DUP11

DATA&FUNC&DECMODE TEST

MD-11-DZDPD-A

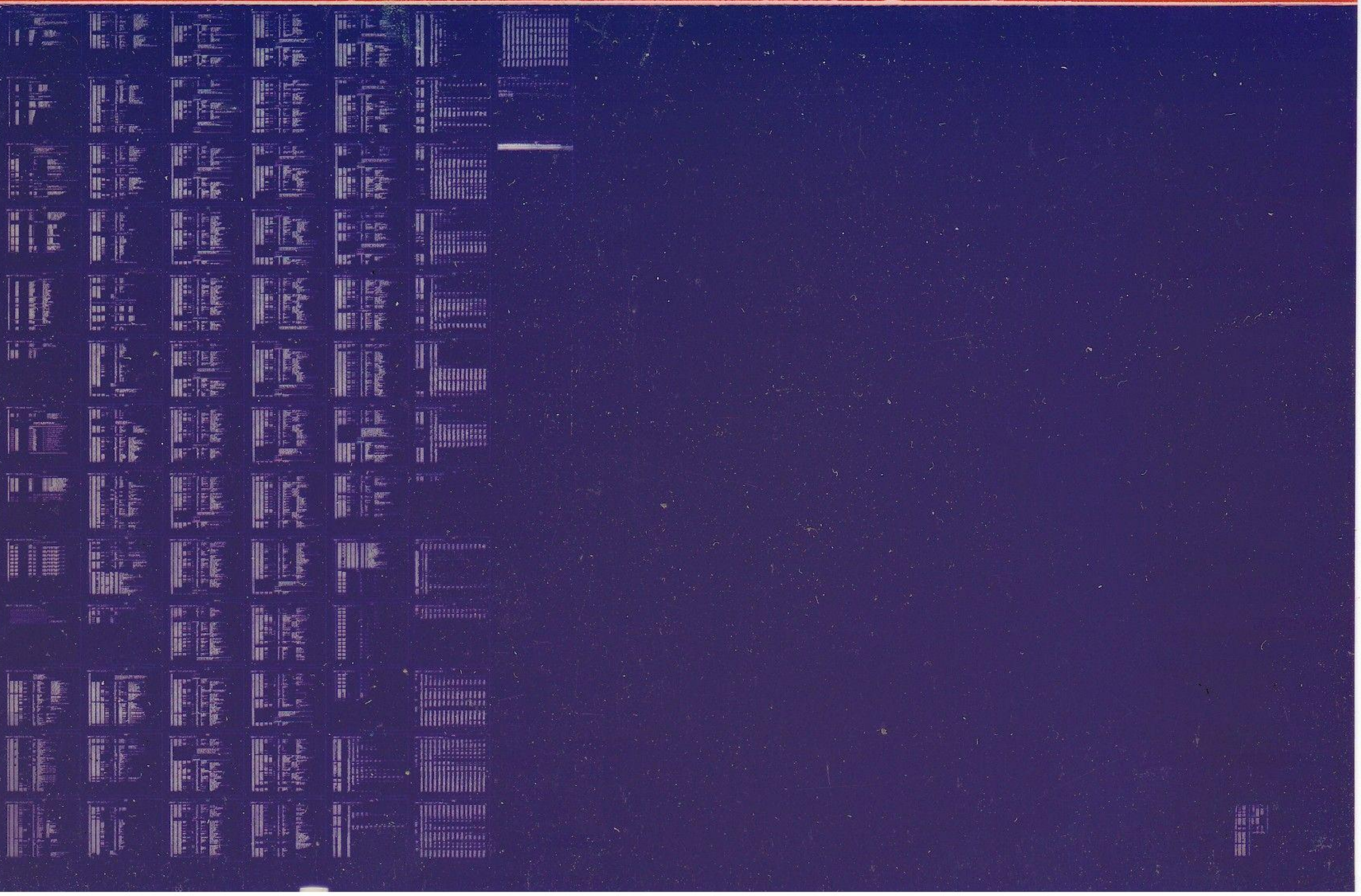
EP-DZDPD-A-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA



```
                                ;*MAINDEC-11-DZDPD-A /(377)/DUP-11 OFFLINE SDLC AND DEC MODE DATA AND FUNCTION TESTS
                                ;*COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
                                ;*-------------------------------------------------------------------

                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE "MAINDEC-11-DZDPD-A /(377)/DUP-11 OFFLINE SDLC AND DEC MODE D
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
                                        ;AND THEN RESUME TESTING


                                ;SWITCH REGISTER OPTIONS
                                ;----------------------

            100000              SW15=100000             ;=1,HALT ON ERROR
            040000              SW14=40000              ;=1,LOOP ON CURRENT TEST
            020000              SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
            010000              SW12=10000              ;=1,DELETE TYPEOUT/BELL ON ERROR.
            004000              SW11=4000               ;=1,INHIBIT ITERATIONS
            002000              SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
            001000              SW09=1000               ;=1,LOOP WITH CURRENT DATA
            000400              SW08=400                ;=1,LOOP ON ERROR
            000200              SW07=200
            000100              SW06=100
            000040              SW05=40
            000020              SW04=20
            000010              SW03=10

                                                        ;SELECT DUP'S DESIRED ACTIVE
                                                        ;NOTE:THIS MUST NOT EXCEED ORIGINAL COUNT
            000004              SW02=4                  ;LOCK ON TEST SELECT
            000002              SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
            000001              SW00=1                  ;ENTER PARAMETERS
```

```
                        ;REGISTER DEFINITIONS
                        ;--------------------

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

                        ;LOCATION EQUIVALENCIES
                        ;----------------------

177776                  PS=177776       ;PROCESSOR STATUS WORD
001150                  STACK=1150      ;START OF PROCESSOR STACK

                        ;INSTRUCTION DEFINITIONS
                        ;-----------------------

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


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

```
                                ;;*****************************************************************************
                                ;-----------------------------------------------------------------------------
                                        ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
                                        ;THE STANDARD "TRAP CATCHER" IS PLACED
                                        ;BETWEEN ADDRESS 0 TO ADDRESS 776.
                                        ;IT LOOKS LIKE "PC+2 HALT".
                                ;-----------------------------------------------------------------------------
                                ;;*****************************************************************************

                        000000  .=0
                                        ;STANDARD INTERRUPT VECTORS
                                        ;-------------------------

                        000024  .=24
000024  004776                          .PFAIL                  ;POWER FAIL HANDLER
000026  000340                         340                     ;SERVICE AT LEVEL 7
000030  004350                         .HLT                    ;ERROR HANDLER
000032  000340                         340                     ;SERVICE AT LEVEL 7
000034  004316                         .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                         340                     ;SERVICE AT LEVEL 7
                        000040  .=40
000040  000000                         0                       ;SAVE FOR ACT-11 OR DDP2
000042  000000                         0                       ;RETURN ADDRESS IF UNDER ACT-11 OR DDP2
000044  000000                         0                       ;SAVE FOR ACT-11 OR DDP2
000046  003104                         $ENDAD                  ;FOR USE WITH ACT-11 OR DDP2
                        000052  .=52
000052  000000                         0                       ;ACT-11 PROGRAM CHARACTERISTICS

                        000174  .=174
000174  000000          DISPREG:0                              ;SOFTWARE DISPLAY REGISTER
000176  000000          SWREG:  0                              ;SOFTWARE SWITCH REGISTER
                        000200  .=200
000200  000137  001562                  JMP     .START          ;GO TO START OF PROGRAM


                        001000  .=1000
001000  005377  040515  047111  MTITLE: .ASCIZ  <377><12>/MAINDEC-11-DZDPD-A /<377>/DUP-11 OFFLINE SDLC AND DEC MODE DAT
(2)
                        001200  .=1200
                                        ;SWR AND LIGHTS
                                        ;--------------

001200  177570          DISPLAY:       177570                  ;11/45 CONSOLE LIGHTS
001202  177570          SWR:    177570                         ;INDIRECT POINTER TO SWITCH REGISTER

                                ;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
                                ;--------------------------------------------------

001204  177560          TKCSR:  177560                         ;TELETYPE KEYBOARD CONTROL REGISTER
001206  177562          TKDBR:  177562                         ;TELETYPE KEYBOARD DATA BUFFER
001210  177564          TPCSR:  177564                         ;TELEPRINTER CONTROL REGISTER
001212  177566          TPDBR:  177566                         ;TELEPRINTER DATA BUFFER

                                ;PROGRAM CONTROL PARAMETERS
                                ;--------------------------
```

```
                                RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
                                NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
                                LOCK:   0                       ;ADDRESS FOR LOCK ON CURRENT DATA
                                ICOUNT: 1                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
                                LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
                                TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
                                PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
                                ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
                                LSTERR: 0                       ;PC OF LAST ERROR CALL

                                                ;PROGRAM VARIABLES
                                                ;-----------------

                                TEMP1:  0                       ;TEMPORARY STORAGE
                                TEMP2:  0                       ;TEMPORARY STORAGE
                                TEMP3:  0                       ;TEMPORARY STORAGE
                                TEMP4:  0                       ;TEMPORARY STORAGE
                                TEMP5:  0                       ;TEMPORARY STORAGE
                                SAVR0:  0                       ;R0 STORAGE
                                SAVR1:  0                       ;R1 STORAGE
                                SAVR2:  0                       ;R2 STORAGE
                                SAVR3:  0                       ;R3 STORAGE
                                SAVR4:  0                       ;R4 STORAGE
                                SAVR5:  0                       ;R5 STORAGE
                                SAVSP:  0                       ;STACK POINTER STORAGE
                                SAVPC:  0                       ;PROGRAM COUNTER STORAGE

                                SAVR0A: 0                       ;R0 STORAGE
                                SAVR1A: 0                       ;R1 STORAGE
                                SAVR2A: 0                       ;R2 STORAGE
                                SAVR3A: 0                       ;R3 STORAGE
                                SAVR4A: 0                       ;R4 STORAGE
                                SAVR5A: 0                       ;R5 STORAGE
                                SAVSPA: 0                       ;STACK POINTER STORAGE
                                SAVPCA: 0                       ;PROGRAM COUNTER STORAGE

                                DUPACTV: .BLKB  1               ;DUP11'S SELECTED ACTIVE.
                                DUPNUM:  .BLKB  1               ;OCTAL NUMBER OF DUP11'S.
                                SAVACT:  .BLKB  1               ;ORIGINAL ACTV.  DEVICES.
                                SAVNUM:  .BLKB  1               ;WORKABLE NUMBER.
                                RUN:     .BLKB  1               ;POINTER ONE PAST RUNNING DEVICE.
                                .EVEN
                                CREAM:   DUP.MAP                ;TABLE POINTER.
```

```
                                        ;CONTROL REGISTER DEFINITIONS
                                        ;----------------------------
                                ;RXCSR BIT DEFINITIONS
100000                  DSCA=BIT15              ;DATA SET CHANGE A
040000                  RING=BIT14              ;RING
020000                  CTS=BIT13               ;CLR TO SEND
010000                  CARDET=BIT12            ;CARRIER DETECT
004000                  RECACT=BIT11            ;REC ACTIVE
002000                  SRD=BIT10               ;SEC REC DATA
001000                  DSR=BIT9                ;DATA SET RDY
000400                  STPSYN=BIT8             ;STRIP SYNC
000200                  RXDONE=BIT7             ;REC DONE
000100                  RINTEN=BIT6             ;REC INTR ENABLE
000040                  DSINTE=BIT5             ;DSC INTR ENABLE
000020                  RCVEN=BIT4              ;REC ENABLE
000010                  STD=BIT3                ;SEC XMIT DATA
000004                  RTS=BIT2                ;REQ TO SEND
000002                  DTR=BIT1                ;DATA TERM RDY
000001                  DSCB=BIT0               ;DATA SET CHANGE B
                                ;RXDBUF BIT DEFINITIONS
100000                  RXDERR=BIT15            ;REC DATA ERROR
040000                  OVRRUN=BIT14            ;OVERRUN ERROR
010000                  CRCERR=BIT12            ;CRC ERROR
002000                  RABORT=BIT10            ;REC ABORT
001000                  REOM=BIT9               ;REC END OF MESSAGE
000400                  RSOM=BIT8               ;REC START OF MESSAGE
                                ;PARCSR BIT DEFINITIONS
100000                  DECMOD=BIT15            ;DEC MODE (DDCMP)
001000                  CRCEN=BIT9              ;CRC ENABLE
010000                  PRISEC=BIT12            ;PRI/SEC SELECT
                                ;TXCSR BIT DEFINITIONS
100000                  TXDLAT=BIT15            ;TX DATA LATE
040000                  MTDATA=BIT14            ;MAINT DATA OUT
020000                  CLK=BIT13               ;CLK
010000                  MMODEB=BIT12            ;MAINT MODE B
004000                  MMODEA=BIT11            ;MAINT MODE A
002000                  BITW=BIT10              ;BIT WINDOW INPUT
001000                  TXACT=BIT9              ;TX ACTIVE
000400                  MRESET=BIT8             ;MASTER RESET
000200                  TXDONE=BIT7             ;XMIT DONE
000100                  TXINTE=BIT6             ;XMIT DONE INTR ENABLE
000020                  SEND=BIT4               ;SEND
000010                  HDXEN=BIT3              ;HDX/FDX
                                ;TXCSR WRD DEFINITIONS
000000                  USER=0                  ;USER MODE
014000                  MMODE=14000             ;MAINT INT MODE
010000                  MEXT=10000              ;MAINT EXT MODE
004000                  SYSTST=4000             ;SYSTEM TEST MODE

                                ;TXDBUF BIT DEFINITIONS
                                ;----------------------
100000                  RCRC7T=BIT15
040000                  RCRCIN=BIT14
020000                  TCRC7T=BIT13
010000                  TCRCIN=BIT12
```

```
                    004000                  TIMER=BIT11             ;MAINTENANCE TIMER
                    002000                  TABORT=BIT10            ;TRANSMIT ABORT
                    001000                  TEOM=BIT9               ;TRANSMIT END OF MESSAGE
                    000400                  TSOM=BIT8               ;TRANSMIT START OF MESSAGE

                                            ;MISC. PROGRAM DEFINITIONS
                                            ;-------------------------
001320  000000                              PRIRTY:.WORD 0
001322  000001                              TCNFLG: .BLKB 1
001323  000001                              OPCLRJ: .BLKB 1
001324  000000                              DATA:   .WORD   0
001326  000000                              SHIFTS: .WORD   0
001330  000000                              MIND:   .WORD   0
001332  000000                              FLAG:   .WORD   0
001334  000001                              STJMFL: .BLKW 1
001336  000001                              SRJMFL: .BLKW 1
```

```
                                        ;PROGRAM CONTROL FLAGS
                                        ;---------------------
001340  000             INIFLG: .BYTE   0               ;PROGRAM INITIALIZATION FLAG
001341  000             ERRFLG: .BYTE   0               ;ERROR OCCURED FLAG
001342  000             LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
001343  000             QV.FLG: .BYTE   0               ;QUICK VERIFY FLAG.
                                                        ;ON FIRST PASS OF EACH DUP11 ITERATIONS
                                                        ;WILL BE SUPPRESSED
                        .EVEN
        000000          $Y=0

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

                        ;;**************************************************************************
                        ;--------------------------------------------------------------------------
001344                  .TRPTAB:
        104400          SCOPE=TRAP+0            ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001344  003160                  .SCOPE
        104401          SCOP1=TRAP+1            ;CALL TO LOOP ON CURRENT DATA HANDLER
001346  003312                  .SCOP1
        104402          TYPE=TRAP+2             ;CALL TO TELETYPE OUTPUT ROUTINE
001350  003336                  .TYPE
        104403          INSTR=TRAP+3            ;CALL TO ASCII STRING INPUT ROUTINE
001352  003412                  .INSTR
        104404          INSTER=TRAP+4           ;CALL TO INPUT ERROR HANDLER
001354  003516                  .INSTER
        104405          PARAM=TRAP+5            ;CALL TO NUMERICAL DATA INPUT ROUTINE
001356  003536                  .PARAM
        104406          SAVO5=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
001360  003736                  .SAVO5
        104407          RESO5=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
001362  003776                  .RESO5
        104410          CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
001364  004030                  .CONVRT
        104411          CNVRT=TRAP+11           ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
001366  004034                  .CNVRT
        104412          PKCLK=TRAP+12           ;CALL TO CLOCK ROUTINE
001370  004734                  .PKCLK
        104413          SETFLG=TRAP+13          ;CALL TO TELETYPE INPUT ROUTINE
001372  004242                  .SETFLG

                        ;--------------------------------------------------------------------------
                        ;;**************************************************************************
```

```
                                        ;DUP11 VECTOR AND REGISTER INDIRECT POINTERS

001374  000000          DUPRVC: 0             ;POINTER TO DUP11 RECEIVER INTERRUPT VECTOR
001376  000000          DUPRPS: 0             ;POINTER TO DUP11 RECEIVER INTERRUPT SERVICE PS
001400  000000          DUPTVC: 0             ;POINTER TO DUP11 TRANSMITTER INTERRUPT VECTOR
001402  000000          DUPTPS: 0             ;POINTER TO DUP11 TRANSMITTER INTERRUPT SERVICE PS
001404  000000          RXCSR: 0              ;POINTER TO DUP11 RECEIVER STATUS REGISTER
001406  000000          RXDBUF: 0             ;POINTER TO DUP11 RECEIVER DATA BUFFER
001410  000000          PARCSR: 0             ;POINTER TO DUP11 PARAMETER STATUS REGISTER
001412  000000          TXCSR: 0              ;POINTER TO DUP11 TRANSMITTER STATUS REGISTER
001414  000000          TXDBUF: 0             ;POINTER TO DUP11 TRANSMITTER DATA BUFFER
001416  000000          DUPSEC: 0             ;POINTER TO DUP11 SECONDARY REGISTER SELECT REGISTER
001420  000000          HUPPSR: 0             ;POINTER TO PARAMETER STATUS HIGH BYTE
001422  000000          HUPRBF: 0             ;POINTER TO RECEIVER BUFFER HIGH BYTE
001424  000000          HUPRCR: 0             ;POINTER TO RECEIVER CONTROL REG HIGH BYTE
001426  000000          HUPTBF: 0             ;POINTER TO TRANSMITTER BUFFER HIGH BYTE
001430  000000          HUPTCR: 0             ;POINTER TO TRANSMITTER CONTROL REG HIGH BYTE


                                        ;DUP11 CONTROL INDICATORS FOR CURRENT DUP11 UNDER TEST
                                        ;-----------------------------------------------------

001432      000         MASK.A: .BYTE 000     ;LAST CHAR TO TEST AND PARITY MASK

001433      010         CLK.A:  .BYTE 8.      ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR

001434  000000          L00.00: 000000        ;PARAMETERS
```

```
                                        ;DUP11 STATUS TABLE AND ADDRESS ASSIGNMENTS
                                        ;------------------------------------------

        001500                  .=1500
001500                          DUP.MAP:
001500  000001                  DUPCR0: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 0
001502  000001                  DUPTR0: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 0
001504  000001                  DUP0.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 0

001506  000001                  DUPCR1: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 1
001510  000001                  DUPTR1: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 1
001512  000001                  DUP1.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 1

001514  000001                  DUPCR2: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 2
001516  000001                  DUPTR2: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 2
001520  000001                  DUP2.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 2

001522  000001                  DUPCR3: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 3
001524  000001                  DUPTR3: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 3
001526  000001                  DUP3.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 3

001530  000001                  DUPCR4: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 4
001532  000001                  DUPTR4: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 4
001534  000001                  DUP4.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 4

001536  000001                  DUPCR5: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 5
001540  000001                  DUPTR5: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 5
001542  000001                  DUP5.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 5

001544  000001                  DUPCR6: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 6
001546  000001                  DUPTR6: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 6
001550  000001                  DUP6.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 6

001552  000001                  DUPCR7: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 7
001554  000001                  DUPTR7: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 7
001556  000001                  DUP7.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 7

