493*	1538*	1635*	1766*	1790*				
		1814*	1838*	1862*	1889*	1922*	1956*	1996*	2094*	2454	2750							
ODDBIT=	001000	615#	733															
OUT	014510	2797	2799	2801	2805	2817#												
PARAM =	104405	848#	1026	2811														
PARAM1	013222	2541#	2562															
PARERR	013276	2544	2546	2548	2557#	2567	2569	2571										
PARTI	013360	2560	2580#															
PASCNT	001230	795#	939*	2389*	2408	2431												
PFTAB	014366	2779	2785#															
POLY. =	000017	637#																
POPRO =	012600	584#	2743															
POP1SP=	005726	582#																
POP2SP=	022626	586#	1403															
PS =	177776	576#	673*	934*	1009*	2145*												
PUSHRO=	010046	583#	2740															
PUSH1S=	005746	581#																
PUSH2S=	024646	585#																
RDSW	014374	2473	2522	2559	2793#	2807*	2817*											
RESREG	014174	2734	2737#															
RESTAR	014312	2764	2770#															
RESTRT	012506	2410	2413	2420#														
RESOS =	104407	852#	2737															
RETURN	001214	789#	946*	1029	1033*	1035	1039*	1121*	1122	1405*	1406	1430*	1431	1492*				

CZDQE MACY11 30A(1052) 28-JUL-78 13:43 PAGE 68
CZDQEE.P11 25-JUL-78 15:28 CROSS REFERENCE TABLE -- USER SYMBOLS

.PFAIL	014300	645	936	2763#	2771			
.RES05	013436	853	2605#					
.SAV05	013376	851	2591#					
.SCOPE	012552	839	2438#					
.SCOPI	012664	841	2462#					
.START	001512	696	934#	946				
.SYNC	015372	1143	1227	1372	2134	2188	2913#	
.TRPSR	013716	649	2674#					
.TRPTA	001314	837#	2679					
.TYPE	012704	843	2470#					

. ABS. 017560 000

ERRORS DETECTED: 0

DSKZ:CZDQEE,DSKZ:CZDQEE,SEQ=DSKZ:CZDQXX,MAC,DSKZ:CZDQEE.P11
RUN-TIME: 6 10 1 SECONDS
RUN-TIME RATIO: 59/18=3.2
CORE USED: 18K (35 PAGES)

DOCUMENT PAGES: 66