001560  000000                  DUP.END:        000000
```

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
C	O	N	T	R	O	L		R	E	G	I	S	T	E	R
								*	V	E	C	T	O	R	*
A	B	C	D	E	F	G	H	*	*	S	Y	N	C	*	*

DEFINITIONS

A- OPTIONAL CLEAR JUMPER IN=1
B- TURNAROUND CONNECTOR ON=1
C-
D-

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

001562  012737  000340  177776  .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
001570  012706  001150                  MOV     #STACK,SP               ;SET UP STACK
001574  012737  004776  000024          MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
001602  113737  001311  001313          MOVB    DUPNUM,SAVNUM           ;SAVE NUMBER OF DEVICES IN SYSTEM
001610  005037  001230                  CLR     PASCNT                  ;CLEAR PASS COUNT
001614  105037  001341                  CLRB    ERRFLG                  ;CLEAR ERROR FLAG
001620  105037  001343                  CLRB    QV.FLG                  ;ZERO QUICK VERIFY FLAG
001624  012737  001500  001316          MOV     #DUP.MAP,CREAM          ;GET MAP POINTER.
001632  112737  000001  001314          MOVB    #1,RUN                  ;POINT POINTER TO FIRST DEVICE.
001640  005037  001232                  CLR     ERRCNT                  ;CLEAR ERROR COUNT
001644  005037  001234                  CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
001650  012737  000001  001226          MOV     #1,TSTNO                ;SET UP FOR TEST 1
001656  012737  001562  001214          MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
                                                                        ;TESTING STARTS
001664  013746  000006                  MOV     @#6,-(SP)               ;SAVE CURRENT VECTORS
001670  013746  000004                  MOV     @#4,-(SP)               ;
001674  012737  001710  000004          MOV     #12$,@#4                ;SETUP FOR TIMEOUT
001702  005777  177274                  TST     @SWR                    ;REFERENCE HARDWARE SWITCH REG
001706  000407                          BR      13$                     ;BR IF IT EXISTS
001710  012737  000176  001202  12$:    MOV     #SWREG,SWR      ;POINT TO SOFT SWR
001716  012737  000174  001200          MOV     #DISPREG,DISPLAY        ;POINT TO SOFT DISPLAY REG
001724  022626                          CMP     (SP)+,(SP)+             ;ADJUST STACK
001726  012637  000004          13$:    MOV     (SP)+,@#4               ;RESTORE VECTORS
001732  012637  000006                  MOV     (SP)+,@#6               ;
001736  105737  001340                  TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
001742  001401                          BEQ     11$
001744  000410                          BR      6$
001746  022737  003104  000042  11$:    CMP     #$ENDAD,@#42    ;IF ACT-11 AUTO MODE,
001754  001404                          BEQ     6$              ;DON'T TYPE ID
001756  104402  001000                  TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
001762  105137  001340                  COMB    INIFLG                  ;IF NOT SET FLAG AND DO
001766  105777  177210          6$:     TSTB    @SWR                    ;BIT7=1??
001772  100002                          BPL     10$
001774  000137  002520                  JMP     1$
002000                          10$:
002000  032777  000001  177174          BIT     #SW00,@SWR      ;ENTER PARAMETERS
002006  001002                          BNE     .+6             ;YES
002010  000137  002360                  JMP     21$             ;NO
002014  105137  001332                  COMB    FLAG
002020  112737  000001  001340          MOVB    #1,INIFLG       ;SET TO MANUAL ENTRY
002026  012700  001500                  MOV     #DUP.MAP,R0     ;CLR MAP
002032  005020                  68$:    CLR     (R0)+
002034  020027  001560                  CMP     R0,#DUP.END     ;DONE WITH MAP?
002040  001374                          BNE     68$             ;BR IF NO
002042  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
002044  005421                          MCSR    ;MESSAGE
002046  104405                          PARAM   ;CONVERT STRING
002050  160000                          160000  ;LOW LIMIT
```

```
                                                    175500  ;HIGH LIMIT
                                                    DUPCRD  ;STORE AT THIS LOCATION
                                        .BYTE       1       ;MASK
                                        .BYTE       1       ;HOW MANY TIMES + 2
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MVEC    ;MESSAGE
                                                    PARAM   ;CONVERT STRING
                                                    300     ;LOW LIMIT
                                                    770     ;HIGH LIMIT
                                                    DUPTRD  ;STORE AT THIS LOCATION
                                        .BYTE       1       ;MASK
                                        .BYTE       1       ;HOW MANY TIMES + 2
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MPAR    ;MESSAGE
                                                    PARAM   ;CONVERT STRING
                                                    4       ;LOW LIMIT
                                                    7       ;HIGH LIMIT
                                                    TEMP2   ;STORE AT THIS LOCATION
                                        .BYTE       0       ;MASK
                                        .BYTE       1       ;HOW MANY TIMES + 2
                                                    MOV     TEMP2,PRIRTY    ;SAVE PRIORITY
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MTOTAL  ;MESSAGE
                                                    PARAM   ;CONVERT STRING
                                                    1       ;LOW LIMIT
                                                    8.      ;HIGH LIMIT
                                                    TEMP1   ;STORE AT THIS LOCATION
                                        .BYTE       0       ;MASK
                                        .BYTE       1       ;HOW MANY TIMES + 2
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MJMPR   ;MESSAGE
                                                    SETFLG  ;SET FLAG BASED UPON INPUT STRING
                                                    OPCLRJ  ;THIS FLAG
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MTCN    ;MESSAGE
                                                    SETFLG  ;SET FLAG BASED UPON INPUT STRING
                                                    TCNFLG  ;THIS FLAG
                                                    TSTB    TCNFLG
                                                    BEQ     71$
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MSTJM   ;MESSAGE
                                                    SETFLG  ;SET FLAG BASED UPON INPUT STRING
                                                    STJMFL  ;THIS FLAG
                                                    INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
                                                    MSRJM   ;MESSAGE
                                                    SETFLG  ;SET FLAG BASED UPON INPUT STRING
                                                    SRJMFL  ;THIS FLAG
                                        71$:        TSTB    OPCLRJ
                                                    BEQ     69$
                                                    BIS     #BIT15,DUPO.A
                                        69$:        TSTB    TCNFLG
                                                    BEQ     70$
                                                    BIS     #BIT14,DUPO.A
                                        70$:        MOVB    #1,SAVACT
                                                    MOVB    TEMP1,DUPNUM
                                                    MOVB    TEMP1,SAVNUM
```

```
                                                65$:    DEC     TEMP1
                                                        BEQ     64$
                                                        SEC
                                                        ROLB    SAVACT
                                                        BR      65$
                                                64$:    MOVB    SAVACT,TEMP2    ;# OF TIMES
                                                        MOVB    SAVACT,DUPACTV
                                                        CLC
                                                        RORB    TEMP2
                                                        MOV     #DUPCR0,R0
                                                        MOV     #DUPCR1,R1
                                                67$:    CLC
                                                        RORB    TEMP2
                                                        BCC     66$
                                                        MOV     (R0)+,(R1)
                                                        ADD     #10,(R1)+       ;CSR
                                                        MOV     (R0)+,(R1)
                                                        ADD     #10,(R1)+       ;VECTOR
                                                        MOV     (R0)+,(R1)+     ;PARAMETERS
                                                        BR      67$
                                                21$:    MOV     #DUP.MAP,R0     ;SETUP TO CLEAR MAP
                                                20$:    CLR     (R0)+           ;CLEAR
                                                        CMP     R0,#DUP.END     ;CHECK FOR FINISH
                                                        BNE     20$             ;BR IF MORE TO GO
                                                        MOV     #DUP.MAP,R0     ;SETUP TO DEFAULT
                                                        MOV     #160050,(R0)    ;LOAD CSR
                                                        MOV     #770,2(R0)      ;LOAD VECTOR
                                                        MOV     #140026,4(R0)   ;LOAD PARAMETERS AND SYNC
                                                        MOVB    #5,PRIRTY       ;LOAD PRIORITY
                                                        MOV     #1,R0           ;SAVE CORE THIS WAY
                                                        MOVB    R0,DUPACTV      ;PRESET PROGRAM CONTROLS
                                                        MOVB    R0,DUPNUM       ;DITTO
                                                        MOVB    R0,SAVACT       ;DITTO
                                                        MOVB    R0,SAVNUM       ;DITTO
                                                        MOVB    R0,TCNFLG       ;DITTO
                                                        MOVB    R0,OPCLRJ       ;DITTO
                                                66$:
                                                16$:    TYPE    ,XHEAD                  ;TYPE HEADER
                                                        MOV     #DUP.MAP,TEMP1          ;SET POINTER
                                                5$:     MOV     @TEMP1,TEMP2            ;SET DATA
                                                        BEQ     1$                      ;ALL DONE WITH DATA
                                                        CONVRT                          ;
                                                        XSTATQ                          ;
                                                        ADD     #2,TEMP1                ;UPDATE POINTER
                                                        BR      5$
                                                1$:     BIT     #SW00,@SWR
                                                        BEQ     7$
                                                        TST     FLAG
                                                        BNE     7$
                                                        JMP     10$
                                                7$:     CLR     FLAG
                                                        TST     @#42                    ;IS PROGRAM RUNNING UNDER MONITOR
                                                        BNE     3$                      ;BR IF YES
                                                        BIT     #SW03,@SWR              ;SELECT SPECIFIC DEVICES??
                                                        BEQ     3$                      ;BR IF NO.
                                                        TYPE    ,MNEW                   ;TYPE THE MESSAGE.
```

```
                                                CLR     R0              ;ZERO DATA LIGHTS
                                                HALT                    ;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
                                                CMPB    @SWR,SAVACT     ;IS THE NUMBER VALID?
                                                BLOS    2$              ;BR IF NUMBER IS OK.
                                                TYPE    ,MERR3          ;TELL USER OF INVALID NUMBER.
                                                HALT                    ;STOP EVERY THING.
                                                BR      .-2             ;RESTART THE PROGRAM AGAIN.
                                        2$:     MOVB    @SWR,DUPACTV    ;GET NEW DEVICE PATTERN
                                                MOVB    DUPACTV,R0      ;SHOW THE USER WHAT HE SELECTED.
                                                BIC     #^C<377>,R0     ;USE ONLY LOW BYTE.
                                                HALT                    ;CONTINUE DYNAMIC SWITCHES.
                                        3$:     MOV     #300,R0         ;PREPARE TO CLEAR THE FLOATING
                                                MOV     #302,R1         ;VECTOR AREA. 300-776
                                        4$:     MOV     R1,(R0)+        ;START PUTTING "PC+2 - HALT"
                                                CLR     (R1)+           ;IN VECTOR AREA.
                                                CMP     (R0)+,(R1)+     ;POP POINTERS
                                                CMP     #1000,R0        ;ALL DONE??
                                                BNE     4$              ;BR IF NO.

                                                ;TEST START AND RESTART
                                                ;----------------------

                177776  .BEGIN: MOV     #340,PS         ;LOCK OUT INTERRUPTS
                                MOV     #STACK,SP       ;SET UP STACK
                                TST     @#42            ;IS PROGRAM UNDER MONITOR CONTROL
                                BNE     2$              ;BR IF YES
                176274          BIT     #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
                                BEQ     1$              ;BR IF NO LOCK DESIRED.
                                TYPE    ,MLOCK          ;TYPE LOCK SELECTED.
                                MOV     #NOP,TTST       ;ADJUST SCOPE ROUTINE.
                                MOV     #NOP,TTST+2     ;SET UP TO LOCK
                                BR      2$              ;CONTINUE ALONG.
        003174  003306  1$:     MOV     BRW,TTST        ;PREPARE NORMAL SCOPE ROUTINE
        003176  003310          MOV     BRX,TTST+2      ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
        001214  006152  2$:     MOV     #CYCLE,RETURN   ;START AT "CYCLE" FIND WHICH DEVICE TO TEST
                005130          TYPE    ,MR             ;TYPE R
        176230  000177          JMP     @RETURN         ;START TESTING
```

```
                                                ;END OF PASS
                                                ;TYPE NAME OF TEST
                                                ;UPDATE PASS COUNT
                                                ;CHECK FOR EXIT TO ACT-11
                                                ;RESTART TEST

002764  005037  001234          .EOP:   CLR     LSTERR          ;CLEAR LAST ERROR PC
002770  105037  001341                  CLRB    ERRFLG          ;CLEAR ERROR FLAG
002774  005237  001230                  INC     PASCNT          ;UPDATE PASS COUNT
003000  013777  001230  176172          MOV     PASCNT,@DISPLAY ;DISPLAY PASS COUNT
003006  104402  005105                  TYPE    ,MEPASS         ;TYPE END PASS
003012  104402  005267                  TYPE    ,MCSRX          ;TYPE CSR
003016  104411  003130                  CNVRT   ,XCSR           ;SHOW IT
003022  104402  005275                  TYPE    ,MVECX          ;TYPE VECTOR
003026  104411  003136                  CNVRT   ,XVEC           ;SHOW IT
003032  104402  005303                  TYPE    ,MPASSX         ;TYPE PASSES
003036  104411  003144                  CNVRT   ,XPASS          ;SHOW IT
003042  104402  005314                  TYPE    ,MERRX          ;TYPE ERRORS
003046  104411  003152                  CNVRT   ,XERR           ;SHOW IT
003052  105337  001313                  DECB    SAVNUM          ;ARE ALL DEVICES TESTED?
003056  001017                          BNE     RESTRT          ;BR IF NO.
003060  112737  000377  001343          MOVB    #377,QV.FLG     ;SET THE QUICK VERIFY FLAG.
003066  113737  001311  001313          MOVB    DUPNUM,SAVNUM   ;RESTORE THE COUNT
003074  013701  000042                  MOV     @#42,R1         ;CHECK FOR ACT-11 OR DDP
003100  001406                          BEQ     RESTRT          ;IF NOT, CONTINUE TESTING
003102  000005                          RESET                   ;STOP THE SHOW--CLEAR THE WORLD
003104                          $ENDAD:
003104  004711                          JSR     PC,(R1)
003106  000240                          NOP
003110  000240                          NOP
003112  000240                          NOP
003114  000240                          NOP
003116  012737  006152  001214  RESTRT: MOV     #CYCLE,RETURN
003124  000137  006152                  JMP     CYCLE
003130  000001                  XCSR:   1
003132     006     002                  .BYTE   6,2
003134  001404                          RXCSR
003136  000001                  XVEC:   1
003140     003     002                  .BYTE   3,2
003142  001374                          DUPRVC
003144  000001                  XPASS:  1
003146     006     002                  .BYTE   6,2
003150  001230                          PASCNT
003152  000001                  XERR:   1
003154     006     002                  .BYTE   6,2
003156  001232                          ERRCNT

                                                ;SCOPE LOOP AND INTERATION HANDLER

003160  005037  001234          .SCOPE: CLR     LSTERR          ;CLEAR LAST ERROR PC
003164  010016                          MOV     R0,(SP)         ;SAVE R0 ON STACK
003166  032777  040000  176006          BIT     #BIT14,@SWR     ;LOOP ON TEST?
003174  001407                  TTST:   BEQ     1$              ;BR IF NO (IF LOCK SW01 = 1;THIS LOCATION  = 240)
003176  000437                          BR      3$              ;GO TO 3$       (DITTO)
003200  105777  176000                  TSTB    @TKCSR          ;KYBD DONE?
003204  100034                          BPL     3$              ;BR IF NO (LOCK:  HIT A KEY ON TTY TO GO TO NEXT TEST)
```

```
                                                ;CLR DONE BIT
                                                ;CONTINUE
                                                ;DELETE ITERATION (QUICK PASS)?
                                                ;BR IF YES
                                                ;HAS FIRST PASS BEEN COMPLETED?
                                                ;BR IF QUICK VERIFY
                                                ;UPDATE ITERATION COUNTER
                                                ;ALL ITERATIONS DONE?
                                                ;BR IF NOT YET
                                                ;PREPARE FOR NEW TEST
                                                ;START ICOUNT AT ZERO
                                                ;RESET ITERATIONS
                                                ;GET NEXT TEST
                                                ;POP R0 OFF STACK
                                                ;FAKE AN RTI
                                                ;GO DO THE TEST
                                                ;CHECK FOR FREEZE ON CURRENT DATA
                                                ;--------------------------------
                                                ;IS SW09=1(SET)?
                                                ;BR IF NOT SET.
                                                ;GOTO THE ADDRESS IN LOCK.
                                                ;GO BACK.
                                                ;TELETYPE OUTPUT ROUTINE
                                                ;----------------------
                                                ;SAVE R5 ON THE STACK.
                                                ;GET ADDRESS OF MESSAGE.
                                                ;POP OVER ADDRESS.
                                                ;INHIBIT ALL PRINT OUT??
                                                ;BR IF NO PRINT OUT WANTED (SW12=1)
                                                ;IS NUMBER MINUS? (MSB=1(BIT7))
                                                ;BR IF NUMBER IS PLUS
                                                ;TYPE A CR/LF!
                                                ;TTY READY?
                                                ;BR IF NO.
                                                ;PRINT CURRENT CHAR.
                                                ;IF NOT ZERO KEEP PRINTING!
                                                ;END OF OUTPUT. RESTORE R5
                                                ;GO HOME
                                                ;-------------------------
                                                ;SAVE R3 ON STACK
                                                ;SAVE R4 ON STACK
                                                ;GET ADDRESS OF MESSAGE.
                                                ;POP OVER ADDRESS.
                                                ;TYPE CHARACTER
                                                ;ZERO TERMINATES MESSAGE
                                                ;GET ADDRESS OF INBUF
```

```
   711  003442  012703  000007              MOV     #7,R3
   712  003446  105777  175532      1$:     TSTB    @TKCSR
   713  003452  100375                      BPL     1$
   714  003454  117714  175526              MOVB    @TKDBR,(R4)
   715  003460  142714  000200              BICB    #200,(R4)
   716  003464  122427  000015              CMPB    (R4)+,#15
   717  003470  001417                      BEQ     INSTR2
   718  003472  105777  175512      2$:     TSTB    @TPCSR
   719  003476  100375                      BPL     2$
   720  003500  017777  175502  175504      MOV     @TKDBR,@TPDBR
   721  003506  005303                      DEC     R3
   722  003510  001356                      BNE     1$
   723  003512  012604                      MOV     (SP)+,R4
   724  003514  012603                      MOV     (SP)+,R3
   725  003516  010346              .INSTE: MOV     R3,-(SP)
   726  003520  010446                      MOV     R4,-(SP)
   727  003522  104402  005060              TYPE    ,MQM
   728  003526  000741                      BR      .INST1
   729  003530  012604              INSTR2: MOV     (SP)+,R4        ;RESTORE R4
   730  003532  012603                      MOV     (SP)+,R3        ;RESTORE R3
   731  003534  000002                      RTI
   732
   733                                      ;CONVERT ASCII STRING TO OCTAL
   734                                      ;-----------------------------
   735
   736  003536  010546              .PARAM: MOV     R5,-(SP)
   737  003540  010446                      MOV     R4,-(SP)
   738  003542  016605  000004              MOV     4(SP),R5
   739  003546  012537  003726              MOV     (R5)+,LOLIM
   740  003552  012537  003730              MOV     (R5)+,HILIM
   741  003556  012537  003732              MOV     (R5)+,DEVADR
   742  003562  112537  003734              MOVB    (R5)+,LOBITS
   743  003566  112537  003735              MOVB    (R5)+,ADRCNT
   744  003572  010566  000004              MOV     R5,4(SP)
   745  003576  005005              PARAM1: CLR     R5
   746  003600  012704  006106              MOV     #INBUF,R4
   747  003604  122714  000015              CMPB    #15,(R4)
   748  003610  001420                      BEQ     PARERR
   749  003612  121427  000060      1$:     CMPB    (R4),#60
   750  003616  002415                      BLT     PARERR
   751  003620  121427  000067              CMPB    (R4),#67
   752  003624  003012                      BGT     PARERR
   753  003626  142714  000060              BICB    #60,(R4)
   754  003632  152405                      BISB    (R4)+,R5
   755  003634  122714  000015              CMPB    #15,(R4)
   756  003640  001406                      BEQ     LIMITS
   757  003642  006305                      ASL     R5
   758  003644  006305                      ASL     R5
   759  003646  006305                      ASL     R5
   760  003650  000760                      BR      1$
   761  003652  104404              PARERR: INSTER
   762  003654  000750                      BR      PARAM1
   763
   764                                      ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
   765                                      ;--------------------------------------
   766
```

```
                                LIMITS: CMP     R5,HILIM
                                        BHI     PARERR
                                        CMP     R5,LOLIM
                                        BLO     PARERR
                                        BITB    LOBITS,R5
                                        BNE     PARERR

                                        ;STORE NUMBER AT SPECIFIED ADDRESS

                                        MOV     DEVADR,R4
                                1$:     MOV     R5,(R4)+
                                        ADD     #2,R5
                                        DECB    ADRCNT
                                        BNE     1$
                                        MOV     (SP)+,R4
                                        MOV     (SP)+,R5
                                        RTI
                                LOLIM:  0
                                HILIM:  0
                                DEVADR: 0
                                LOBITS: 0
                                ADRCNT=LOBITS+1

                                        ;SAVE PC OF TEST THAT FAILED AND R0-R5
                                        ;-------------------------------------

                                .SAV05: MOV     4(SP),SAVPC     ;SAVE R7 (PC)

                                        ;SAVE R0-R5

                                SV05:   MOV     R5,SAVR5        ;SAVE R5
                                        MOV     R4,SAVR4        ;SAVE R4
                                        MOV     R3,SAVR3        ;SAVE R3
                                        MOV     R2,SAVR2        ;SAVE R2
                                        MOV     R1,SAVR1        ;SAVE R1
                                        MOV     R0,SAVR0        ;SAVE R0
                                        RTI                     ;LEAVE.

                                        ;RESTORE R0-R5

                                .RES05: MOV     SAVR0,R0        ;RESTORE R0
                                        MOV     SAVR1,R1        ;RESTORE R1
                                        MOV     SAVR2,R2        ;RESTORE R2
                                        MOV     SAVR3,R3        ;RESTORE R3
                                        MOV     SAVR4,R4        ;RESTORE R4
                                        MOV     SAVR5,R5        ;RESTORE R5
                                        RTI                     ;LEAVE


                                        ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
                                        ;-------------------------------------------------------

                                .CONVR: TYPE    ,MCRLF
                                .CNVRT: MOV     R0,-(SP)
                                        MOV     R1,-(SP)
                                        MOV     R3,-(SP)
```

```
                                                        MOV     R4,-(SP)
                                                        MOV     R5,-(SP)
                                                        MOV     @12(SP),R1
                                                        ADD     #2,12(SP)
                                                        MOV     (R1)+,WRDCNT
                                                1$:     MOVB    (R1)+,CHRCNT
                                                        MOVB    (R1)+,SPACNT
                                                        MOV     @(R1)+,BINWRD
                                                2$:     MOV     BINWRD,R4
                                                        MOVB    CHRCNT,R5
                                                        MOV     #TEMP,R0
                                                3$:     MOV     R4,R3
                                                        BIC     #177770,R3
                                                        ADD     #060,R3
                                                        MOVB    R3,(R0)+
                                                        CLC
                                                        ROR     R4
                                                        CLC
                                                        ROR     R4
                                                        CLC
                                                        ROR     R4
                                                        DEC     R5
                                                        BNE     3$
                                                        MOV     #MDATA,R3
                                                4$:     MOVB    -(R0),(R3)+
                                                        DECB    CHRCNT
                                                        BNE     4$
                                                        TSTB    SPACNT
                                                        BEQ     6$
                                                5$:     MOVB    #040,(R3)+
                                                        DECB    SPACNT
                                                        BNE     5$
                                                6$:     CLRB    (R3)
                                                        TYPE    ,MDATA
                                                        DEC     WRDCNT
                                                        BNE     1$
                                                        MOV     (SP)+,R5
                                                        MOV     (SP)+,R4
                                                        MOV     (SP)+,R3
                                                        MOV     (SP)+,R1
                                                        MOV     (SP)+,R0
                                                        RTI
                                        WRDCNT: 0
                                        CHRCNT: 0
                                        SPACNT=CHRCNT+1
                                        BINWRD: 0


                                                ;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
                                                ;BUFFER TO THE CHARACTERS "N" AND "Y".
                                                ;IF THE CHARACTER IS "N" CLEAR THE FLAG
                                                ;IF THE CHARACTER IS "Y" SET THE FLAG

                                        .SETFLG:MOV     @(SP),R5
                                                BIC     #40,INBUF
                                                CMPB    #'N,INBUF       ;IS IT "N" ?
```

```
879  004262  001002                          BNE     1$
880  004264  105015                          CLRB    (R5)    ;000
881  004266  000406                          BR      2$
882  004270  122737  000131  006106  1$:     CMPB    #'Y,INBUF       ;IS IT "Y" ?
883  004276  001005                          BNE     3$
884  004300  112715  177777                  MOVB    #-1,(R5)        ;377
885  004304  062716  000002          2$:     ADD     #2,(SP)
886  004310  000002                          RTI
887  004312  104404                  3$:     INSTER  ;RETRY
888  004314  000752                          BR      .SETFLG
889
890
891                                          ;TRAP DISPATCH SERVICE
892                                          ;ARGUMENT OF TRAP IS EXTRACTED
893                                          ;AND USED AS OFFSET TO OBTAIN POINTER
894                                          ;TO SELECTED SUBROUTINE
895
896  004316  011646                  .TRPSR: MOV     (SP),-(SP)              ;GET PC OF RETURN
897  004320  162716  000002                  SUB     #2,(SP)                 ;=PC OF TRAP
898  004324  017616  000000                  MOV     @(SP),(SP)              ;GET TRP
899  004330  006316                  TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
900  004332  042716  177001                  BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
901  004336  062716  001344                  ADD     #.TRPTAB,(SP)           ;POINTER TO SUBROUTINE ADDRESS
902  004342  017616  000000                  MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
903  004346  000136                          JMP     @(SP)+                  ;GO TO SUBROUTINE
904
905                                          ;ERROR HANDLER
906                                          ;-------------
907
908  004350  032777  010000  174624  .HLT:   BIT     #SW12,@SWR      ;BELL ON ERROR?
909  004356  001406                          BEQ     XBX             ;BR IF NO BELL
910  004360  105777  174624                  TSTB    @TPCSR          ;TTY READY.
911  004364  100003                          BPL     XBX             ;DON'T WAIT IF TTY NOT READY.
912  004366  112777  000207  174616          MOVB    #207,@TPDBR     ;PUSH A BELL AT THE TTY.
913  004374  032777  020000  174600  XBX:    BIT     #SW13,@SWR      ;DELETE ERROR PRINT OUT?
914  004402  001105                          BNE     HALTS           ;BR IF NO PRINT OUT WANTED.
915  004404  021637  001234                  CMP     (SP),LSTERR     ;WAS THIS ERROR FOUND LAST TIME?
916  004410  001404                          BEQ     1$              ;BR IF YES
917  004412  011637  001234                  MOV     (SP),LSTERR     ;RECORD BEING HERE
918  004416  105037  001341                  CLRB    ERRFLG          ;PREPARE HEADER
919  004422  104406                  1$:     SAVD5                   ;SAVE ALL PROC REGISTERS
920  004424  011605                          MOV     (SP),R5         ;GET THE PC OF ERROR
921  004426  162705  000002                  SUB     #2,R5           ;GET ADDRESS OF TRAP CALL
922  004432  011504                          MOV     (R5),R4         ;GET HLT INSTRUCTION
923  004434  006304                          ASL     R4              ;MULT BY TWO
924  004436  061504                          ADD     (R5),R4         ;DOUBLE IT
925  004440  006304                          ASL     R4              ;MULT AGAIN
926  004442  042704  177001                  BIC     #177001,R4      ;CLEAR JUNK
927  004446  062704  023222                  ADD     #.ERRTAB,R4     ;GET POINTER
928  004452  012437  004566                  MOV     (R4)+,ERRMSG    ;GET ERROR MESSAGE
929  004456  012437  004600                  MOV     (R4)+,DATAHD    ;GET DATA HEADRER
930  004462  011437  004612                  MOV     (R4),DATABP     ;GET DATA TABLE
931  004466  105737  001341                  TSTB    ERRFLG          ;TYPE HEADREER
932  004472  001403                          BEQ     TYPMSG          ;BR IF YES
933  004474  005737  004612                  TST     DATABP          ;DOES DATA TABLE EXIST?
934  004500  001040                          BNE     TYPDAT          ;BR IF YES.
```

```
   935  004502  104402  005064            TYPMSG: TYPE    ,MCRLF
   936  004506  104402  005064                    TYPE    ,MCRLF
   937  004512  005737  001220                    TST     LOCK
   938  004516  001402                            BEQ     1$
   939  004520  104402  005337                    TYPE    ,MASTEK
   940  004524  104402  005325            1$:     TYPE    ,MTSTN
   941  004530  104411  004726                    CNVRT   ,XTSTN          ;SHOW IT
   942  004534  104402  005414                    TYPE    ,MERRPC         ;TYPE PC.
   943  004540  104411  004720                    CNVRT   ,ERTABO         ;SHOW IT
   944  004544  104402  005064                    TYPE    ,MCRLF          ;GIVE A CR/LF
   945  004550  112737  177777  001341            MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
   946  004556  005737  004566                    TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
   947  004562  001402                            BEQ     WRKO.FM         ;BR IF NO.
   948  004564  104402                            TYPE                    ;TYPE
   949  004566  000000            ERRMSG: 0                               ;    ERROR MESSAGE
   950  004570                    WRKO.FM:                                ;
   951  004570  005737  004600                    TST     DATAHD          ;DATA HEADER?
   952  004574  001402                            BEQ     TYPDAT          ;BR IF NO
   953  004576  104402                            TYPE                    ;TYPE
   954  004600  000000            DATAHD: 0                               ;    DATA HEADER
   955  004602  005737  004612    TYPDAT: TST     DATABP          ;DATA TABLE?
   956  004606  001402                            BEQ     RESREG          ;BR IF NO.
   957  004610  104410                            CONVRT                  ;SHOW
   958  004612  000000            DATABP: 0                               ;    DATA TABLE
   959  004614  104407            RESREG: RESO5                           ;RESTORE PROC REGISTERS
   960  004616  022737  003104  000042  HALTS:   CMP     #$ENDAD,@#42    ;IF ACT-11 AUTO MODE--HALT!!
   961  004624  001403                            BEQ     1$
   962  004626  005777  174350                    TST     @SWR            ;HALT ON ERROR?
   963  004632  100010                            BPL     EXITER          ;BR IF NO HALT ON ERROR
   964  004634  010046            1$:     PUSHRO                          ;SAVE R0
   965  004636  016600  000002                    MOV     2(SP),R0        ;SHOW ERROR PC IN DATA LIGHTS
   966  004642  042777  014000  174542            BIC     #SYSTST!MEXT,@TXCSR
   967  004650  000000                            HALT                    ;HALT
   968  004652  012600                            POPRO                   ;GET R0
   969  004654  005237  001232    EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
   970  004660  032777  000400  174314            BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
   971  004666  001007                            BNE     1$              ;BR IF YES
   972  004670  032777  002000  174304            BIT     #SW10,@SWR      ;GOTO NEXT TEST?
   973  004676  001407                            BEQ     2$              ;BR IF NO
   974  004700  013737  001216  001214            MOV     NEXT,RETURN     ;SET FOR NEXT TEST
   975  004706  012706  001150            1$:     MOV     #STACK,SP       ;RESET SP
   976  004712  000177  174276                    JMP     @RETURN         ;GOTO SPECIFIED TEST
   977  004716  000002                    2$:     RTI                     ;RETURN
   978  004720  000001            ERTABO: 1
   979  004722      006     002                    .BYTE   6,2
   980  004724  001266                            SAVPC
   981  004726  000001            XTSTN:  1
   982  004730      003     002                    .BYTE   3,2
   983  004732  001226                            TSTNO
   984  004734  017600  000000    .PKCLK: MOV     @(SP),R0        ;GET THE # OF TICKS TO POKE
   985  004740  062716  000002                    ADD     #2,(SP)         ;POP OVER THE #
   986  004744                    1$:
   987  004744  052777  020000  174440            BIS     #CLK,@TXCSR     ;POKE CLOCK UP
   988  004752  005300                            DEC     R0              ;ARE WE DONE?
   989  004754  001405                            BEQ     2$              ;YES-GO TO 2$
   990  004756  042777  020000  174426            BIC     #CLK,@TXCSR             ;POKE CLOCK DOWN
```

```
                                                DEC     R0              ;ARE WE DONE?
                                                BNE     1$              ;NO-REPEAT
                                        2$:     RTI                     ;RETURN


                                        ;WAIT ROUTINE
                                        $MALL:  NOP                     ;STALL
                                                RTS     PC              ;RETURN


                                        ;POWER FAIL ROUTINE
005002  012737  005006  000024  .PFAIL: MOV     #PWRUP,24       ;LOAD PFAIL VECTOR FOR POWER UP
005004  000000                          HALT                    ;
005006  000005                  PWRUP:  RESET                   ;WAIT TTY TO COME UP
005010  012706  001150                  MOV     #STACK,SP       ;REINIT STACK POINTER
005014  012737  004776  000024          MOV     #.PFAIL,24      ;LOAD PFAIL VECTOR FOR POWER DOWN
005022  104402                          TYPE
005024  005067                          MPOWER
005026  000177  174162                  JMP     @RETURN
                                        ;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2,HALT

005032  012702  000300          CLRVEC: MOV     #300,R2                 ;R2 COMM VECTOR AREA ADRS
005036  012701  000302                  MOV     #302,R1                 ;INIT R1 WITH ADRS OF HALT
005042  010122                  1$:     MOV     R1,(R2)+                ;MOV .+2 TO PC
005044  005022                          CLR     (R2)+                   ;MOV HALT TO PC
005046  022121                          CMP     (R1)+,(R1)+             ;INC TO NEXT VECTOR AREA
005050  022701  000776                  CMP     #776,R1                 ;END OF VECTOR AREA
005054  001372                          BNE     1$                      ;NO
005056  000207                          RTS     PC                      ;RETURN


005060  020040  000077          MQM:    .ASCIZ  /  ?/
005064  005015     000          MCRLF:  .ASCIZ  <15><12>
005067     377  053520  020122 MPOWER: .ASCIZ <377>/PWR FAILED. /
005105     015  042777  042116 MEPASS: .ASCIZ <15><377>/END PASS DZDPDA /
005130  051377     000          MR:     .ASCIZ <377>/R/
005133     377  051120  043517 MERR2:  .ASCIZ <377>/PROGRAM INDICATES NO DEVICES PRESENT./
005202  044777  051516  043125 MERR3:  .ASCIZ <377>/INSUFFICIENT DATA!/
005226  052377  051505  020124 MTSTPC: .ASCIZ <377>/TEST PC-/
005240  046377  041517  020113 MLOCK:  .ASCIZ <377>/LOCK ON SELECTED TEST/
005267     103  051123  020072 MCSRX:  .ASCIZ  /CSR: /
005275     126  041505  020072 MVECX:  .ASCIZ /VEC: /
005303     120  051501  042523 MPASSX: .ASCIZ /PASSES: /
005314  051105  047522  051522 MERRX:  .ASCIZ /ERRORS: /
005325     124  051505  020124 MTSTN:  .ASCIZ /TEST NO: /
005337     052     000          MASTEK: .ASCIZ /*/
005341     377  042523  020124 MNEW:   .ASCIZ <377>/SET SWITCH REG TO DUP11'S DESIRED ACTIVE./
005414  041520  020072     000 MERRPC: .ASCIZ /PC: /
005421     377  042522  020103 MCSR:   .ASCIZ <377>/REC CSR ADRS /
005440  053377  041505  040440 MVEC:   .ASCIZ <377>/VEC ADRS /
005453     377  051511  052040 MJMPR:  .ASCIZ <377>/IS THE OPTIONAL CLR JMPR IN?  (Y OR N)   /
005526  044777  020123  044124 MTCN:   .ASCIZ <377>/IS THE H325 CONNECTOR ON? (Y OR N)   /
005575     377  020043  043117 MTOTAL: .ASCIZ <377>/# OF DUP'S (IN OCTAL)    /
005630  050377  044522  051117 MPAR:   .ASCIZ <377>/PRIORITY (4 TO 7) /
005654  051777  041505  052040 MSTJM:  .ASCIZ <377>/SEC TX JMPR IN? (Y OR N) /
```

```
   (2)  005707     377  042523  020103  MSRJM:  .ASCIZ  <377>/SEC RX JMPR IN? (Y OR N) /
   (2)  005742  046777  050101  047440  XHEAD:  .ASCIZ  <377>/MAP OF DUP11 STATUS/<377>
   (2)                                  .EVEN
   (2)  005770  000002                  XSTATQ: 2
1024    005772     006     003                  .BYTE   6,3
1025    005774  001236                          TEMP1
1026    005776     006     002                  .BYTE   6,2
1027    006000  001240                          TEMP2
1028                                 .EVEN
1029
1030    006002  000000                  TEMP:   0
1031            006044                  .=.+40
1032    006044  000000                  MDATA:  0
1033            006106                  .=.+40
1034    006106  000000                  INBUF:  0
1035            006150                  .=.+40
1036    006150  000001                  TRP.PC: .BLKW 1
1037
```

```
1038
1039                                                    ;
1040                                                    ;ROUTINE USED TO "CYCLE" THROUGH UP TO EIGHT DUP11'S
1041                                                    ;THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
1042                                                    ;AND RUNS THE SPECIFIED DUP11'S.   THIS ROUTINE *MUST*
1043                                                    ;BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
1044                                                    ;SETUP NECESSARY.
1045                                                    ;
1046
1047  006152  105737  001310          CYCLE:  TSTB    DUPACTV         ;ARE ANY DUP11'S TO BE TESTED?
1048  006156  001004                          BNE     1$              ;BR IF OK.
1049  006160  104402  005133                  TYPE    ,MERR2          ;NO DUP11'S SELECTED!!
1050  006164  000000                          HALT                    ;STOP THE SHOW.
1051  006166  000776                          BR      .-2             ;DISQUALIFY CONT. SW.
1052  006170  133737  001314  001310  1$:     BITB    RUN,DUPACTV     ;IS THIS ONE "ACTIVE"
1053  006176  001020                          BNE     2$              ;BR IF GOOD ONE FOUND.
1054  006200  000241                          CLC                     ;CLEAR PROC. CARRY BIT.
1055  006202  106137  001314                  ROLB    RUN             ;UPDATE POINTER
1056  006206  105537  001314                  ADCB    RUN             ;CATCH CARRY FROM RUN
1057  006212  062737  000006  001316          ADD     #6,CREAM        ;UPDATE ADDRESS POINTER.
1058  006220  022737  001560  001316          CMP     #DUP.END,CREAM
1059  006226  001360                          BNE     1$
                                                        ;KEEP GOING; NOT ALL TESTED FOR.
1060  006230  012737  001500  001316          MOV     #DUP.MAP,CREAM  ;RESET ADDRESS POINTER.
1061  006236  000754                          BR      1$              ;KEEP LOOKING FOR ACTIVE DUP11
1062  006240  000241                  2$:     CLC                     ;CLEAR PROC. CARRY.
1063  006242  106137  001314                  ROLB    RUN             ;UPDATE POINTER.
1064  006246  105537  001314                  ADCB    RUN             ;CATCH CARRY.
1065  006252  013700  001316                  MOV     CREAM,R0        ;GET ADDRESS POINTER.
1066  006256  062737  000006  001316          ADD     #6,CREAM        ;UPDATE.
1067  006264  022737  001560  001316          CMP     #DUP.END,CREAM
1068                                                                  ;ALL DONE?
1069  006272  001003                          BNE     3$              ;BR IF NO.
1070  006274  012737  001500  001316          MOV     #DUP.MAP,CREAM  ;RESTORE POINTER.
1071  006302  012037  001404          3$:     MOV     (R0)+,RXCSR     ;LOAD SYSTEM CTRL. REG
1072  006306  012037  001374                  MOV     (R0)+,DUPRVC    ;LOAD VECTOR
1073  006312  012037  001434                  MOV     (R0)+,LOO.00    ;GET PARAMETERS
1074  006316  012700  000002                  MOV     #2,R0           ;SAVE CORE THIS WAY!
1075  006322  013737  001404  001424          MOV     RXCSR,HUPRCR    ;GET CONTROL REG HIGH BYTE
1076  006330  005237  001424                  INC     HUPRCR          ;GOT IT
1077  006334  013737  001424  001406          MOV     HUPRCR,RXDBUF   ;GET RX CONTROL REG BUFFER
1078  006342  005237  001406                  INC     RXDBUF          ;GOT IT
1079  006346  013737  001406  001416          MOV     RXDBUF,DUPSEC   ;GOT SECONDARY REG SELECT REG
1080  006354  013737  001406  001410          MOV     RXDBUF,PARCSR   ;GOT PARAMETER STATUS REGISTER
1081  006362  013737  001406  001422          MOV     RXDBUF,HUPRBF   ;GET RX BUFFER HIGH BYTE
1082  006370  005237  001422                  INC     HUPRBF          ;GOT IT
1083  006374  013737  001422  001420          MOV     HUPRBF,HUPPSR   ;GOT PAR STATUS REG HIGH BYTE
1084  006402  013737  001420  001412          MOV     HUPPSR,TXCSR    ;GET TX CONTROL REGISTER
1085  006410  005237  001412                  INC     TXCSR           ;GOT IT
1086  006414  013737  001412  001430          MOV     TXCSR,HUPTCR    ;GET TX CONTROL REG HIGH BYTE
1087  006422  005237  001430                  INC     HUPTCR          ;GOT IT
1088  006426  013737  001430  001414          MOV     HUPTCR,TXDBUF   ;BET TX BUFFER
1089  006434  005237  001414                  INC     TXDBUF          ;GOT IT
1090  006440  013737  001414  001426          MOV     TXDBUF,HUPTBF   ;GET TX BUFFER HIGH BYTE
1091  006446  005237  001426                  INC     HUPTBF          ;GOT IT
1092
1093  006452  013737  001374  001376          MOV     DUPRVC,DUPRPS   ;RX VECTOR
```

```
1094  006460  060037  001376                  ADD     R0,DUPRPS       ;RX PRIORITY LEVEL
1095  006464  013737  001376  001400          MOV     DUPRPS,DUPTVC
1096  006472  060037  001400                  ADD     R0,DUPTVC       ;TX VECTOR
1097  006476  013737  001400  001402          MOV     DUPTVC,DUPTPS
1098  006504  060037  001402                  ADD     R0,DUPTPS       ;TX PRIORITY LEVEL
1099
1100
1101  006510  012700  001434                  MOV     #L00.00,R0      ;LOAD STAUS 00-00
1102  006514  012701  001432                  MOV     #MASK.A,R1      ;PREPARE MASK.
1103  006520  012702  001433                  MOV     #CLK.A,R2       ;PREPARE CLOCKS
1104  006524  004737  006670                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1105  006530  005737  000042                  TST     @#42
1106  006534  001050                          BNE     4$
1107  006536  032777  000002  172436          BIT     #SW01,@SWR      ;IF SW01=1,GET STARTING TEST #
1108  006544  001444                          BEQ     4$
1109  006546  104402  005064          7$:     TYPE    ,MCRLF
1110  006552  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1111  006554  005325                          MTSTN   ;MESSAGE
1112  006556  104405                          PARAM   ;CONVERT STRING
1113  006560  000001                          1       ;LOW LIMIT
1114  006562  001000                          1000    ;HIGH LIMIT
1115  006564  001226                          TSTNO   ;STORE AT THIS LOCATION
1116  006566     000                  .BYTE   0       ;MASK
1117  006567     001                  .BYTE   1       ;HOW MANY TIMES + 2
1118  006570  012700  007106                  MOV     #TST1,R0
1119  006574  022710  012737          5$:     CMP     #12737,(R0)
1120  006600  001017                          BNE     6$
1121  006602  023760  001226  000002          CMP     TSTNO,2(R0)
1122  006610  001013                          BNE     6$
1123  006612  022760  001226  000004          CMP     #TSTNO,4(R0)
1124  006620  001007                          BNE     6$
1125  006622  010037  001214                  MOV     R0,RETURN       ;SAVE PC
1126  006626  104402  005064                  TYPE    ,MCRLF
1127  006632  104402  005130                  TYPE    ,MR
1128  006636  000412                          BR      8$
1129  006640  005720                  6$:     TST     (R0)+
1130  006642  020027  021412                  CMP     R0,#TLAST+10
1131  006646  001352                          BNE     5$
1132  006650  104402  005060                  TYPE    ,MQM
1133  006654  000734                          BR      7$
1134
1135  006656  012737  007106  001214  4$:     MOV     #TST1,RETURN    ;PREPARE RETURN ADDRESS
1136  006664  000177  172324          8$:     JMP     @RETURN         ;GO START TESTING.
1137
1138  006670  011003                  FIX.00: MOV     (R0),R3         ;GET PARAMETERS.
1139  006672  000207                  5$:     RTS     PC              ;
```

```
                                                SETVEC: MOV     (R5)+,@DUPRVC
                                                        MOV     (R5)+,@DUPTVC
                                                        MOVB    (R5)+,@DUPRPS
                                                        MOVB    (R5)+,@DUPTPS
                                                        RTS     R5
                                                NO.ATRAP:
                                                        HLT     12
                                                        RTI

                                                NO.BTRAP:
                                                        HLT     13
                                                        RTI

                                                SIMBCC: MOV     R0,-(SP)
                                                        MOV     R1,-(SP)
                                                        MOV     R2,-(SP)
                                                        MOV     (R5)+,TEMP1
                                                        MOV     (R5)+,TEMP2
                                                        MOV     (R5)+,TEMP3
                                                1$:     CLR     BCCFBK
                                                        MOV     TEMP3,R0
                                                        ROR     TEMP2
                                                        ADC     R0
                                                        BIT     #BIT0,R0
                                                        BEQ     2$
                                                        COM     BCCFBK
                                                2$:     MOV     XPOLY,R0
                                                        COM     R0
                                                        BIC     R0,BCCFBK
                                                        CLC
                                                        ROR     TEMP3
                                                        MOV     BCCFBK,R0
                                                        MOV     TEMP3,R1
                                                        MOV     R1,R2
                                                        BIC     R1,R0
                                                        BIC     BCCFBK,R2
                                                        BIS     R2,R0
                                                        BIC     XPOLY,TEMP3
                                                        BIS     R0,TEMP3
                                                        DEC     TEMP1
                                                        BNE     1$
                                                        MOV     TEMP3,CALBCC
                                                        MOV     (SP)+,R2
                                                        MOV     (SP)+,R1
                                                        MOV     (SP)+,R0
                                                        RTS     R5
                                                XPOLY:  0
                                                BCCFBK: 0
                                                CALBCC: 0
                                                CRC16=120001
                                                CRC.CCITT=102010
```

```
                                        ;*************************** TEST 1 ****************************
                                        ;*PRIORITY INTERRUPT TEST.
                                        ;*SET PROCESSOR STATUS TO PRIORITY 7
                                        ;*AND VERIFY THAT THE DUP11 WILL NOT INTERRUPT.
                                        ;;*************************************************************

                                ;;*************
                                ;             *
                                ; TEST 1
                                ;             *
                                ;;*************
                                ;;*************************************************************************
007106  012737  000001  001226  TST1:   MOV     #1,@#TSTNO
007114  012737  007212  001216          MOV     #TST2,NEXT
007122  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
007130  052777  000400  172254          BIS     #MRESET,@TXCSR          ;RESET THE DEVICE
007136  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
007142  004537  006674                  JSR     R5,SETVEC               ;SET UP VECTORS
007146  006716                          NO.ATRAP                        ;VECTOR "A"
007150  006722                          NO.BTRAP                                ;VECTOR "B"
007152     340     340                  .BYTE   340,340                 ;LEVEL
007154  012777  004100  172230          MOV     #TXINTE!SYSTST,@TXCSR   ;TURN ON DUP TX INT. ENABLEAND ENTER SYSTST MODE
007162  012737  000340  177776          MOV     #340,PS                 ;SET CPU PRIORITY--CHANGE HERE IF NOT = 5
007170  000240                          NOP                             ;STALL
007172  000240                          NOP                             ;DITTO
007174  000240                          NOP                             ;DITTO
007176  005077  172210          1$:     CLR     @TXCSR                  ;DISABLE THE DUP11
007202  104400                          SCOPE                           ;SCOPE THIS TEST
007204  012716  007176          2$:     MOV     #1$,(SP)                ;SETUP FOR RETURN
007210  000002                          RTI                             ;RETURN


                                        ;*************************** TEST 2 ****************************
                                        ;*PRIORITY INTERRUPT TEST.
                                        ;*SET PROCESSOR STATUS TO PRIORITY 6
                                        ;*AND VERIFY THAT THE DUP11 WILL NOT INTERRUPT.
                                        ;;*************************************************************

                                ;;*************
                                ;             *
                                ; TEST 2
                                ;             *
                                ;;*************
                                ;;*************************************************************************
007212  012737  000002  001226  TST2:   MOV     #2,@#TSTNO
007220  012737  007326  001216          MOV     #TST3,NEXT
007226  122737  000005  001320          CMPB    #5,PRIRTY               ;COMPARE REAL WITH NORMAL
007234  001026                          BNE     1$                      ;BR IF NOT  A MATCH
007236  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
007244  052777  000400  172140          BIS     #MRESET,@TXCSR          ;RESET THE DEVICE
007252  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
007256  004537  006674                  JSR     R5,SETVEC               ;SET UP VECTORS
007262  006716                          NO.ATRAP                        ;VECTOR "A"
007264  006722                          NO.BTRAP                                ;VECTOR "B"
007266     340     340                  .BYTE   340,340                 ;LEVEL
007270  012777  004100  172114          MOV     #TXINTE!SYSTST,@TXCSR   ;TURN ON DUP TX INT. ENABLEAND ENTER SYSTST MODE
```

```
1252  007276  012737  000300  177776          MOV     #300,PS         ;SET CPU PRIORITY--CHANGE HERE IF NOT = 5
1253  007304  000240                          NOP                     ;STALL
1254  007306  000240                          NOP                     ;DITTO
1255  007310  000240                          NOP                     ;DITTO
1256  007312  005077  172074          1$:     CLR     @TXCSR          ;DISABLE THE DUP11
1257  007316  104400                          SCOPE                   ;SCOPE THIS TEST
1258  007320  012716  007312          2$:     MOV     #1$,(SP)        ;SETUP FOR RETURN
1259  007324  000002                          RTI                     ;RETURN
1260
1261
1262                                          ;************************** TEST 3 **************************
1263                                          ;*PRIORITY INTERRUPT TEST.
1264                                          ;*SET PROCESSOR STATUS TO PRIORITY 5
1265                                          ;*AND VERIFY THAT THE DUP11 WILL NOT INTERRUPT.
1266                                          ;;*********************************************************************
1267
1268                                  ;;*************
1269                                  ;             *
1270                                  ; TEST 3
1271                                  ;             *
1272                                  ;;*************
1273                                  ;;*********************************************************************
1274  007326  012737  000003  001226  TST3:   MOV     #3,@#TSTNO
1275  007334  012737  007442  001216          MOV     #TST4,NEXT
1276  007342  122737  000005  001320          CMPB    #5,PRIRTY       ;COMPARE REAL WITH NORMAL
1277  007350  001026                          BNE     1$              ;BR IF NOT  A MATCH
1278  007352  012737  000340  177776          MOV     #340,PS         ;LOCK OUT INTERRUPTS
1279  007360  052777  000400  172024          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1280  007366  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1281  007372  004537  006674                  JSR     R5,SETVEC       ;SET UP VECTORS
1282  007376  006716                          NO.ATRAP                ;VECTOR "A"
1283  007400  006722                          NO.BTRAP                ;VECTOR "B"
1284  007402     340     340                  .BYTE   340,340         ;LEVEL
1285  007404  012777  004100  172000          MOV     #TXINTE!SYSTST,@TXCSR   ;TURN ON DUP TX INT. ENABLEAND ENTER SYSTST MODE
1286  007412  012737  000240  177776          MOV     #240,PS         ;SET CPU PRIORITY--CHANGE HERE IF NOT = 5
1287  007420  000240                          NOP                     ;STALL
1288  007422  000240                          NOP                     ;DITTO
1289  007424  000240                          NOP                     ;DITTO
1290  007426  005077  171760          1$:     CLR     @TXCSR          ;DISABLE THE DUP11
1291  007432  104400                          SCOPE                   ;SCOPE THIS TEST
1292  007434  012716  007426          2$:     MOV     #1$,(SP)        ;SETUP FOR RETURN
1293  007440  000002                          RTI                     ;RETURN
1294
1295
1296                                          ;************************** TEST 4 **************************
1297                                          ;*PRIORITY INTERRUPT TEST.
1298                                          ;*SET PROCESSOR STATUS TO PRIORITY 4
1299                                          ;*AND VERIFY THAT THE DUP11 WILL INTERRUPT.
1300                                          ;;*********************************************************************
1301
1302                                  ;;*************
1303                                  ;             *
1304                                  ; TEST 4
1305                                  ;             *
1306                                  ;;*************
1307                                  ;;*********************************************************************
```

```
1308  007442  012737  000004  001226  TST4:   MOV     #4,@#TSTNO
1309  007450  012737  007560  001216          MOV     #TST5,NEXT
1310  007456  122737  000005  001320          CMPB    #5,PRIRTY       ;COMPARE REAL WITH NORMAL
1311  007464  001027                          BNE     1$              ;BR IF NOT  A MATCH
1312  007466  012737  000340  177776          MOV     #340,PS         ;LOCK OUT INTERRUPTS
1313  007474  052777  000400  171710          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1314  007502  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1315  007506  004537  006674                  JSR     R5,SETVEC       ;SET UP VECTORS
1316  007512  006716                          NO.ATRAP                ;VECTOR "A"
1317  007514  007552                          2$                      ;VECTOR "B"
1318  007516     340     340                  .BYTE   340,340         ;LEVEL
1319  007520  012777  004100  171664          MOV     #TXINTE!SYSTST,@TXCSR   ;TURN ON DUP TX INT. ENABLEAND ENTER SYSTST MODE
1320  007526  012737  000200  177776          MOV     #200,PS         ;SET CPU PRIORITY--CHANGE HERE IF NOT = 5
1321  007534  000240                          NOP                     ;STALL
1322  007536  000240                          NOP                     ;DITTO
1323  007540  000240                          NOP                     ;DITTO
1324  007542  104013                          HLT     13      ;DUP FAILED TO INTERRUPT-POSSBILY WRONG PRIORITY-CHANGE IF NOT 5
1325  007544  005077  171642          1$:     CLR     @TXCSR          ;DISABLE THE DUP11
1326  007550  104400                          SCOPE                   ;SCOPE THIS TEST
1327  007552  012716  007544          2$:     MOV     #1$,(SP)        ;SETUP FOR RETURN
1328  007556  000002                          RTI                     ;RETURN
1329
1330                                          ;************************** TEST 5 **************************
1331                                          ;*TEST TO PROVE THE HALF-DUPLEX FUNCTION
1332                                          ;*PROVE THAT THE RECEIVER WILL NOT RECOGNIZE
1333                                          ;*DATA IF SEND IS ASSERTED.
1334                                          ;;*********************************************************************
1335
1336                                  ;;*************
1337                                  ;             *
1338                                  ; TEST 5
1339                                  ;             *
1340                                  ;;*************
1341                                  ;;*********************************************************************
1342  007560  012737  000005  001226  TST5:   MOV     #5,@#TSTNO
1343  007566  012737  010120  001216          MOV     #TST6,NEXT
1344  007574  105737  001322                  TSTB    TCNFLG
1345  007600  001500                          BEQ     1$
1346  007602  012737  000340  177776          MOV     #340,PS         ;LOCK OUT INTERRUPTS
1347  007610  052777  000400  171574          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1348  007616  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1349  007622  052777  010010  171562          BIS     #MEXT!HDXEN,@TXCSR  ;ENTER MAINT EXT AND HALF-DUPLEX MODES
1350  007630  004537  006674                  JSR     R5,SETVEC       ;SET UP VECTORS
1351  007634  010110                          2$                      ;RECEIVER
1352  007636  006722                          NO.BTRAP                ;TRANSMITTER
1353  007640     340     340                  .BYTE   340,340         ;LEVEL
1354  007642  005037  177776                  CLR     PS              ;LOWER PROC. STATUS
1355  007646  052777  000020  171530          BIS     #RCVEN,@RXCSR   ;TURN ON RECEIVER
1356  007654  052777  000100  171522          BIS     #RINTEN,@RXCSR  ;TURN ON INT. ENABLE
1357  007662  052777  000020  171522          BIS     #SEND,@TXCSR    ;TURN ON TRANSMITTER
1358  007670  012737  000005  007720          MOV     #5,68$          ;LOAD THE NUMBER
1359  007676  032777  004000  171510  66$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1360  007704  001374                          BNE     66$             ;BR IF SET
1361  007706  032777  004000  171500  67$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1362  007714  001774                          BEQ     67$             ;BR IF CLEAR
1363  007716  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
```

```
1364  007720  000005                    68$:    5                       ;OF TIMES TO REPEAT
1365  007722  001365                            BNE     66$             ;BR IF MORE TO GO
1366  007724  105777  171462                    TSTB    @TXCSR
1367  007730  100401                            BMI     3$
1368  007732  104005                            HLT     5               ;TXDONE FAILED TO SET
1369  007734  012777  000400  171452  3$:       MOV     #TSOM,@TXDBUF   ;LOAD TX BUFFER
1370  007742  012737  000005  007772            MOV     #5,73$          ;LOAD THE NUMBER
1371  007750  032777  004000  171436  71$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1372  007756  001374                            BNE     71$             ;BR IF SET
1373  007760  032777  004000  171426  72$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1374  007766  001774                            BEQ     72$             ;BR IF CLEAR
1375  007770  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
1376  007772  000005                    73$:    5                       ;OF TIMES TO REPEAT
1377  007774  001365                            BNE     71$             ;BR IF MORE TO GO
1378  007776  105777  171410                    TSTB    @TXCSR          ;CHECK FOR DONE
1379  010002  100401                            BMI     4$              ;BR IF SET
1380  010004  104000                            HLT                     ;EXTERNAL CLOCKING STOPPED
1381                                                                    ;OR DATA WAS NOT RECEIVERD.CHECK
1382                                                                    ;EIA DATA AND CLOCK PATHS
1383  010006  005077  171402            4$:     CLR     @TXDBUF         ;LOAD A CHARACTER
1384  010012  105777  171374                    TSTB    @TXCSR          ;CHECK FOR DONE
1385  010016  100375                            BPL     .-4             ;BR IF NOT SET
1386  010020  012777  001000  171366            MOV     #TEOM,@TXDBUF   ;END THE MESSAGE
1387  010026  012737  000050  010056            MOV     #40.,78$                ;LOAD THE NUMBER
1388  010034  032777  004000  171352  76$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1389  010042  001374                            BNE     76$             ;BR IF SET
1390  010044  032777  004000  171342  77$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1391  010052  001774                            BEQ     77$             ;BR IF CLEAR
1392  010054  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
1393  010056  000050                    78$:    40.                     ;OF TIMES TO REPEAT
1394  010060  001365                            BNE     76$             ;BR IF MORE TO GO
1395  010062  012737  000340  177776  1$:       MOV     #340,PS         ;RAISE PROCESSOR STATUS
1396  010070  012706  001150                    MOV     #STACK,SP       ;RESET STACK
1397  010074  052777  000400  171310            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1398  010102  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1399  010106  104400                            SCOPE                   ;SCOPE THIS TEST
1400
1401  010110  104007                    2$:     HLT     7               ;RECEIVER INTERRUTED AND SHOULD
1402  010112  012716  010062                    MOV     #1$,(SP)        ;NOT HAVE--THIS IS HALF
1403  010116  000002                            RTI                     ;DUPLEX.
1404
1405                                            ;************************** TEST 6 ****************************
1406                                            ;*TEST OF THE DUP RUNNING A BINARY COUNT
1407                                            ;*PATTERN WITHOUT A CRC CALCULATION
1408                                            ;;*****************************************************************
1409
1410                                    ;:************
1411                                    ;            *
1412                                    ; TEST 6
1413                                    ;            *
1414                                    ;:************
1415                                    ;;*****************************************************************************
1416  010120  012737  000006  001226  TST6:   MOV     #6,@#TSTNO
1417  010126  012737  010560  001216          MOV     #TST7,NEXT
1418  010134  052777  000400  171250          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1419  010142  004737  004772                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
```

```
1420  010146  012737  000001  001236          MOV     #1,TEMP1        ;LOAD DATA
1421  010154  005037  001240                  CLR     TEMP2           ;CLEAR EXPECTED
1422  010160  012737  000340  177776          MOV     #340,PS         ;PS = 7
1423  010166  052777  004000  171216          BIS     #SYSTST,@TXCSR  ;ENTER SYSTEM TEST MODE
1424  010174  004537  006674                  JSR     R5,SETVEC       ;LOAD INTERRUPT VECTORS
1425  010200  010374                          11$                     ;RECEIVER
1426  010202  010454                          12$                     ;TRANSMITER
1427  010204     340     340                  .BYTE   340,340         ;LEVEL
1428  010206  052777  001000  171174          BIS     #CRCEN,@PARCSR          ;TURN OFF CRC
1429  010214  052777  000020  171162          BIS     #RCVEN,@RXCSR   ;TURN ON THE RECEIVER
1430  010222  052777  000100  171154          BIS     #RINTEN,@RXCSR  ;TURN ON REC INTERRUPT ENABLE
1431  010230  105777  171156          1$:     TSTB    @TXCSR          ;TEST FOR TX DONE
1432  010234  100375                          BPL     1$              ;BR IF NOT SET
1433  010236  052777  000020  171146  2$:     BIS     #SEND,@TXCSR    ;TURN ON SEND
1434  010244  012777  000400  171142          MOV     #TSOM,@TXDBUF   ;TURN ON START OF MESSAGE
1435  010252  012737  000005  010302          MOV     #5,68$          ;LOAD THE NUMBER
1436  010260  032777  004000  171126  66$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1437  010266  001374                          BNE     66$             ;BR IF SET
1438  010270  032777  004000  171116  67$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1439  010276  001774                          BEQ     67$             ;BR IF CLEAR
1440  010300  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
1441  010302  000005                  68$:    5                       ;OF TIMES TO REPEAT
1442  010304  001365                          BNE     66$             ;BR IF MORE TO GO
1443  010306  105777  171100          3$:     TSTB    @TXCSR          ;WAIT FOR DONE
1444  010312  100401                          BMI     4$              ;BR IF SET
1445  010314  104000                          HLT                     ;EXTERNAL CLOCKING STOPPED
1446  010316  005077  171072          4$:     CLR     @TXDBUF         ;PUSH OUT DATA
1447  010322  052777  000100  171062          BIS     #TXINTE,@TXCSR  ;TURN ON TRANSMITTER INT ENABLE
1448  010330  005037  177776                  CLR     PS              ;LOWER PROCESOR STATUS
1449  010334                          5$:
1450  010334  012737  000040  010364          MOV     #32.,73$                ;LOAD THE NUMBER
1451  010342  032777  004000  171044  71$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1452  010350  001374                          BNE     71$             ;BR IF SET
1453  010352  032777  004000  171034  72$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1454  010360  001774                          BEQ     72$             ;BR IF CLEAR
1455  010362  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
1456  010364  000040                  73$:    32.                     ;OF TIMES TO REPEAT
1457  010366  001365                          BNE     71$             ;BR IF MORE TO GO
1458  010370  104001                          HLT     1               ;FAILED TO INTERRUPT IN TIME
1459  010372  104400                  6$:     SCOPE                   ;SCOPE THIS TEST
1460
1461
1462                                  ;INTERRUPT SERVICE ROUTINES
1463                                  ;-------------------------
1464
1465                                  ;RECEIVER:
1466  010374  017737  171006  001324  11$:     MOV     @RXDBUF,DATA    ;GET THE REGISTER AND DATA
1467  010402  123737  001240  001324          CMPB    TEMP2,DATA      ;CHECK IT
1468  010410  001401                          BEQ     .+4             ;BR IF OK
1469  010412  104002                          HLT     2               ;COMPARISON ERROR
1470  010414  105237  001240                  INCB    TEMP2           ;COUNT UP EXPECTED
1471  010420  105737  001240                  TSTB    TEMP2           ;CHECK TO SEE IF DONE
1472  010424  001012                          BNE     7$              ;BR IF NO
1473  010426  105777  170752          10$:    TSTB    @RXCSR          ;CHECK FOR DONE
1474  010432  100375                          BPL     10$             ;BR IF NOT YET
1475  010434  032777  001000  170744          BIT     #REOM,@RXDBUF   ;CHECK FOR END OF MSG
```

```
1476  010442  001001                          BNE     .+4             ;BR IF SET
1477  010444  104003                          HLT     3               ;END OF MSG FAILED TO SET
1478  010446  012716  010372                  MOV #6$,(SP)            ;CRUNCH STACK
1479
1480  010452  000002                  7$:     RTI                     ;RETURN
1481
1482
1483                                  ;TRANSMITTER:
1484  010454  113777  001236  170732  12$:     MOVB    TEMP1,@TXDBUF   ;LOAD THE TRANSMITTER BUFFER
1485  010462  105237  001236                   INCB    TEMP1           ;UP THE COUNT
1486  010466  122737  000377  001236           CMPB    #377,TEMP1      ;ARE WE DONE
1487  010474  001026                          BNE     13$             ;BR IF NO
1488  010476  012777  010506  170674          MOV     #21$,@DUPTVC    ;SETUP FOR NEXT PART
1489  010504  000422                          BR      13$             ;LEAVE
1490  010506  012777  000377  170700  21$:     MOV     #377,@TXDBUF    ;LOAD BUFFER
1491  010514  012777  010524  170656          MOV     #22$,@DUPTVC    ;SETUP NEXT PART
1492  010522  000413                          BR      13$             ;LEAVE
1493  010524  012777  001000  170662  22$:     MOV     #TEOM,@TXDBUF   ;SET END OF MSG
1494  010532  000240                          NOP                     ;STALL
1495  010534  000240                          NOP                     ;DITTO
1496  010536  042777  000120  170646          BIC     #SEND!TXINTE,@TXCSR ;TURN OFF TRANSMITTER
1497  010544  012777  006722  170626          MOV     #NO.BTRAP,@DUPTVC ;LOAD VECTOR
1498  010552  012716  010334          13$:    MOV     #5$,(SP)        ;CRUNCH STACK
1499  010556  000002                          RTI                     ;RETURNS
1500
1501
1502                                  ;************************** TEST 7 ****************************
1503                                  ;*TEST OF THE DUP RUNNING A BINARY COUNT
1504                                  ;*PATTERN WITH A CRC CALCULATION
1505                                  ;:**********************************************************************
1506
1507                                  ;:************
1508                                  ;            *
1509                                  ; TEST 7
1510                                  ;            *
1511                                  ;:************
1512                                  ;:**********************************************************************
1513  010560  012737  000007  001226  TST7:   MOV     #7,@#TSTNO
1514  010566  012737  011360  001216          MOV     #TST10,NEXT
1515  010574  052777  000400  170610          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1516  010602  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1517  010606  005001                          CLR     R1              ;CLEAR OUT DATA
1518  010610  012737  102010  007100          MOV     #CRC.CCITT,XPOLY        ;SET UP THE POLYNOMIAL
1519  010616  012737  177777  007104          MOV     #-1,CALBCC      ;SETUP FOR THE FIRST TIME
1520  010624  013737  007104  010646  16$:     MOV     CALBCC,20$      ;ALLOW FOR THE NEXT CHARACTER
1521  010632  010137  010644                  MOV     R1,17$          ;LOAD DATA
1522  010636  004537  006726                  JSR     R5,SIMBCC       ;GO CALCULATE SOFTWARE BCC
1523  010642  000010                          B.                      ;BASED ON THESE PARAMETERS
1524  010644  000001                  17$:     .BLKW   1       ;DATA
1525  010646  000001                  20$:     .BLKW   1       ;PREVIOUS BCC
1526  010650  105201                          INCB    R1              ;INCREMENT DATA
1527  010652  001364                          BNE     16$             ;BR IF MORE TO GO
1528  010654  012737  000001  001236          MOV     #1,TEMP1        ;LOAD DATA
1529  010662  005037  001240                  CLR     TEMP2           ;CLEAR EXPECTED
1530  010666  012737  000340  177776          MOV     #340,PS         ;PS = 7
1531  010674  052777  004000  170510          BIS     #SYSTST,@TXCSR  ;ENTER SYSTEM TEST MODE
```

```
1532  010702  004537  006674                    JSR     R5,SETVEC       ;LOAD INTERRUPT VECTORS
1533  010706  011074                            11$                     ;RECEIVER
1534  010710  011142                            12$                     ;TRANSMITER
1535  010712    340     340                     .BYTE   340,340         ;LEVEL
1536  010714  052777  000020  170462            BIS     #RCVEN,@RXCSR   ;TURN ON THE RECEIVER
1537  010722  052777  000100  170454            BIS     #RINTEN,@RXCSR  ;TURN ON REC INTERRUPT ENABLE
1538  010730  105777  170456          1$:       TSTB    @TXCSR          ;TEST FOR TX DONE
1539  010734  100375                            BPL     1$              ;BR IF NOT SET
1540  010736  052777  000020  170446  2$:       BIS     #SEND,@TXCSR    ;TURN ON SEND
1541  010744  012777  000400  170442            MOV     #TSOM,@TXDBUF   ;TURN ON START OF MESSAGE
1542  010752  012737  000005  011002            MOV     #5,68$          ;LOAD THE NUMBER
1543  010760  032777  004000  170426  66$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1544  010766  001374                            BNE     66$             ;BR IF SET
1545  010770  032777  004000  170416  67$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1546  010776  001774                            BEQ     67$             ;BR IF CLEAR
1547  011000  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
1548  011002  000005                  68$:      5                       ;OF TIMES TO REPEAT
1549  011004  001365                            BNE     66$             ;BR IF MORE TO GO
1550  011006  105777  170400          3$:       TSTB    @TXCSR          ;WAIT FOR DONE
1551  011012  100401                            BMI     4$              ;BR IF SET
1552  011014  104000                            HLT                     ;EXTERNAL CLOCKING STOPPED
1553  011016  005077  170372          4$:       CLR     @TXDBUF         ;PUSH OUT DATA
1554  011022  052777  000100  170362            BIS     #TXINTE,@TXCSR  ;TURN ON TRANSMITTER INT ENABLE
1555  011030  005037  177776                    CLR     PS              ;LOWER PROCESOR STATUS
1556  011034                          5$:
1557  011034  012737  000040  011064            MOV     #32.,73$                ;LOAD THE NUMBER
1558  011042  032777  004000  170344  71$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1559  011050  001374                            BNE     71$             ;BR IF SET
1560  011052  032777  004000  170334  72$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1561  011060  001774                            BEQ     72$             ;BR IF CLEAR
1562  011062  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
1563  011064  000040                  73$:      32.                     ;OF TIMES TO REPEAT
1564  011066  001365                            BNE     71$             ;BR IF MORE TO GO
1565  011070  104001                            HLT     1               ;FAILED TO INTERRUPT IN TIME
1566  011072  104400                  6$:       SCOPE                   ;SCOPE THIS TEST
1567
1568
1569                                    ;INTERRUPT SERVICE ROUTINES
1570                                    ;-------------------------
1571
1572                                    ;RECEIVER:
1573  011074  017737  170306  001324    11$:      MOV     @RXDBUF,DATA    ;GET THE REGISTER AND DATA
1574  011102  123737  001240  001324              CMPB    TEMP2,DATA      ;CHECK IT
1575  011110  001401                              BEQ     .+4             ;BR IF OK
1576  011112  104002                              HLT     2               ;COMPARISON ERROR
1577  011114  105237  001240                      INCB    TEMP2           ;COUNT UP EXPECTED
1578  011120  105737  001240                      TSTB    TEMP2           ;CHECK TO SEE IF DONE
1579  011124  001005                              BNE     7$              ;BR IF NO
1580  011126  004537  006674                      JSR     R5,SETVEC       ;YES--RESET THE VECTORS
1581  011132  011246                              14$                     ;RECEIVER
1582  011134  011142                              12$                     ;TRANSMITTER
1583  011136    340     340                       .BYTE   340,340         ;LEVEL
1584
1585  011140  000002                  7$:       RTI                     ;RETURN
1586
1587
```

```
1588                                                    ;TRANSMITTER:
1589  011142  113777  001236  170244  12$:    MOVB    TEMP1,@TXDBUF   ;LOAD THE TRANSMITTER BUFFER
1590  011150  105237  001236                  INCB    TEMP1           ;UP THE COUNT
1591  011154  122737  000377  001236          CMPB    #377,TEMP1      ;ARE WE DONE
1592  011162  001026                          BNE     13$             ;BR IF NO
1593  011164  012777  011174  170206          MOV     #21$,@DUPTVC    ;SETUP FOR NEXT PART
1594  011172  000422                          BR      13$             ;LEAVE
1595  011174  012777  000377  170212  21$:    MOV     #377,@TXDBUF    ;LOAD BUFFER
1596  011202  012777  011212  170170          MOV     #22$,@DUPTVC    ;SETUP NEXT PART
1597  011210  000413                          BR      13$             ;LEAVE
1598  011212  012777  001000  170174  22$:    MOV     #TEOM,@TXDBUF   ;SET END OF MSG
1599  011220  000240                          NOP                     ;STALL
1600  011222  000240                          NOP                     ;DITTO
1601  011224  042777  000120  170160          BIC     #SEND!TXINTE,@TXCSR ;TURN OFF TRANSMITTER
1602  011232  012777  006722  170140          MOV     #NO.BTRAP,@DUPTVC ;LOAD VECTOR
1603  011240  012716  011034          13$:    MOV     #5$,(SP)        ;CRUNCH STACK
1604  011244  000002                          RTI                     ;RETURNS
1605
1606  011246  117737  170134  001324  14$:    MOVB    @RXDBUF,DATA    ;GET FIRST PART OF CRC
1607  011254  105777  170124                  TSTB    @RXCSR          ;WAIT FOR SECOND PART
1608  011260  100375                          BPL     .-4             ;DITTO
1609  011262  017737  170120  001242          MOV     @RXDBUF,TEMP3   ;GET THE REST OF THE CRC
1610  011270  113737  001242  001325          MOVB    TEMP3,DATA+1    ;SET UP CRC CHARACTER
1611  011276  012716  011304                  MOV     #15$,(SP)       ;SETUP FOR RETURN
1612  011302  000002                          RTI                     ;RETURN
1613  011304  012737  000340  177776  15$:    MOV     #340,PS         ;RAISE PS
1614  011312  005137  007104                  COM     CALBCC          ;INVERT BCC
1615  011316  023737  007104  001324          CMP     CALBCC,DATA     ;COMPARE SOFTWARE AND HARDWARE BCC
1616  011324  001401                          BEQ     .+4             ;BR IF OK
1617  011326  104004                          HLT     4               ;BCC COMPARISON ERROR
1618  011330  032737  010000  001242          BIT     #CRCERR,TEMP3   ;CHECK THE ERROR BIT
1619  011336  001401                          BEQ     .+4             ;BR IF NO ERROR
1620  011340  104004                          HLT     4               ;BCC ERROR--RECEIVER DOESN'T
1621                                                                  ;AGREE WITH WHAT TX SENT
1622  011342  052777  000400  170042          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1623  011350  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1624  011354  000137  011072                  JMP     6$              ;LEAVE
1625
1626                                          ;************************** TEST 10 **************************
1627                                          ;*TEST OF THE DUP RUNNING A BINARY COUNT
1628                                          ;*PATTERN WITH A CRC CALCULATION
1629                                          ;;*********************************************************************
1630
1631                                      ;;*************
1632                                      ;             *
1633                                      ; TEST 10
1634                                      ;             *
1635                                      ;;*************
1636                                      ;;*************************************************************************
1637  011360  012737  000010  001226  TST10:  MOV     #10,@#TSTNO
1638  011366  012737  012166  001216          MOV     #TST11,NEXT
1639  011374  052777  000400  170010          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1640  011402  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1641  011406  105737  001322                  TSTB    TCNFLG
1642  011412  001532                          BEQ     6$
1643  011414  005001                          CLR     R1              ;CLEAR OUT DATA
```

```
1644  011416  012737  102010  007100          MOV     #CRC.CCITT,XPOLY        ;SET UP THE POLYNOMIAL
1645  011424  012737  177777  007104          MOV     #-1,CALBCC      ;SETUP FOR THE FIRST TIME
1646  011432  013737  007104  011454  16$:     MOV     CALBCC,20$      ;ALLOW FOR THE NEXT CHARACTER
1647  011440  010137  011452                  MOV     R1,17$          ;LOAD DATA
1648  011444  004537  006726                  JSR     R5,SIMBCC       ;GO CALCULATE SOFTWARE BCC
1649  011450  000010                          8.                      ;BASED ON THESE PARAMETERS
1650  011452  000001                  17$:    .BLKW   1       ;DATA
1651  011454  000001                  20$:    .BLKW   1       ;PREVIOUS BCC
1652  011456  105201                          INCB    R1              ;INCREMENT DATA
1653  011460  001364                          BNE     16$             ;BR IF MORE TO GO
1654  011462  012737  000001  001236          MOV     #1,TEMP1        ;LOAD DATA
1655  011470  005037  001240                  CLR     TEMP2           ;CLEAR EXPECTED
1656  011474  012737  000340  177776          MOV     #340,PS         ;PS = 7
1657  011502  052777  010000  167702          BIS     #MEXT,@TXCSR
1658  011510  004537  006674                  JSR     R5,SETVEC       ;LOAD INTERRUPT VECTORS
1659  011514  011702                          11$                     ;RECEIVER
1660  011516  011750                          12$                     ;TRANSMITER
1661  011520     340     340                  .BYTE   340,340         ;LEVEL
1662  011522  052777  000020  167654          BIS     #RCVEN,@RXCSR   ;TURN ON THE RECEIVER
1663  011530  052777  000100  167646          BIS     #RINTEN,@RXCSR  ;TURN ON REC INTERRUPT ENABLE
1664  011536  105777  167650          1$:     TSTB    @TXCSR          ;TEST FOR TX DONE
1665  011542  100375                          BPL     1$              ;BR IF NOT SET
1666  011544  052777  000020  167640  2$:     BIS     #SEND,@TXCSR    ;TURN ON SEND
1667  011552  012777  000400  167634          MOV     #TSOM,@TXDBUF   ;TURN ON START OF MESSAGE
1668  011560  012737  000005  011610          MOV     #5,68$          ;LOAD THE NUMBER
1669  011566  032777  004000  167620  66$:     BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1670  011574  001374                          BNE     66$             ;BR IF SET
1671  011576  032777  004000  167610  67$:     BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1672  011604  001774                          BEQ     67$             ;BR IF CLEAR
1673  011606  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
1674  011610  000005                  68$:    5                       ;OF TIMES TO REPEAT
1675  011612  001365                          BNE     66$             ;BR IF MORE TO GO
1676  011614  105777  167572          3$:     TSTB    @TXCSR          ;WAIT FOR DONE
1677  011620  100401                          BMI     4$              ;BR IF SET
1678  011622  104000                          HLT                     ;EXTERNAL CLOCKING STOPPED
1679  011624  005077  167564          4$:     CLR     @TXDBUF         ;PUSH OUT DATA
1680  011630  052777  000100  167554          BIS     #TXINTE,@TXCSR  ;TURN ON TRANSMITTER INT ENABLE
1681  011636  005037  177776                  CLR     PS              ;LOWER PROCESOR STATUS
1682  011642                          5$:
1683  011642  012737  000040  011672          MOV     #32.,73$                ;LOAD THE NUMBER
1684  011650  032777  004000  167536  71$:     BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1685  011656  001374                          BNE     71$             ;BR IF SET
1686  011660  032777  004000  167526  72$:     BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1687  011666  001774                          BEQ     72$             ;BR IF CLEAR
1688  011670  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
1689  011672  000040                  73$:    32.                     ;OF TIMES TO REPEAT
1690  011674  001365                          BNE     71$             ;BR IF MORE TO GO
1691  011676  104001                          HLT     1               ;FAILED TO INTERRUPT IN TIME
1692  011700  104400                  6$:     SCOPE                   ;SCOPE THIS TEST
1693
1694
1695                                  ;INTERRUPT SERVICE ROUTINES
1696                                  ;-------------------------
1697
1698                                  ;RECEIVER:
1699  011702  017737  167500  001324  11$:    MOV     @RXDBUF,DATA    ;GET THE REGISTER AND DATA
```

```
 1700  011710  123737  001240  001324          CMPB    TEMP2,DATA      ;CHECK IT
 1701  011716  001401                          BEQ     .+4             ;BR IF OK
 1702  011720  104002                          HLT     2               ;COMPARISON ERROR
 1703  011722  105237  001240                  INCB    TEMP2           ;COUNT UP EXPECTED
 1704  011726  105737  001240                  TSTB    TEMP2           ;CHECK TO SEE IF DONE
 1705  011732  001005                          BNE     7$              ;BR IF NO
 1706  011734  004537  006674                  JSR     R5,SETVEC       ;YES--RESET THE VECTORS
 1707  011740  012054                          14$                     ;RECEIVER
 1708  011742  011750                          12$                     ;TRANSMITTER
 1709  011744     340     340                  .BYTE   340,340         ;LEVEL
 1710
 1711  011746  000002                  7$:     RTI                     ;RETURN
 1712
 1713
 1714                                          ;TRANSMITTER:
 1715  011750  113777  001236  167436  12$:     MOVB    TEMP1,@TXDBUF   ;LOAD THE TRANSMITTER BUFFER
 1716  011756  105237  001236                  INCB    TEMP1           ;UP THE COUNT
 1717  011762  122737  000377  001236          CMPB    #377,TEMP1      ;ARE WE DONE
 1718  011770  001026                          BNE     13$             ;BR IF NO
 1719  011772  012777  012002  167400          MOV     #21$,@DUPTVC    ;SETUP FOR NEXT PART
 1720  012000  000422                          BR      13$             ;LEAVE
 1721  012002  012777  000377  167404  21$:     MOV     #377,@TXDBUF    ;LOAD BUFFER
 1722  012010  012777  012020  167362          MOV     #22$,@DUPTVC    ;SETUP NEXT PART
 1723  012016  000413                          BR      13$             ;LEAVE
 1724  012020  012777  001000  167366  22$:     MOV     #TEOM,@TXDBUF   ;SET END OF MSG
 1725  012026  000240                          NOP                     ;STALL
 1726  012030  000240                          NOP                     ;DITTO
 1727  012032  042777  000120  167352          BIC     #SEND!TXINTE,@TXCSR ;TURN OFF TRANSMITTER
 1728  012040  012777  006722  167332          MOV     #NO.BTRAP,@DUPTVC ;LOAD VECTOR
 1729  012046  012716  011642          13$:    MOV     #5$,(SP)        ;CRUNCH STACK
 1730  012052  000002                          RTI                     ;RETURNS
 1731
 1732  012054  117737  167326  001324  14$:     MOVB    @RXDBUF,DATA    ;GET FIRST PART OF CRC
 1733  012062  105777  167316                  TSTB    @RXCSR          ;WAIT FOR SECOND PART
 1734  012066  100375                          BPL     .-4             ;DITTO
 1735  012070  017737  167312  001242          MOV     @RXDBUF,TEMP3   ;GET THE REST OF THE CRC
 1736  012076  113737  001242  001325          MOVB    TEMP3,DATA+1    ;SET UP CRC CHARACTER
 1737  012104  012716  012112                  MOV     #15$,(SP)       ;SETUP FOR RETURN
 1738  012110  000002                          RTI                     ;RETURN
 1739  012112  012737  000340  177776  15$:     MOV     #340,PS         ;RAISE PS
 1740  012120  005137  007104                  COM     CALBCC          ;INVERT BCC
 1741  012124  023737  007104  001324          CMP     CALBCC,DATA     ;COMPARE SOFTWARE AND HARDWARE BCC
 1742  012132  001401                          BEQ     .+4             ;BR IF OK
 1743  012134  104004                          HLT     4               ;BCC COMPARISON ERROR
 1744  012136  032737  010000  001242          BIT     #CRCERR,TEMP3   ;CHECK THE ERROR BIT
 1745  012144  001401                          BEQ     .+4             ;BR IF NO ERROR
 1746  012146  104004                          HLT     4               ;BCC ERROR--RECEIVER DOESN'T
 1747                                                                  ;AGREE WITH WHAT TX SENT
 1748  012150  052777  000400  167234          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
 1749  012156  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
 1750  012162  000137  011700                  JMP     6$              ;LEAVE
 1751
 1752                                          ;************************* TEST 11 ****************************
 1753                                          ;*THIS TEST WILL CHECK FOR ABORT SEQUENCE
 1754                                          ;*OF THE DUP IN A DATA STREAM
 1755                                          ;;**********************************************************
```

```
                                        ;:************
                                        ;            *
                                        ; TEST 11
                                        ;            *
                                        ;:************
                                        ;:*****************************************************************************
012166  012737  000011  001226  TST11:  MOV     #11,@#TSTNO
012174  012737  012502  001216          MOV     #TST12,NEXT
012202  052777  000400  167202          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
012210  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
012214  004537  006674                  JSR     R5,SETVEC       ;SET UP INTERRUPT VECTORS
012220  012400                          4$                      ;BASED ON THESE
012222  006722                          NO.BTRAP                ;PARAMETERS
012224      340     340                 .BYTE   340,340         ;LEVEL
012226  005000                          CLR     R0              ;CLEAR
012230  005003                          CLR     R3              ;DITTO
012232  012737  000340  177776          MOV     #340,PS         ;PS=7
012240  052777  010377  167142          BIS     #PRISEC!377,@PARCSR ;LOAD SEC STATION AND ADRS
012246  052777  000120  167130          BIS     #RCVEN!RINTEN,@RXCSR ;TURN ON THE RECEIVER
012254  052777  004020  167130          BIS     #SEND!SYSTST,@TXCSR     ;TURN ON TRANSMITTER
012262  005037  177776                  CLR     PS
012266  105777  167120          1$:     TSTB    @TXCSR          ;CHECK FOR TXDONE
012272  100375                          BPL     1$              ;BR IF NOT SET
012274  052777  000400  167112          BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
012302  105777  167104          2$:     TSTB    @TXCSR          ;WAIT FOR DONE
012306  100375                          BPL     2$              ;AND THEN
012310  012777  000377  167076  3$:     MOV     #377,@TXDBUF    ;LOAD A CHARACTER
012316  005200                          INC     R0              ;UPDATE CHARACTER COUNTER
012320  022700  000005                  CMP     #5,R0           ;ARE ALL CHARACTERS LOADED?
012324  001366                          BNE     2$              ;BR IF NO
012326  052777  002000  167060          BIS     #TABORT,@TXDBUF ;TURN ON ABORT
012334  012737  000310  012364          MOV     #200.,68$               ;LOAD THE NUMBER
012342  032777  004000  167044  66$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
012350  001374                          BNE     66$             ;BR IF SET
012352  032777  004000  167034  67$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
012360  001774                          BEQ     67$             ;BR IF CLEAR
012362  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
012364  000310                  68$:    200.                    ;OF TIMES TO REPEAT
012366  001365                          BNE     66$             ;BR IF MORE TO GO
012370  104001                          HLT     1               ;RECEIVER DID NOT INTERRUPT IN TIME
012372  012706  001150          11$:    MOV     #STACK,SP       ;RESET STACK
012376  104400                          SCOPE                   ;SCOPE THIS TEST
                                        ;RECEIVER INTERRUPT SERVICE ROUTINE
012400  017701  167000          4$:     MOV     @RXCSR,R1       ;GET THE CONTROL REGISTER
012404  017702  166776                  MOV     @RXDBUF,R2      ;GET THE BUFFER
012410  032701  000200                  BIT     #RXDONE,R1      ;CHECK FOR DONE
012414  001001                          BNE     5$              ;BR IF DONE SET
012416  104007                          HLT     7               ;FALSE INTERRUPT
012420  122702  000377          5$:     CMPB    #377,R2         ;CHECK DATA CHARACTER
012424  001401                          BEQ     6$              ;BR IF A MATCH
012426  104002                          HLT     2               ;DATA ERROR
012430  005203                  6$:     INC     R3              ;INC THE # OF CHARS TO DO
012432  022703  000003                  CMP     #3,R3           ;CHECK TO SEE IF DONE
012436  001020                          BNE     10$             ;BR IF MORE TO GO
012440  105777  166740          12$:    TSTB    @RXCSR          ;CHECK FOR
012444  100375                          BPL     12$             ;DONE
```

```
                                MOV     @RXDBUF,R2      ;READ THE BUFFER
                                BIT     #RABORT,R2      ;TEST ABORT
                                BNE     7$              ;BR IF SET
                                HLT     10              ;FAILED TO RECEIVE ABORT
                        7$:     MOV     #11$,(SP)       ;SET UP FOR RETURN
                                BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
                                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
                        10$:    RTI                     ;RETURN


                                ;************************** TEST 12 ****************************
                                ;*THIS TEST PROVES THE RECEIVER WILL STOP
                                ;*ACCEPTING DATA IF SHUT OFF IN THE MIDDLE
                                ;*OF A MESSAGE, AND THAT IT WILL NOT
                                ;*RESTART UNTIL IT RECEIVES A FLAG
                                ;*****************************************************************

                        ;:************
                        ;               *
                        ; TEST 12
                        ;               *
                        ;:************
                        ;:*****************************************************************************
                        TST12:  MOV     #12,@#TSTNO
                                MOV     #TST13,NEXT
                                BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
                                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
                                CLR     R0              ;CLEAR FOR SOFTWARE
                                JSR     R5,SETVEC       ;SET UP THE VECTORS
                                3$                      ;RECEIVER
                                14$                     ;TRANSMITTER
                                .BYTE   340,340         ;LEVEL
                                MOV     #340,PS         ;PROC STATUS=7
                                BIS     #CRCEN,@PARCSR
                                BIS     #RCVEN!RINTEN,@RXCSR ;TURN ON RECEIVER
                                BIS     #SEND!SYSTST,@TXCSR     ;START TRANSMITTER
                                CLR     PS              ;LOWER PS
                        1$:     TSTB    @TXCSR          ;CHECK FOR DONE
                                BPL     1$              ;BR IF NOT YET
                                BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
                                BIS     #TXINTE,@TXCSR  ;TURN ON INT. ENABLE
                                MOV     #500.,68$               ;LOAD THE NUMBER
                        66$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
                                BNE     66$             ;BR IF SET
                        67$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
                                BEQ     67$             ;BR IF CLEAR
                                DEC     (PC)+           ;DECREMENT THE NUMBER
                        68$:    500.                    ;OF TIMES TO REPEAT
                                BNE     66$             ;BR IF MORE TO GO
                                HLT     1               ;DEVICE FAILED TO INTERRUPT IN TIME
                        2$:     MOV     #STACK,SP       ;RESET THE STACK
                                SCOPE                   ;SCOPE THIS TEST
                        20$:    JSR     R5,SETVEC       ;SET UP VECTORS
                                23$                     ;RECEIVER
                                NO.BTRAP                ;TRANSMITTER
                                .BYTE   340,340         ;LEVEL
```

```
1868  012700  052777  000020  166476          BIS     #RCVEN,@RXCSR
1869  012706  105777  166500          21$:    TSTB    @TXCSR          ;TEST DONE
1870  012712  100375                  BPL     21$             ;BR IF NOT SET
1871  012714  012777  000070  166472          MOV     #70,@TXDBUF     ;PUSH OUT DATA CHARACTER
1872  012722  012737  000062  012752          MOV     #50.,73$                ;LOAD THE NUMBER
1873  012730  032777  004000  166456  71$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
1874  012736  001374                  BNE     71$             ;BR IF SET
1875  012740  032777  004000  166446  72$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
1876  012746  001774                  BEQ     72$             ;BR IF CLEAR
1877  012750  005327                  DEC     (PC)+           ;DECREMENT THE NUMBER
1878  012752  000062          73$:    50.                     ;OF TIMES TO REPEAT
1879  012754  001365                  BNE     71$             ;BR IF MORE TO GO
1880  012756  104001                  HLT     1               ;FAILED TO INTERRUPT IN TIME
1881  012760  000737                  BR      2$              ;FINISH
1882                          ;INTERRUPT SVC ROUTINES
1883
1884                          ;RECEIVER
1885  012762  017704  166416  3$:     MOV     @RXCSR,R4       ;GET THE CONTROL REGISTER
1886  012766  017705  166414          MOV     @RXDBUF,R5      ;GET THE BUFFER
1887  012772  032705  000400          BIT     #RSOM,R5        ;CHECK FOR START OF MSG
1888  012776  001001                  BNE     4$              ;BR IF SET
1889  013000  104011                  HLT     11              ;FAILED TO RECEIVE SOM
1890  013002  032704  000200  4$:     BIT     #RXDONE,R4      ;CHECK FOR DONE
1891  013006  001001                  BNE     5$              ;BR IF SET
1892  013010  104007                  HLT     7               ;FALSE INTERRUPT
1893  013012  122705  000377  5$:     CMPB    #377,R5         ;CHECK DATA
1894  013016  001401                  BEQ     6$              ;BR IF A MATCH
1895  013020  104002                  HLT     2               ;DATA ERROR
1896  013022  012777  013032  166344  6$:     MOV     #10$,@DUPRVC    ;RELOAD THE VECTOR
1897  013030  000002          7$:     RTI                     ;RETURN
1898  013032  017705  166350  10$:    MOV     @RXDBUF,R5      ;GET THE BUFFER
1899  013036  122705  000377          CMPB    #377,R5         ;CHECK THE CHARACTER
1900  013042  001401                  BEQ     11$             ;BR IF A MATCH
1901  013044  104002                  HLT     2               ;DATA ERROR
1902  013046  042777  000020  166330  11$:    BIC     #RCVEN,@RXCSR   ;TURN OFF THE RECEIVER
1903  013054  012777  013064  166312          MOV     #12$,@DUPRVC    ;RELOAD THE VECTOR
1904  013062  000762                  BR      7$              ;RETURN
1905  013064  017704  166314  12$:    MOV     @RXCSR,R4       ;GET THE CONTROL REGISTER
1906  013070  012705  001406          MOV     #RXDBUF,R5      ;GET THE BUFFER
1907  013074  122705  000252          CMPB    #252,R5         ;CHECK THE CHARACTER
1908  013100  001402                  BEQ     13$             ;BR IF A MATCH
1909  013102  104007                  HLT     7               ;FALSE INTERRUPT
1910  013104  000751                  BR      7$
1911  013106  104007          13$:    HLT     7               ;DEVICE INTERRUPTED AFTER RX ENABLE
1912  013110  000747                  BR      7$              ;WAS CLEARED
1913  013112  017704  166266  23$:    MOV     @RXCSR,R4       ;GET THE CONTROL REG
1914  013116  017705  166264          MOV     @RXDBUF,R5      ;GET THE BUFFER
1915  013122  032715  000400          BIT     #RSOM,(R5)      ;CHECK START OF MSG
1916  013126  001001                  BNE     24$             ;BR IF SET
1917  013130  104011                  HLT     11              ;SOM FAILED TO SET
1918  013132  122705  000070  24$:    CMPB    #70,R5          ;CHECK DATA
1919  013136  001401                  BEQ     25$             ;BR IF A MATCH
1920  013140  104002                  HLT     2               ;DATA FAILED TO MATCH AFTER
1921                                                          ;RESTARTING RECEIVER
1922  013142          25$:
1923  013142  052777  000400  166242          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
```

```
1924  013150  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1925  013154  012716  012660                    MOV     #2$,(SP)        ;CRUNCH STACK
1926  013160  000002                            RTI                     ;RETURN
1927                                    ;TRANSMITTER
1928  013162  105777  166224             14$:   TSTB    @TXCSR          ;CHECK DONE
1929  013166  100401                            BMI     30$             ;BR IF SET
1930  013170  104007                            HLT     7               ;FALSE INTERRUPT
1931  013172  012777  000377  166214     30$:   MOV     #377,@TXDBUF    ;LOAD A CHARACTER
1932  013200  005200                            INC     R0              ;IN THE # TO DO
1933  013202  022700  000002                    CMP     #2,R0           ;CHECK TO SEE IF ALL ARE SENT
1934  013206  001030                            BNE     15$             ;BR IF MORE TO GO
1935  013210  012777  013222  166162            MOV     #16$,@DUPTVC    ;RELOAD THE VECT
1936  013216  005000                            CLR     R0              ;CLEAR CHAR COUNT
1937  013220  000423                            BR      15$
1938  013222  105777  166164             16$:   TSTB    @TXCSR          ;TEST DONE
1939  013226  100401                            BMI     17$             ;BR IF SET
1940  013230  104007                            HLT     7               ;FALSE INTERRUPT
1941  013232  012777  000252  166154     17$:   MOV     #252,@TXDBUF    ;LOAD A DATA CHARACTER
1942  013240  005200                            INC     R0              ;INC THE # TO DO
1943  013242  022700  000003                    CMP     #3,R0           ;CHECK FOR ALL DONE
1944  013246  001010                            BNE     15$             ;BR IF MORE TO GO
1945  013250  012777  001400  166136            MOV     #TEOM!TSOM,@TXDBUF      ;END MSG
1946  013256  042777  000100  166126            BIC     #TXINTE,@TXCSR
1947  013264  012716  012666                    MOV     #20$,(SP)       ;CRUNCH STACK
1948  013270  000002                     15$:   RTI
1949
1950                                    ;************************** TEST 13 **************************
1951                                    ;*THIS TEST WILL TRANSMIT CONTIGUOUS ONES CHARACTERS
1952                                    ;*IN SECONDARY MODE WITH A BCC CHECK.
1953                                    ;:************************************************************
1954                                    ;:************
1955                                    ;       *
1956                                    ; TEST 13
1957                                    ;       *
1958                                    ;:************
1959                                    ;:************************************************************
1960  013272  012737  000013  001226    TST13:  MOV     #13,@#TSTNO
1961  013300  012737  013732  001216            MOV     #TST14,NEXT
1962  013306  052777  000400  166076            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
1963  013314  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
1964  013320  012737  000340  177776            MOV     #340,PS         ;SET STATUS=7
1965  013326  005000                            CLR     R0
1966  013330  005002                            CLR     R2              ;SETUP FOR SOFTWARE
1967  013332  012701  000377                    MOV     #377,R1         ;CALCULATION OF BCC
1968  013336  012737  102010  007100            MOV     #CRC.CCITT,XPOLY ;LOAD THE POLYNOMINAL
1969  013344  012737  177777  007104            MOV     #-1,CALBCC      ;SETUP FOR FIRST TIME
1970  013352  013737  007104  013374     1$:    MOV     CALBCC,3$       ;ALLOW FOR THE NEXT CHARACTER
1971  013360  010137  013372                    MOV     R1,2$           ;LOAD DATA
1972  013364  004537  006726                    JSR     R5,SIMBCC       ;GO CALCULATE SOFTWARE BCC
1973  013370  000010                            B.                      ;BASED ON THOSE PARAMETERS
1974  013372  000001                     2$:    .BLKW   1               ;DATA
1975  013374  000001                     3$:    .BLKW   1               ;PREVIOUS BCC
1976  013376  005200                            INC     R0              ;INC THE # OF CHARS TO DO
1977  013400  022700  000005                    CMP     #5,R0           ;ARE WE DONE?
1978  013404  001362                            BNE     1$              ;BR IF NO
1979  013406  005000                            CLR     R0              ;CLEAR OUT HOLD
```

```
 1980  013410  004537  006674                    JSR     R5,SETVEC       ;LOAD INTERRUPT VECTORS
 1981  013414  013544                            6$                      ;RECEIVER
 1982  013416  013650                            11$                     ;TRANSMITTER
 1983  013420      340      340                  .BYTE   340,340         ;LEVEL
 1984  013422  052777  010377  165760            BIS     #PRISEC!377,@PARCSR     ;ENTER SECONDARY MODE
 1985  013430  052777  000120  165746            BIS     #RCVEN!RINTEN,@RXCSR ;TURN ON RECEIVER AND INTERRUPTS
 1986  013436  052777  004020  165746            BIS     #SEND!SYSTST,@TXCSR     ;TURN ON TRANSMITTER
 1987  013444  105777  165742          20$:      TSTB    @TXCSR
 1988  013450  100375                            BPL     20$
 1989  013452  012777  000400  165734            MOV     #TSOM,@TXDBUF   ;START MESSAGE
 1990  013460  052777  000100  165724            BIS     #TXINTE,@TXCSR  ;TURN ON INTERRUPT ENABLE
 1991  013466  005037  177776                    CLR     PS              ;LOWER PS
 1992  013472                          4$:
 1993  013472  012737  000040  013522            MOV     #32.,68$                ;LOAD THE NUMBER
 1994  013500  032777  004000  165706  66$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
 1995  013506  001374                            BNE     66$             ;BR IF SET
 1996  013510  032777  004000  165676  67$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
 1997  013516  001774                            BEQ     67$             ;BR IF CLEAR
 1998  013520  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
 1999  013522  000040                  68$:      32.                     ;OF TIMES TO REPEAT
 2000  013524  001365                            BNE     66$             ;BR IF MORE TO GO
 2001  013526  104001                            HLT     1               ;FAILED TO INTERRUPT IN TIME
 2002  013530                          5$:
 2003  013530  052777  000400  165654            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
 2004  013536  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
 2005  013542  104400                            SCOPE                   ;SCOPE THIS TEST
 2006
 2007                                  ;INTERRUPT SERVICE ROUTINES.
 2008
 2009                                  ;RECEIVER
 2010  013544  017737  165636  001324  6$:       MOV     @RXDBUF,DATA    ;GET THE DATA
 2011  013552  120137  001324                    CMPB    R1,DATA         ;CHECK IT
 2012  013556  001401                            BEQ     .+4             ;BR IF A MATCH
 2013  013560  104002                            HLT     2               ;DATA ERROR
 2014  013562  005200                            INC     R0              ;UPDATE THE # OF CHARS TO DO
 2015  013564  022700  000004                    CMP     #4,R0           ;CHECK FOR ALL CHARS DONE
 2016  013570  001003                            BNE     7$              ;BR IF MORE TO GO
 2017  013572  012777  013602  165574            MOV     #10$,@DUPRVC    ;SETUP TO GET BCC
 2018  013600  000002                  7$:       RTI                     ;RETURN
 2019
 2020  013602  117737  165600  001324  10$:      MOVB    @RXDBUF,DATA    ;GET THE FIRST HALF OF BCC
 2021  013610  105777  165570                    TSTB    @RXCSR          ;WAIT FOR
 2022  013614  100375                            BPL     .-4             ;THE SECOND HALF
 2023  013616  117737  165564  001325            MOVB    @RXDBUF,DATA+1  ;GET THE SECOND HALF
 2024  013624  005137  007104                    COM     CALBCC          ;INVERT BCC
 2025  013630  023737  007104  001324            CMP     CALBCC,DATA     ;CHECK IT
 2026  013636  001401                            BEQ     .+4             ;BR IF OK
 2027  013640  104004                            HLT     4               ;BCC COMPARE ERROR
 2028  013642  012716  013530                    MOV     #5$,(SP)        ;FINISH TEST
 2029  013646  000002                            RTI                     ;RETURN
 2030
 2031                                  ;TRANSMITTER
 2032  013650  012777  000377  165536  11$:      MOV     #377,@TXDBUF    ;LOAD A DATA CHARACTER
 2033  013656  005202                            INC     R2              ;INC THE # OF CHARS TO DO
 2034  013660  022702  000005                    CMP     #5,R2           ;CHECK TO SEE OF DONE
 2035  013664  001017                            BNE     13$             ;BR IF MORE TO GO
```

```
2036  013666  012777  013676  165504          MOV     #12$,@DUPTVC    ;SETUP NEXT VECTOR
2037  013674  000413                          BR      13$             ;RETURN
2038  013676  012777  001000  165510  12$:    MOV     #TEOM,@TXDBUF   ;END MSG
2039  013704  000240                          NOP                     ;WAIT
2040  013706  000240                          NOP                     ;DITTO
2041  013710  042777  000120  165474          BIC     #SEND!TXINTE,@TXCSR ;TURN OFF TRANSMITTER
2042  013716  012777  006722  165454          MOV     #NO.BTRAP,@DUPTVC ;RESET THE VECTOR
2043  013724  012716  013472          13$:    MOV     #4$,(SP)        ;GO BACK TO WAIT LOOP
2044  013730  000002                          RTI                     ;RETURN
2045
2046                                          ;************************** TEST 14 **************************
2047                                          ;*THIS TEST PROVES THE INTERACTION OF DEC MODE,
2048                                          ;*TSOM, SYNC, TXACT, TXDONE
2049                                          ;;*********************************************************************
2050
2051                                  ;:************
2052                                  ;            *
2053                                  ; TEST 14    *
2054                                  ;            *
2055                                  ;:************
2056                                  ;;*********************************************************************
2057  013732  012737  000014  001226  TST14:  MOV     #14,@#TSTNO
2058  013740  012737  014104  001216          MOV     #TST15,NEXT
2059  013746  052777  000400  165436          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2060  013754  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2061  013760  012777  101026  165422          MOV     #DECMOD!26!CRCEN,@PARCSR
2062  013766  052777  004000  165416          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2063  013774  052777  000020  165410          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2064  014002  012777  000426  165404          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2065  014010  012737  000005  014040          MOV     #5,68$          ;LOAD THE NUMBER
2066  014016  032777  004000  165370  66$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2067  014024  001374                          BNE     66$             ;BR IF SET
2068  014026  032777  004000  165360  67$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2069  014034  001774                          BEQ     67$             ;BR IF CLEAR
2070  014036  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
2071  014040  000005                  68$:    5                       ;OF TIMES TO REPEAT
2072  014042  001365                          BNE     66$             ;BR IF MORE TO GO
2073  014044  017704  165342                  MOV     @TXCSR,R4               ;GET THE CSR
2074  014050  032704  000200                  BIT     #TXDONE,R4              ;CHECK TRANSMITTER DONE
2075  014054  001001                          BNE     1$                      ;BR IF SET
2076  014056  104016                          HLT     16                      ;TXDONE FAILED TO SET
2077  014060  032704  001000          1$:     BIT     #TXACT,R4               ;TEST ACTIVE
2078  014064  001001                          BNE     2$                      ;BR IF SET
2079  014066  104017                          HLT     17                      ;ACTIVE FAILED TO SET
2080  014070                          2$:
2081  014070  052777  000400  165314          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2082  014076  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2083  014102  104400                          SCOPE                   ;SCOPE THIS TEST
2084
2085                                          ;************************** TEST 15 **************************
2086                                          ;*THIS TEST PROVES THE INTERACTION OF TEOM,
2087                                          ;*SEND, TXACT AND TXDONE IN DEC MODE.
2088                                          ;;*********************************************************************
2089
2090
2091                                  ;:************
```

```
2092                                    ;              *
2093                                    ; TEST 15
2094                                    ;              *
2095                                    ;:*************
2096                                    ;:*************************************************************************
2097  014104  012737  000015  001226  TST15:  MOV     #15,@#TSTNO
2098  014112  012737  014300  001216           MOV     #TST16,NEXT
2099  014120  052777  000400  165264           BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2100  014126  004737  004772                   JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2101  014132  012777  101026  165250           MOV     #DECMOD!26!CRCEN,@PARCSR
2102  014140  052777  004000  165244           BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2103  014146  052777  000020  165236           BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2104  014154  012777  000426  165232           MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2105  014162  105777  165224          1$:      TSTB    @TXCSR                  ;CHECK FOR DONE
2106  014166  100375                           BPL     1$                      ;BR IF NOT YET
2107  014170  012777  000426  165216           MOV     #TSOM!26,@TXDBUF        ;LOAD A SECOND SYNC
2108  014176  105777  165210          2$:      TSTB    @TXCSR                  ;AND NOW WAIT
2109  014202  100375                           BPL     2$                      ;FOR DONE AGAIN
2110  014204  012777  001000  165202           MOV     #TEOM,@TXDBUF           ;SET END OF MSG
2111  014212  042777  000020  165172           BIC     #SEND,@TXCSR            ;TURN OFF TRANSMITTER
2112  014220  012737  000025  014250           MOV     #25,68$         ;LOAD THE NUMBER
2113  014226  032777  004000  165160  66$:     BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2114  014234  001374                           BNE     66$             ;BR IF SET
2115  014236  032777  004000  165150  67$:     BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2116  014244  001774                           BEQ     67$             ;BR IF CLEAR
2117  014246  005327                           DEC     (PC)+           ;DECREMENT THE NUMBER
2118  014250  000025                  68$:     25                      ;OF TIMES TO REPEAT
2119  014252  001365                           BNE     66$             ;BR IF MORE TO GO
2120  014254  105777  165132                   TSTB    @TXCSR                  ;CHECK DONE
2121  014260  100401                           BMI     3$                      ;BR IF SET
2122  014262  104016                           HLT     16                      ;DONE FAILED TO SET AFTER TURNING OFF TX.
2123  014264  032777  001000  165120  3$:      BIT     #TXACT,@TXCSR           ;CHECK ACTIVE
2124  014272  001401                           BEQ     4$                      ;BR IF OFF
2125  014274  104020                           HLT     20                      ;ACTIVE IS STILL SET-SHOULD BE RESET
2126  014276  104400                  4$:      SCOPE                           ;SCOPE FOR THIS TEST.
2127
2128
2129                                                    ;*************************** TEST 16 ****************************
2130                                                    ;*THIS TEST PROVES THAT THE DUP WILL NOT
2131                                                    ;*SYNC UP IN LESS THAN TWO SYNCS
2132                                                    ;:************************************************************
2133
2134                                    ;:*************
2135                                    ;              *
2136                                    ; TEST 16
2137                                    ;              *
2138                                    ;:*************
2139                                    ;:*************************************************************************
2140  014300  012737  000016  001226  TST16:  MOV     #16,@#TSTNO
2141  014306  012737  014646  001216           MOV     #TST17,NEXT
2142  014314  052777  000400  165070           BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2143  014322  004737  004772                   JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2144  014326  012777  101026  165054           MOV     #DECMOD!26!CRCEN,@PARCSR
2145  014334  052777  004000  165050           BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2146  014342  052777  000020  165034           BIS     #RCVEN,@RXCSR   ;LOAD RCVEN
2147  014350  052777  000020  165034           BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
```

```
2148  014356  012777  000426  165030          MOV   #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2149  014364  105777  165022          1$:     TSTB  @TXCSR                  ;CHECK TRANSMITTER DONE
2150  014370  100375                        BPL   1$                      ;WAIT TILL SET
2151  014372  012777  000125  165014        MOV   #125,@TXDBUF            ;LOAD DATA
2152  014400  012737  000005  014430        MOV   #5,68$            ;LOAD THE NUMBER
2153  014406  032777  004000  165000  66$:   BIT   #TIMER,@TXDBUF   ;CHECK THE TIMER BIT
2154  014414  001374                        BNE   66$               ;BR IF SET
2155  014416  032777  004000  164770  67$:   BIT   #TIMER,@TXDBUF   ;CHECK THE BIT
2156  014424  001774                        BEQ   67$               ;BR IF CLEAR
2157  014426  005327                        DEC   (PC)+             ;DECREMENT THE NUMBER
2158  014430  000005                  68$:  5                       ;OF TIMES TO REPEAT
2159  014432  001365                        BNE   66$               ;BR IF MORE TO GO
2160  014434  105777  164744                TSTB  @RXCSR                  ;CHECK FOR RECEIVER DONE
2161  014440  100002                        BPL   2$                      ;BR IF NOT SET
2162  014442  104021                        HLT   21                      ;DEVICE SYNCS UP IN LESS THAN 2 SYNCS!!
2163  014444  000472                        BR    5$                      ;LEAVE
2164  014446                          2$:
2165  014446  052777  000400  164736        BIS   #MRESET,@TXCSR  ;RESET THE DEVICE
2166  014454  004737  004772                JSR   PC,SMALL        ;WAIT FOR RESET TO FINISH
2167  014460  012777  101026  164722        MOV   #CRCEN!DECMOD!26,@PARCSR  ;LOAD THE MODE AND SYNC CHAR
2168  014466  052777  000020  164710        BIS   #RCVEN,@RXCSR           ;TURN ON RECEIVER
2169  014474  052777  004000  164710        BIS   #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2170  014502  052777  000020  164702        BIS   #SEND,@TXCSR            ;TURN ON TRANSMITTER
2171  014510  012777  000426  164676        MOV   #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2172  014516  105777  164670          69$:  TSTB  @TXCSR            ;CHECK DONE
2173  014522  100375                        BPL   69$               ;BR IF NOT SET
2174  014524  012777  000426  164662        MOV   #TSOM!26,@TXDBUF    ;SEND SYNC
2175  014532  105777  164654          3$:   TSTB  @TXCSR                  ;CHECK DONE
2176  014536  100375                        BPL   3$                      ;WAIT
2177  014540  012777  000125  164646        MOV   #125,@TXDBUF            ;LOAD DATA
2178  014546  012737  000020  014576        MOV   #20,74$           ;LOAD THE NUMBER
2179  014554  032777  004000  164632  72$:   BIT   #TIMER,@TXDBUF   ;CHECK THE TIMER BIT
2180  014562  001374                        BNE   72$               ;BR IF SET
2181  014564  032777  004000  164622  73$:   BIT   #TIMER,@TXDBUF   ;CHECK THE BIT
2182  014572  001774                        BEQ   73$               ;BR IF CLEAR
2183  014574  005327                        DEC   (PC)+             ;DECREMENT THE NUMBER
2184  014576  000020                  74$:  20                      ;OF TIMES TO REPEAT
2185  014600  001365                        BNE   72$               ;BR IF MORE TO GO
2186  014602  105777  164576                TSTB  @RXCSR                  ;CHECK FOR DONE
2187  014606  100401                        BMI   4$                      ;BR IF SET
2188  014610  104022                        HLT   22                      ;FAILED TO RECEIVE DATA
2189  014612  017737  164570  001236  4$:   MOV   @RXDBUF,TEMP1          ;READ DATA
2190  014620  122737  000125  001236        CMPB  #125,TEMP1              ;CHECK IT
2191  014626  001401                        BEQ   5$                      ;BR IF MATCH
2192  014630  104022                        HLT   22                      ;DATA COMPARE ERROR
2193  014632                          5$:
2194  014632  052777  000400  164552        BIS   #MRESET,@TXCSR  ;RESET THE DEVICE
2195  014640  004737  004772                JSR   PC,SMALL        ;WAIT FOR RESET TO FINISH
2196  014644  104400                        SCOPE                         ;SCOPE THIS TEST
2197
2198
2199                                  ;************************** TEST 17 ****************************
2200                                  ;*THIS TEST PROVES THE RECEIVER WILL STRIP THE FIRST
2201                                  ;*TWO SYNCS AND WILL PRESENT ALL SUBSEQUENT SYNCS.
2202                                  ;;***********************************************************
2203
```

```
2204                                    ;:************
2205                                    ;            *
2206                                    ; TEST 17
2207                                    ;            *
2208                                    ;:************
2209                                    ;:*****************************************************************************
2210  014646  012737  000017  001226    TST17:  MOV     #17,@#TSTNO
2211  014654  012737  015070  001216            MOV     #TST20,NEXT
2212  014662  052777  000400  164522            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2213  014670  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2214  014674  012777  101026  164506            MOV     #DECMOD!26!CRCEN,@PARCSR
2215  014702  052777  004000  164502            BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2216  014710  052777  000020  164466            BIS     #RCVEN,@RXCSR   ;LOAD RCVEN
2217  014716  052777  000020  164466            BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2218  014724  012777  000426  164462            MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2219  014732  032777  004000  164454  64$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2220  014740  001374                            BNE     64$             ;BR IF SET
2221  014742  032777  004000  164444  65$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2222  014750  001774                            BEQ     65$             ;BR IF CLEAR
2223  014752  105777  164434          69$:      TSTB    @TXCSR          ;CHECK DONE
2224  014756  100375                            BPL     69$             ;BR IF NOT SET
2225  014760  012777  000426  164426            MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2226  014766  105777  164420          70$:      TSTB    @TXCSR          ;CHECK DONE
2227  014772  100375                            BPL     70$             ;BR IF NOT SET
2228  014774  012777  000426  164412            MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2229  015002  012737  000020  015032            MOV     #20,75$         ;LOAD THE NUMBER
2230  015010  032777  004000  164376  73$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2231  015016  001374                            BNE     73$             ;BR IF SET
2232  015020  032777  004000  164366  74$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2233  015026  001774                            BEQ     74$             ;BR IF CLEAR
2234  015030  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
2235  015032  000020                    75$:    20                      ;OF TIMES TO REPEAT
2236  015034  001365                            BNE     73$             ;BR IF MORE TO GO
2237  015036  105777  164342                    TSTB    @RXCSR                  ;CHECK FOR DONE
2238  015042  100401                            BMI     1$                      ;BR IF SET
2239  015044  104021                            HLT     21                      ;DONE NOT SET-DEVICE FAILED TO SYNC UP
2240  015046  117737  164334  001236  1$:       MOVB    @RXDBUF,TEMP1           ;READ BUFFER
2241  015054  122737  000026  001236            CMPB    #26,TEMP1               ;CHECK FOR SYNC
2242  015062  001401                            BEQ     2$                      ;BR IF OK
2243  015064  104022                            HLT     22                      ;DATA ERROR
2244  015066  104400                    2$:     SCOPE                           ;SCOPE THIS TEST
2245
2246
2247                                            ;************************** TEST 20 ***************************
2248                                            ;*THIS TEST PROVES THE DUP11 WILL
2249                                            ;*IDLE SYNCS.  IDLE 64. SYNCS
2250                                            ;:*************************************************************
2251
2252                                    ;:************
2253                                    ;            *
2254                                    ; TEST 20
2255                                    ;            *
2256                                    ;:************
2257                                    ;:*****************************************************************************
2258  015070  012737  000020  001226    TST20:  MOV     #20,@#TSTNO
2259  015076  012737  015334  001216            MOV     #TST21,NEXT
```

```
2260  015104  052777  000400  164300          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2261  015112  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2262  015116  012777  101026  164264          MOV     #DECMOD!26!CRCEN,@PARCSR
2263  015124  052777  004000  164260          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2264  015132  052777  000020  164244          BIS     #RCVEN,@RXCSR   ;LOAD RCVEN
2265  015140  052777  000020  164244          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2266  015146  012777  000426  164240          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2267  015154  105777  164232          64$:    TSTB    @TXCSR          ;CHECK DONE
2268  015160  100375                          BPL     64$             ;BR IF NOT SET
2269  015162  012777  000426  164224          MOV     #TSOM!26,@TXDBUF   ;SEND SYNC
2270  015170  105777  164216          65$:    TSTB    @TXCSR          ;CHECK DONE
2271  015174  100375                          BPL     65$             ;BR IF NOT SET
2272  015176  012777  000426  164210          MOV     #TSOM!26,@TXDBUF   ;SEND SYNC
2273  015204  005037  001236                  CLR     TEMP1
2274  015210  005037  001240                  CLR     TEMP2
2275  015214  012737  000100  001236          MOV     #64.,TEMP1              ;LOAD # OF SYNCS
2276  015222  012737  000010  015252          MOV     #10,70$         ;LOAD THE NUMBER
2277  015230  032777  004000  164156  68$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2278  015236  001374                          BNE     68$             ;BR IF SET
2279  015240  032777  004000  164146  69$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2280  015246  001774                          BEQ     69$             ;BR IF CLEAR
2281  015250  005327                  70$:    DEC     (PC)+           ;DECREMENT THE NUMBER
2282  015252  000010                          10                      ;OF TIMES TO REPEAT
2283  015254  001365                          BNE     68$             ;BR IF MORE TO GO
2284  015256  105777  164130          1$:     TSTB    @TXCSR                  ;CHECK DONE
2285  015262  100401                          BMI     2$                      ;BR IF SET
2286  015264  104016                          HLT     16                      ;DONE FAILED TO SET
2287  015266  012777  000426  164120  2$:     MOV     #TSOM!26,@TXDBUF        ;LOAD A SYNC
2288  015274  005337  001236                  DEC     TEMP1                   ;LOWER THE # OF SYNCS TO DO
2289  015300  001001                          BNE     4$                      ;BR IF MORE TO GO
2290  015302  104400                  3$:     SCOPE                           ;SCOPE THIS TEST
2291
2292  015304  105777  164074          4$:     TSTB    @RXCSR          ;CHECK RECEIVER DONE
2293  015310  100375                          BPL     4$              ;WAIT TILL SET
2294  015312  017737  164070  001240          MOV     @RXDBUF,TEMP2   ;GET THE BUFFER
2295  015320  122737  000026  001240          CMPB    #26,TEMP2       ;CHECK IT FOR SYNC
2296  015326  001753                          BEQ     1$              ;BR IF OK
2297  015330  104021                          HLT     21              ;CHARACTER IS TEMP2 NOT A SYNC!
2298  015332  000763                          BR      3$              ;LEAVE TEST
2299
2300
2301                                  ;************************** TEST 21 ***************************
2302                                  ;*THIS TEST PROVES THE STRIP SYNC
2303                                  ;*FUNCTION OF THE RECEIVER. SYNC UP
2304                                  ;*THE RECEIVER, SEND DATA WITH A SYNC
2305                                  ;*CHARACTER IMBEDDED AND CHECK FOR
2306                                  ;*THE SYNC TO BE RECEIVED.
2307                                  ;;*************************************************************
2308                          ;;**********
2309                          ;         *
2310                          ; TEST 21
2311                          ;         *
2312                          ;;**********
2313                          ;;*************************************************************************
2314  015334  012737  000021  001226  TST21:  MOV     #21,@#TSTNO
2315  015342  012737  016054  001216          MOV     #TST22,NEXT
```

```
2316  015350  012737  000340  177776          MOV     #340,PS                 ;RAISE STATUS
2317  015356  004537  006674                  JSR     R5,SETVEC               ;SET UP VECTORS
2318  015362  015642                          5$                              ;BASED ON
2319  015364  006722                          NO.BTRAP                        ;THESE
2320  015366     340     340                  .BYTE   340,340                 ;PARAMETERS
2321
2322  015370  052777  000400  164014          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2323  015376  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2324  015402  012777  101026  164000          MOV     #DECMOD!26!CRCEN,@PARCSR
2325  015410  052777  004000  163774          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2326  015416  052777  000420  163760          BIS     #RCVEN!STPSYN,@RXCSR    ;LOAD RCVEN!STPSYN
2327  015424  052777  000020  163760          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2328  015432  012777  000426  163754          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2329  015440  105777  163746          64$:    TSTB    @TXCSR          ;CHECK DONE
2330  015444  100375                  BPL     64$             ;BR IF NOT SET
2331  015446  012777  000426  163740          MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2332  015454  105777  163732          65$:    TSTB    @TXCSR          ;CHECK DONE
2333  015460  100375                  BPL     65$             ;BR IF NOT SET
2334  015462  012777  000426  163724          MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2335  015470  105777  163716          66$:    TSTB    @TXCSR          ;CHECK DONE
2336  015474  100375                  BPL     66$             ;BR IF NOT SET
2337  015476  012777  000426  163710          MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2338  015504  005037  177776                  CLR     PS                      ;LOWER PS
2339  015510  052777  000100  163666          BIS     #RINTEN,@RXCSR          ;TURN ON INTERRUPTS
2340  015516  105777  163670          1$:     TSTB    @TXCSR                  ;CHECK TX DONE
2341  015522  100375                  BPL     1$                      ;WAIT FOR SET
2342  015524  012777  000252  163662          MOV     #252,@TXDBUF            ;LOAD A CHARACTER
2343  015532  105777  163654          2$:     TSTB    @TXCSR                  ;CHECK TX DONE
2344  015536  100375                  BPL     2$                      ;WAIT TO BE SET
2345  015540  012777  000026  163646          MOV     #26,@TXDBUF             ;LOAD THE SYNC CHAR
2346  015546  105777  163640          3$:     TSTB    @TXCSR                  ;CHECK DONE AGAIN
2347  015552  100375                  BPL     3$                      ;WAIT
2348  015554  012777  000125  163632          MOV     #125,@TXDBUF            ;LOAD ANOTHER CHARACTER
2349  015562  105777  163624          4$:     TSTB    @TXCSR                  ;CHECK DONE
2350  015566  100375                  BPL     4$                      ;WAIT
2351  015570  012777  001000  163616          MOV     #TEOM,@TXDBUF           ;SET END OF MESSAGE
2352  015576  042777  000020  163606          BIC     #SEND,@TXCSR            ;TURN OFF TRANSMITTER
2353  015604  012737  000050  015634          MOV     #40.,71$                ;LOAD THE NUMBER
2354  015612  032777  004000  163574  69$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2355  015620  001374                  BNE     69$             ;BR IF SET
2356  015622  032777  004000  163564  70$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2357  015630  001774                  BEQ     70$             ;BR IF CLEAR
2358  015632  005327                  DEC     (PC)+           ;DECREMENT THE NUMBER
2359  015634  000050          71$:    40.                     ;OF TIMES TO REPEAT
2360  015636  001365                  BNE     69$             ;BR IF MORE TO GO
2361  015640  104023                  HLT     23                      ;FAILED TO TAKE A RECEIVER INTERRUPT
2362
2363                                  ;RECEIVER INTERRUPT SERVICE ROUTINE
2364  015642  017700  163536          5$:     MOV     @RXCSR,R0               ;READ CSR
2365  015646  017701  163534                  MOV     @RXDBUF,R1              ;READ BUFFER
2366  015652  032700  000200                  BIT     #RXDONE,R0              ;CHECK FOR DONE
2367  015656  001001                  BNE     6$                      ;BR IF SET
2368  015660  104024                  HLT     24                      ;RX DONE FAILED TO SET-ERRONEOUS INTERRUPT
2369  015662  032700  004000          6$:     BIT     #RECACT,R0              ;CHECK FOR ACTIVE
2370  015666  001001                  BNE     7$                      ;BR IF SET
2371  015670  104025                  HLT     25                      ;RX ACTIVE FAILED TO SET
```

```
2372  015672                                7$:
2373  015672  005701                              TST     R1              ;CHECK FOR ERROR
2374  015674  100001                              BPL     10$             ;BR IF NO ERROR
2375  015676  104026                              HLT     26              ;RECEIVER ERROR
2376  015700  122701  000252                10$:   CMPB    #252,R1         ;CHECK FOR CORRECT DATA
2377  015704  001401                              BEQ     11$             ;BR IF OK
2378  015706  104022                              HLT     22              ;DATA FAILED TO MATCH
2379  015710  012777  015720  163456        11$:   MOV     #12$,@DUPRVC    ;LOAD VECTOR
2380  015716  000455                              BR      20$             ;CONTINUE
2381  015720  017700  163460                12$:   MOV     @RXCSR,R0       ;READ CSR
2382  015724  017701  163456                       MOV     @RXDBUF,R1      ;READ BUFFER
2383  015730  032700  000200                       BIT     #RXDONE,R0      ;CHECK FOR DONE
2384  015734  001001                              BNE     13$             ;BR IF OK
2385  015736  104024                              HLT     24              ;RX DONE FAILED TO SET-ERRONEOUS INTERRUPT
2386  015740  005701                        13$:   TST     R1              ;TEST FOR ERROR
2387  015742  100001                              BPL     14$             ;BR IF NO ERROR
2388  015744  104026                              HLT     26              ;ERROR SET
2389  015746  122701  000026                14$:   CMPB    #26,R1          ;CHECK CHARACTER
2390  015752  001422                              BEQ     16$             ;BR IF OK-IF NOT, THEN
2391  015754  122701  000125                       CMPB    #125,R1         ;CHECK FOR CLEARING SYNC
2392  015760  001402                              BEQ     15$             ;BR IF A NEXT CHARACTER
2393  015762  104022                              HLT     22              ;ERRONEOUS CHARACTER
2394  015764  000415                              BR      16$             ;BR TO END OF TEST
2395  015766  104021                        15$:   HLT     21              ;STRIPPED OUT THE SYNC CHAR!!
2396  015770  012777  016000  163376               MOV     #21$,@DUPRVC    ;SET UP VECTOR
2397  015776  000425                              BR      20$             ;LEAVE
2398  016000  017700  163400                21$:   MOV     @RXCSR,R0       ;GET CSR
2399  016004  017701  163376                       MOV     @RXDBUF,R1      ;GET BUFFER
2400  016010  122701  000125                       CMPB    #125,R1         ;CHECK DATA
2401  016014  001401                              BEQ     16$             ;BR IF A MATCH
2402  016016  104022                              HLT     22              ;DATA COMPARE ERROR
2403  016020  032777  004000  163356        16$:   BIT     #RECACT,@RXCSR  ;TEST ACTIVE
2404  016026  001001                              BNE     17$             ;BR IF ON
2405  016030  104025                              HLT     25              ;ACTIVE SHOULD BE ON
2406  016032                                17$:
2407  016032  052777  000400  163352               BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2408  016040  004737  004772                       JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2409  016044  012706  001150                       MOV     #STACK,SP               ;RESET STACK
2410  016050  104400                              SCOPE                   ;SCOPE THIS TEST
2411  016052  000002                        20$:   RTI                     ;RETURN
2412
2413
2414                                              ;************************** TEST 22 **************************
2415                                              ;*THIS TEST PROVES THAT A BINARY COUNT
2416                                              ;*PATTERN CAN BE RUN IN DEC MODE
2417                                              ;*WITHOUT A BCC CALCULATION
2418                                              ;;*********************************************************************
2419                                        ;;************
2420                                        ;           *
2421                                        ; TEST 22
2422                                        ;           *
2423                                        ;;************
2424                                        ;;*********************************************************************
2425  016054  012737  000022  001226        TST22:  MOV     #22,@#TSTNO
2426  016062  012737  016536  001216                MOV     #TST23,NEXT
2427  016070  012737  000340  177776                MOV     #340,PS
```

```
2428  016076  005000                          CLR     R0                      ;CLR OUT DATA POINTER
2429  016100  005001                          CLR     R1                      ;DITTO
2430  016102  004537  006674                  JSR     R5,SETVEC               ;SET UP INTERRUPTS
2431  016106  016316                          4$                              ;RECEIVER
2432  016110  016470                          17$                             ;TRANSMITTER
2433  016112    340     340                   .BYTE   340,340                 ;LEVEL
2434
2435  016114  052777  000400  163270          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2436  016122  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2437  016126  012777  101026  163254          MOV     #DECMOD!26!CRCEN,@PARCSR
2438  016134  052777  004000  163250          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2439  016142  052777  000020  163234          BIS     #RCVEN,@RXCSR   ;LOAD RCVEN
2440  016150  052777  000020  163234          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2441  016156  012777  000426  163230          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2442  016164  032777  004000  163222  64$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2443  016172  001374                          BNE     64$             ;BR IF SET
2444  016174  032777  004000  163212  65$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2445  016202  001774                          BEQ     65$             ;BR IF CLEAR
2446  016204  105777  163202          69$:    TSTB    @TXCSR          ;CHECK DONE
2447  016210  100375                          BPL     69$             ;BR IF NOT SET
2448  016212  012777  000426  163174          MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2449  016220  005037  177776                  CLR     PS
2450  016224  052777  000100  163152          BIS     #RINTEN,@RXCSR          ;TURN ON INT ENABLES
2451  016232  052777  000100  163152          BIS     #TXINTE,@TXCSR          ;DITTO
2452  016240                          30$:
2453  016240  012737  000310  016270          MOV     #200.,74$               ;LOAD THE NUMBER
2454  016246  032777  004000  163140  72$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2455  016254  001374                          BNE     72$             ;BR IF SET
2456  016256  032777  004000  163130  73$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2457  016264  001774                          BEQ     73$             ;BR IF CLEAR
2458  016266  005327                          DEC     (PC)+           ;DECREMENT THE NUMBER
2459  016270  000310                  74$:    200.                    ;OF TIMES TO REPEAT
2460  016272  001365                          BNE     72$             ;BR IF MORE TO GO
2461  016274  104023                          HLT     23                      ;FAILED TO FINISH TEST
2462  016276                          3$:
2463  016276  052777  000400  163106          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2464  016304  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2465  016310  012706  001150                  MOV     #STACK,SP               ;RESET THE STACK
2466  016314  104400                          SCOPE                           ;SCOPE THIS TEST
2467
2468
2469                                          ;RECEIVER INT SVC ROUTINE
2470  016316  017702  163062          4$:     MOV     @RXCSR,R2               ;SAVE CSR
2471  016322  017703  163060                  MOV     @RXDBUF,R3              ;SAVE BUFFER
2472  016326  032702  004000                  BIT     #RECACT,R2              ;TEST RX ACTIVE
2473  016332  001004                          BNE     5$                      ;BR IF OK
2474  016334  104025                          HLT     25                      ;ACTIVE NOT SET
2475  016336  012716  016276                  MOV     #3$,(SP)                ;SETUP FOR RETURN
2476  016342  000432                          BR      12$
2477  016344  032702  000200          5$:     BIT     #RXDONE,R2              ;TEST DONE
2478  016350  001004                          BNE     6$                      ;BR IF OK
2479  016352  104024                          HLT     24                      ;FALSE INTERRUPT
2480  016354  012716  016276                  MOV     #3$,(SP)                ;SETUP FOR RETURN
2481  016360  000423                          BR      12$
2482  016362  005703                  6$:     TST     R3                      ;CHECK FOR ERROR
2483  016364  100004                          BPL     7$                      ;BR IF NO ERROR
```

```
2484  016366  104026                          HLT     26                      ;DATA ERROR
2485  016370  012716  016276                  MOV     #3$,(SP)                ;SET UP RETURN
2486  016374  000415                          BR      12$
2487  016376  120103                  7$:     CMPB    R1,R3                   ;CHECK DATA
2488  016400  001404                          BEQ     10$                     ;BR IF OK
2489  016402  104022                          HLT     22                      ;BAD DATA
2490  016404  012716  016276                  MOV     #3$,(SP)                ;SETUP RETURN
2491  016410  000407                          BR      12$
2492  016412  005201                  10$:    INC     R1                      ;UPDATE DATA
2493  016414  001002                          BNE     11$                     ;BR IF MORE TO GO
2494  016416  012716  016276                  MOV     #3$,(SP)        ;SETUP RETURN
2495  016422  012777  016432  162744  11$:    MOV     #22$,@DUPRVC            ;SETUP NEW RETURN FOR INTERRUPT
2496  016430  000002                  12$:    RTI                             ;RETURN
2497  016432  017702  162746          22$:    MOV     @RXCSR,R2
2498  016436  017703  162744                  MOV     @RXDBUF,R3
2499  016442  005703                          TST     R3
2500  016444  100001                          BPL     23$
2501  016446  104026                          HLT     26                      ;ERROR
2502  016450  120103                  23$:    CMPB    R1,R3
2503  016452  001401                          BEQ     24$
2504  016454  104022                          HLT     22                      ;DATA COMPARE ERROR
2505  016456  105201                  24$:    INCB    R1
2506  016460  001363                          BNE     12$
2507  016462  012716  016276                  MOV     #3$,(SP)
2508  016466  000760                          BR      12$
2509
2510                                          ;TRANSMITTER
2511  016470  010077  162720          17$:    MOV     R0,@TXDBUF              ;PUSH OUT DATA
2512  016474  105200                          INCB    R0                      ;UPDATE IT
2513  016476  001014                          BNE     21$                     ;BR IF MORE
2514  016500  105777  162706          20$:    TSTB    @TXCSR                  ;CHECK FOR NEXT DONE
2515  016504  100375                          BPL     20$                     ;WAIT
2516  016506  052777  001000  162700        BIS     #TEOM,@TXDBUF           ;END MSG
2517  016514  042777  000120  162670        BIC     #SEND!TXINTE,@TXCSR             ;SHUT OF TRANSMITTER
2518  016522  012777  006722  162650        MOV     #NO.BTRAP,@DUPTVC       ;RESET VECTOR ADRS
2519  016530  012716  016240          21$:    MOV     #30$,(SP)
2520  016534  000002                          RTI                             ;RETURN
2521
2522
2523
2524                                          ;************************** TEST 23 **************************
2525                                          ;*THIS TEST PROVES THAT A BINARY COUNT
2526                                          ;*PATTERN CAN BE RUN IN DEC MODE
2527                                          ;*WITH A BCC CALCULATION USING
2528                                          ;*THE CRC16 POLYNOMIAL
2529                                          ;;*********************************************************************
2530                                  ;;**********
2531                                  ;         *
2532                                  ; TEST 23
2533                                  ;         *
2534                                  ;;**********
2535                                  ;;*********************************************************************
2536  016536  012737  000023  001226  TST23:  MOV     #23,@#TSTNO
2537  016544  012737  017360  001216          MOV     #TST24,NEXT
2538  016552  012737  000340  177776          MOV     #340,PS
2539  016560  005000                          CLR     R0
```

```
2540  016562  012737  120001  007100          MOV     #CRC16,XPOLY            ;SET THE POLYNOMIAL
2541  016570  005037  007104                  CLR     CALBCC                  ;CLEAR OUT OLD BCC
2542  016574  013737  007104  016616  1$:     MOV     CALBCC,35$              ;LOAD BCC
2543  016602  010037  016614                  MOV     R0,2$                   ;LOAD DATA
2544  016606  004537  006726                  JSR     R5,SIMBCC               ;CALCULATE A SOFTWARE BCC
2545  016612  000010                          8.                              ;BASED
2546  016614  000000                  2$:     .WORD   0                       ;ON THESE
2547  016616  000000                  35$:    .WORD   0                       ;PARAMETERS
2548  016620  105200                          INCB    R0                      ;UPDATE DATA
2549  016622  001364                          BNE     1$                      ;BR IF MORE TO GO
2550  016624  005000                          CLR     R0                      ;CLR OUT DATA POINTER
2551  016626  005001                          CLR     R1                      ;DITTO
2552  016630  004537  006674                  JSR     R5,SETVEC               ;SET UP INTERRUPTS
2553  016634  017044                          4$                              ;RECEIVER
2554  016636  017312                          17$                             ;TRANSMITTER
2555  016640     340     340                  .BYTE   340,340                 ;LEVEL
2556
2557  016642  052777  000400  162542          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2558  016650  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
2559  016654  012777  100026  162526          MOV     #DECMOD!26,@PARCSR      ;LOAD THE MODE AND SYNC CHARACTER
2560  016662  052777  004000  162522          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2561  016670  052777  000020  162506          BIS     #RCVEN,@RXCSR   ;LOAD RCVEN
2562  016676  052777  000020  162506          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2563  016704  012777  000426  162502          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2564  016712  032777  004000  162474  64$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2565  016720  001374                          BNE     64$                     ;BR IF SET
2566  016722  032777  004000  162464  65$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2567  016730  001774                          BEQ     65$                     ;BR IF CLEAR
2568  016732  105777  162454          69$:    TSTB    @TXCSR                  ;CHECK DONE
2569  016736  100375                          BPL     69$                     ;BR IF NOT SET
2570  016740  012777  000426  162446          MOV     #TSOM!26,@TXDBUF        ;SEND SYNC
2571  016746  005037  177776                  CLR     PS
2572  016752  052777  000100  162424          BIS     #RINTEN,@RXCSR          ;TURN ON INT ENABLES
2573  016760  052777  000100  162424          BIS     #TXINTE,@TXCSR          ;DITTO
2574  016766                          30$:
2575  016766  012737  000310  017016          MOV     #200.,74$               ;LOAD THE NUMBER
2576  016774  032777  004000  162412  72$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2577  017002  001374                          BNE     72$                     ;BR IF SET
2578  017004  032777  004000  162402  73$:    BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2579  017012  001774                          BEQ     73$                     ;BR IF CLEAR
2580  017014  005327                          DEC     (PC)+                   ;DECREMENT THE NUMBER
2581  017016  000310                  74$:    200.                            ;OF TIMES TO REPEAT
2582  017020  001365                          BNE     72$                     ;BR IF MORE TO GO
2583  017022  104023                          HLT     23                              ;FAILED TO FINISH TEST
2584  017024                          3$:
2585  017024  052777  000400  162360          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2586  017032  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
2587  017036  012706  001150                  MOV     #STACK,SP                       ;RESET THE STACK
2588  017042  104400                          SCOPE                           ;SCOPE THIS TEST
2589
2590
2591                                  ;RECEIVER INT SVC ROUTINE
2592  017044  017702  162334          4$:     MOV     @RXCSR,R2               ;SAVE CSR
2593  017050  017703  162332                  MOV     @RXDBUF,R3              ;SAVE BUFFER
2594  017054  032702  004000                  BIT     #RECACT,R2              ;TEST RX ACTIVE
2595  017060  001004                          BNE     5$                      ;BR IF OK
```

```
2596  017062  104025                          HLT     25                      ;ACTIVE NOT SET
2597  017064  012716  017024                  MOV     #3$,(SP)                ;SETUP FOR RETURN
2598  017070  000433                          BR      12$
2599  017072  032702  000200          5$:     BIT     #RXDONE,R2              ;TEST DONE
2600  017076  001004                          BNE     6$                      ;BR IF OK
2601  017100  104024                          HLT     24                      ;FALSE INTERRUPT
2602  017102  012716  017024                  MOV     #3$,(SP)                ;SETUP FOR RETURN
2603  017106  000424                          BR      12$
2604  017110  005703                  6$:     TST     R3                      ;CHECK FOR ERROR
2605  017112  100004                          BPL     7$                      ;BR IF NO ERROR
2606  017114  104026                          HLT     26                      ;DATA ERROR
2607  017116  012716  017024                  MOV     #3$,(SP)                ;SET UP RETURN
2608  017122  000416                          BR      12$
2609  017124  120103                  7$:     CMPB    R1,R3                   ;CHECK DATA
2610  017126  001404                          BEQ     10$                     ;BR IF OK
2611  017130  104022                          HLT     22                      ;BAD DATA
2612  017132  012716  017024                  MOV     #3$,(SP)                ;SETUP RETURN
2613  017136  000410                          BR      12$
2614  017140  005201                  10$:    INC     R1                      ;UPDATE DATA
2615  017142  001003                          BNE     11$                     ;BR IF MORE TO GO
2616  017144  012716  017162                  MOV     #13$,(SP)       ;SETUP TO FINISH TEST
2617  017150  000403                          BR      12$
2618  017152  012777  017252  162214  11$:    MOV     #22$,@DUPRVC            ;SETUP NEW RETURN FOR INTERRUPT
2619  017160  000002                  12$:    RTI                             ;RETURN
2620  017162  105777  162216          13$:    TSTB    @RXCSR                  ;TEST DONE
2621  017166  100375                          BPL     13$                     ;WAIT
2622  017170  017737  162212  001236          MOV     @RXDBUF,TEMP1           ;GET DATA
2623  017176  105777  162202          14$:    TSTB    @RXCSR                  ;CHECK DONE FOR  HALF OF CRC
2624  017202  100375                          BPL     14$                     ;WAIT
2625  017204  017737  162176  001240          MOV     @RXDBUF,TEMP2           ;MOVE IT
2626  017212  113737  001240  001237          MOVB    TEMP2,TEMP1+1           ;COMBINE BCC CHARACTER
2627  017220  023737  007104  001236          CMP     CALBCC,TEMP1            ;BR IF A MATCH
2628  017226  001401                          BEQ     15$                     ;AFTER CHECKING IT
2629  017230  104027                          HLT     27                      ;CRC COMPARE ERROR--THE
2630                                                                          ;SOFTWARE DOESN'T AGREE
2631                                                                          ;WITH WHAT THE TRANSMITTER
2632                                                                          ;SENT.  SEE THE FRONT OF
2633                                                                          ;THE LISTING FOR SPECIAL
2634                                                                          ;CRC DEBUG AID TEST.
2635  017232  032737  010000  001240  15$:     BIT     #CRCERR,TEMP2           ;CHECK FOR ERROR
2636  017240  001001                          BNE     16$                     ;BR IF OK
2637  017242  104030                          HLT     30                      ;HARDWARE DETECTED CRC ERROR
2638                                                                          ;RECEIVER DOESN'T AGREE WITH
2639                                                                          ;WHAT THE TRANSMITTER SENT
2640                                                                          ;SEE FRONT OF LISTING FOR
2641                                                                          ;SPECIAL CRC DEBUG AID
2642  017244  012716  017024          16$:    MOV     #3$,(SP)                ;LOAD END OF TEST
2643  017250  000743                          BR      12$                     ;RETURN
2644  017252  017702  162126          22$:    MOV     @RXCSR,R2
2645  017256  017703  162124                  MOV     @RXDBUF,R3
2646  017262  005703                          TST     R3
2647  017264  100001                          BPL     23$
2648  017266  104026                          HLT     26                      ;ERROR
2649  017270  120103                  23$:    CMPB    R1,R3
2650  017272  001401                          BEQ     24$
2651  017274  104022                          HLT     22                      ;DATA COMPARE ERROR
```

```
2652  017276  105201                    24$:    INCB    R1
2653  017300  001327                            BNE     12$
2654  017302  012777  017162  162064            MOV     #13$,@DUPRVC
2655  017310  000723                            BR      12$
2656
2657                                            ;TRANSMITTER
2658  017312  010077  162076            17$:    MOV     R0,@TXDBUF              ;PUSH OUT DATA
2659  017316  105200                            INCB    R0                      ;UPDATE IT
2660  017320  001014                            BNE     21$                     ;BR IF MORE
2661  017322  105777  162064            20$:    TSTB    @TXCSR                  ;CHECK FOR NEXT DONE
2662  017326  100375                            BPL     20$                     ;WAIT
2663  017330  052777  001000  162056            BIS     #TEOM,@TXDBUF           ;END MSG
2664  017336  042777  000120  162046            BIC     #SEND!TXINTE,@TXCSR             ;SHUT OF TRANSMITTER
2665  017344  012777  006722  162026            MOV     #NO.BTRAP,@DUPTVC       ;RESET VECTOR ADRS
2666  017352  012716  016766            21$:    MOV     #30$,(SP)
2667  017356  000002                            RTI                             ;RETURN
2668
2669
2670
2671
2672                                            ;************************** TEST 24 **************************
2673                                            ;*TEST TO PROVE THE DEVICE IDLES SYNCS AND
2674                                            ;*WILL SHIFT OUT DATA AT THE APPROPRIATE TIME
2675                                            ;:**********************************************************
2676
2677                                    ;:*************
2678                                    ;             *
2679                                    ; TEST 24
2680                                    ;             *
2681                                    ;:*************
2682                                    ;:**************************************************************************
2683  017360  012737  000024  001226    TST24:  MOV     #24,@#TSTNO
2684  017366  012737  017556  001216            MOV     #TST25,NEXT
2685  017374  052777  000400  162010            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2686  017402  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2687  017406  052777  014000  161776            BIS     #MMODE,@TXCSR   ;ENTER MAINT MODE
2688  017414  012777  000020  161762            MOV     #RCVEN,@RXCSR           ;TURN ON RECEIVER
2689  017422  012777  100026  161760            MOV     #DECMOD!26,@PARCSR              ;ENTER DECMODE AND SYNC CHAR
2690  017430  052777  000020  161754            BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2691  017436  012777  000426  161750            MOV     #TSOM!26,@TXDBUF        ;PUSH OUT SYNCS
2692  017444  104412  000044                    PKCLK   ,36.
2693  017450  012777  000252  161736            MOV     #252,@TXDBUF            ;LOAD DATA
2694  017456  104412  000024                    PKCLK   ,20.                    ;PUSH OUT ANOTHER SYNC
2695  017462  105777  161716                    TSTB    @RXCSR          ;CHECK TO SEE IF SYNC ARRIVED
2696  017466  100401                            BMI     1$                      ;BR IF YES
2697  017470  104021                            HLT     21
2698  017472  017737  161710  001324    1$:     MOV     @RXDBUF,DATA            ;GET THE REC CHAR
2699  017500  122737  000026  001324            CMPB    #26,DATA                ;CHECK FOR SYNC
2700  017506  001401                            BEQ     2$                      ;BR IF MATCH
2701  017510  104021                            HLT     21                      ;FAILED TO RECEIVE THIRD SYNC
2702  017512  042777  000020  161672    2$:     BIC     #SEND,@TXCSR    ;TURN OFF TRANSMITTER
2703  017520  104412  000016                    PKCLK   ,14.            ;PUSH OUT DATA
2704  017524  105777  161654                    TSTB    @RXCSR                  ;CHECK FOR REC DATA
2705  017530  100401                            BMI     3$                      ;BR IF YES
2706  017532  104026                            HLT     26                      ;FAILED TO GET A DATA DONE
2707  017534  017737  161646  001324    3$:     MOV     @RXDBUF,DATA            ;GET THE DATA
```

```
2708  017542  122737  000252  001324          CMPB    #252,DATA               ;COMPARE IT
2709  017550  001401                          BEQ     4$                      ;BR IF A MATCH
2710  017552  104022                          HLT     22                      ;DATA ERROR
2711  017554  104400                  4$:     SCOPE                           ;SCOPE THIS TEST
2712
2713
2714                                          ;*************************** TEST 25 ****************************
2715                                          ;*TEST TO PROVE THE DEVICE WILL
2716                                          ;*WORK WITH EVERY POSSIBLE SYNC CHARACTER
2717                                          ;;*************************************************************************
2718                                  ;:*************
2719                                  ;           *
2720                                  ; TEST 25
2721                                  ;           *
2722                                  ;:*************
2723                                  ;;*************************************************************************
2724  017556  012737  000025  001226  TST25:  MOV     #25,@#TSTNO
2725  017564  012737  020036  001216          MOV     #TST26,NEXT
2726  017572  012737  000340  177776          MOV     #340,PS
2727  017600  012702  000003                  MOV     #3,R2                   ;SET UP # OF SYNC
2728  017604  005037  001236                  CLR     TEMP1
2729  017610  005037  001240                  CLR     TEMP2
2730  017614  005000                          CLR     R0                      ;CLEAR OUT DATA
2731  017616  005001                          CLR     R1                      ;CLEAR SYNC
2732  017620  052737  000400  001236          BIS     #TSOM,TEMP1             ;LOAD TSOM
2733  017626  052737  101000  001240          BIS     #DECMODE!CRCEN,TEMP2            ;LOAD DEC MODE
2734  017634                          1$:
2735  017634  052777  000400  161550          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2736  017642  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
2737  017646  052777  000020  161530          BIS     #RCVEN,@RXCSR
2738  017654  013777  001240  161526          MOV     TEMP2,@PARCSR           ;LOAD THE MODE AND SYNC CHARACTER
2739  017662  052777  004020  161522          BIS     #SYSTST!SEND,@TXCSR     ;TURN ON MODE AND TRANSMITTER
2740  017670  105777  161516          2$:     TSTB    @TXCSR                  ;TEST FOR DONE
2741  017674  100375                          BPL     2$                      ;WAIT
2742  017676  013777  001236  161510          MOV     TEMP1,@TXDBUF           ;LOAD TSOM AND SYNC CHAR
2743  017704  005302                          DEC     R2                      ;DECREMENT # OF SYNCS TO SEND
2744  017706  001370                          BNE     2$                      ;BR IF MORE TOO GO
2745  017710                          7$:
2746  017710  032777  004000  161476  64$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2747  017716  001374                          BNE     64$                     ;BR IF SET
2748  017720  032777  004000  161466  65$:    BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2749  017726  001774                          BEQ     65$                     ;BR IF CLEAR
2750  017730  105777  161450                  TSTB    @RXCSR
2751  017734  100405                          BMI     3$
2752  017736  005200                          INC     R0
2753  017740  022700  000040                  CMP     #40,R0
2754  017744  001361                          BNE     7$
2755  017746  104015                          HLT     15                      ;DONE FAILED TO SET AFTER THIRD SYNC
2756  017750  017737  161432  001244  3$:     MOV     @RXDBUF,TEMP4           ;GET THE BUFFER
2757  017756  032737  004000  001242          BIT     #RECACT,TEMP3           ;TEST ACTIVE
2758  017764  001001                          BNE     4$                      ;BR IF SET
2759  017766  104025                          HLT     25                      ;ACTIVE FAILED TO SET
2760  017770  005737  001244          4$:     TST     TEMP4                   ;TEST FOR ERROR
2761  017774  100001                          BPL     5$                      ;BR IF NO ERROR
2762  017776  104026                          HLT     26                      ;HARDWARE ERROR
2763  020000  123737  001236  001244  5$:     CMPB    TEMP1,TEMP4                     ;COMPARE SYNCS
```

```
2764  020006  001401                          BEQ     6$                      ;BR IF OK
2765  020010  104022                          HLT     22                      ;SYNC CHAR DOES NOT MATCH SENT
2766  020012                          6$:
2767  020012  012702  000003                  MOV     #3,R2                   ;SET UP FOR NEXT SYNC
2768  020016  005000                          CLR     R0                      ;DITTO
2769  020020  105201                          INCB    R1                      ;DITTO
2770  020022  110137  001236                  MOVB    R1,TEMP1
2771  020026  110137  001240                  MOVB    R1,TEMP2
2772  020032  001300                          BNE     1$                      ;BR IF MORE TO GO
2773  020034  104400                          SCOPE                           ;SCOPE THIS TEST
2774
2775
2776
2777                                          ;*************************** TEST 26 ****************************
2778                                          ;*THIS TEST PROVES THAT THE CRC ERROR BIT FUNCTIONS
2779                                          ;*CORRECTLY. FORCE AN ERROR AND VERIFY THE BIT.
2780                                          ;;*************************************************************************
2781                                  ;;*************
2782                                  ;               *
2783                                  ; TEST 26       *
2784                                  ;               *
2785                                  ;;*************
2786                                  ;;*************************************************************************
2787  020036  012737  000026  001226  TST26:  MOV     #26,@#TSTNO
2788  020044  012737  020376  001216          MOV     #TST27,NEXT
2789  020052  012737  000340  177776          MOV     #340,PS                 ;RAISE PROCESSOR STATUS
2790  020060  004537  006674                  JSR     R5,SETVEC               ;SETUP VECTORS
2791  020064  020340                          6$                              ;RECEIVER
2792  020066  020272                          3$                              ;TRANSMITTER
2793  020070    340     340                   .BYTE   340,340                 ;LEVEL
2794  020072  005001                          CLR     R1                      ;CLEAR CHAR COUNT
2795
2796  020074  052777  000400  161310          BIS     #MRESET,@TXCSR          ;RESET THE DEVICE
2797  020102  004737  004772                  JSR     PC,SMALL                ;WAIT FOR RESET TO FINISH
2798  020106  012777  100026  161274          MOV     #DECMOD!26,@PARCSR      ;LOAD THE MODE AND SYNC CHARACTER
2799  020114  052777  004000  161270          BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2800  020122  052777  000020  161254          BIS     #RCVEN,@RXCSR           ;LOAD RCVEN
2801  020130  052777  000020  161254          BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2802  020136  012777  000426  161250          MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2803  020144  105777  161242          64$:     TSTB    @TXCSR                  ;CHECK DONE
2804  020150  100375                          BPL     64$                     ;BR IF NOT SET
2805  020152  012777  000426  161234          MOV     #TSOM!26,@TXDBUF        ;SEND SYNC
2806  020160  105777  161226          65$:     TSTB    @TXCSR                  ;CHECK DONE
2807  020164  100375                          BPL     65$                     ;BR IF NOT SET
2808  020166  012777  000426  161220          MOV     #TSOM!26,@TXDBUF        ;SEND SYNC
2809  020174  005037  177776                  CLR     PS                      ;LOWER PROCESSOR STSTUS
2810  020200  052777  000100  161176          BIS     #RINTEN,@RXCSR          ;TURN ON INTERRUPT ENABLES
2811  020206  052777  000100  161176          BIS     #TXINTE,@TXCSR          ;DITTO
2812  020214                          1$:
2813  020214  012737  000040  020244          MOV     #32.,70$                ;LOAD THE NUMBER
2814  020222  032777  004000  161164  68$:     BIT     #TIMER,@TXDBUF          ;CHECK THE TIMER BIT
2815  020230  001374                          BNE     68$                     ;BR IF SET
2816  020232  032777  004000  161154  69$:     BIT     #TIMER,@TXDBUF          ;CHECK THE BIT
2817  020240  001774                          BEQ     69$                     ;BR IF CLEAR
2818  020242  005327                          DEC     (PC)+                   ;DECREMENT THE NUMBER
2819  020244  000040                  70$:     32.                             ;OF TIMES TO REPEAT
```

```
                                        BNE     68$             ;BR IF MORE TO GO
                                        HLT     23              ;FAILED TO FINISH TEST
                                2$:
                                        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
                                        JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
                                        MOV     #STACK,SP       ;RESET THE STACK
                                        SCOPE
                                        ;INTERRUPT SERVICE ROUTINES
                                ;TRANSMITTER
                                3$:     CLR     R0              ;CLEAR DATA
                                        MOV     R0,@TXDBUF      ;LOAD DATA TO BUFFER
                                        MOV     #4$,@DUPTVC     ;SETUP FOR NEXT INTERRUPT
                                        BR      5$              ;LEAVE
                                4$:     MOV     #TEOM,@TXDBUF   ;END OF MSG--OUTPUT CRC
                                        BIC     #SEND!TXINTE,@TXCSR     ;TRUN OFF THE
                                        MOV     #NO.BTRAP,@DUPTVC       ;TRANSMITTER AND TXINTEN
                                5$:     MOV     #1$,(SP)        ;SETUP TO RETURN
                                        RTI                     ;RETURN

                                ;RECEIVER
                                6$:     MOV     @RXDBUF,DATA    ;GET THE DATA
                                        INC     R1              ;CHECK FOR LAST CHAR
                                        CMP     #4,R1           ;AND BRANCH IF
                                        BNE     10$             ;NOT YET
                                        BIT     #CRCERR,DATA    ;CHECK FOR CRC ERROR
                                        BEQ     7$              ;BR IF CRC ERROR SEEN
                                        HLT     14              ;FAILED TO CATCH CRC ERROR!!!!!
                                7$:     MOV     #2$,(SP)        ;FINISH TEST
                                10$:    RTI                     ;RETURN


                                        ;************************** TEST 27 ****************************
                                        ;*THIS TEST PROVES THE DEVICE WILL HANDLE THE
                                        ;*DDCMP PROTOCALL.  SEND AND RECEIVE SYNCS,
                                        ;*FOLLOWED BY DATA,BCC,DATA AND FINAL BCC.
                                        ;:************************************************************

                                ;:************
                                ;            *
                                ; TEST 27
                                ;            *
                                ;:************
                                ;:*************************************************************************
                                TST27:  MOV     #27,@#TSTNO
                                        MOV     #TST30,NEXT
                                        MOV     #340,PS         ;RAISE PROCESSOR STATUS
                                        JSR     R5,SETVEC       ;SET UP VECTORS
                                        10$                     ;BASED ON
                                        2$                      ;THESE
                                        .BYTE   340,340         ;PARAMETERS
                                        CLR     TEMP1
                                        CLR     TEMP2
                                        CLR     TEMP3
                                        CLR     TEMP4
```

```
2876  020452  005037  001246                    CLR     TEMP5
2877
2878  020456  052777  000400  160726            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2879  020464  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2880  020470  012777  100026  160712            MOV     #DECMOD!26,@PARCSR      ;LOAD THE MODE AND SYNC CHARACTER
2881  020476  052777  004000  160706            BIS     #SYSTST,@TXCSR          ;ENTER SYSTEM TEST MODE
2882  020504  052777  000420  160672            BIS     #RCVEN!STPSYN,@RXCSR    ;LOAD RCVEN!STPSYN
2883  020512  052777  000020  160672            BIS     #SEND,@TXCSR            ;TURN ON TRANSMITTER
2884  020520  012777  000426  160666            MOV     #TSOM!26,@TXDBUF        ;OUTPUT A SYNC CHAR
2885  020526  105777  160660          64$:      TSTB    @TXCSR          ;CHECK DONE
2886  020532  100375                            BPL     64$             ;BR IF NOT SET
2887  020534  012777  000426  160652            MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2888  020542  105777  160644          65$:      TSTB    @TXCSR          ;CHECK DONE
2889  020546  100375                            BPL     65$             ;BR IF NOT SET
2890  020550  012777  000426  160636            MOV     #TSOM!26,@TXDBUF    ;SEND SYNC
2891  020556  052777  000100  160620            BIS     #RINTEN,@RXCSR  ;TURN ON INTERRUPTS
2892  020564  052777  000100  160620            BIS     #TXINTE,@TXCSR  ;DITTO
2893  020572  005037  177776                    CLR     PS              ;LOWER PROCESSOR STATUS
2894  020576                          100$:
2895  020576  012737  000144  020626            MOV     #100.,70$               ;LOAD THE NUMBER
2896  020604  032777  004000  160602  68$:      BIT     #TIMER,@TXDBUF  ;CHECK THE TIMER BIT
2897  020612  001374                            BNE     68$             ;BR IF SET
2898  020614  032777  004000  160572  69$:      BIT     #TIMER,@TXDBUF  ;CHECK THE BIT
2899  020622  001774                            BEQ     69$             ;BR IF CLEAR
2900  020624  005327                            DEC     (PC)+           ;DECREMENT THE NUMBER
2901  020626  000144                  70$:      100.                    ;OF TIMES TO REPEAT
2902  020630  001365                            BNE     68$             ;BR IF MORE TO GO
2903  020632  104023                            HLT     23              ;FAILED TO FINISH TEST
2904  020634                          1$:
2905  020634  052777  000400  160550            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
2906  020642  004737  004772                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
2907  020646  012706  001150                    MOV     #STACK,SP       ;RESET THE STACK
2908  020652  104400                            SCOPE                   ;SCOPE THIS TEST
2909
2910                                  ;INTERRUPT SERVICE ROUTINES
2911                                          ;TRANSMITTER
2912
2913  020654  012777  000252  160532  2$:      MOV     #252,@TXDBUF    ;LOAD FIRST DATA CHAR
2914  020662  012737  000026  001236            MOV     #26,TEMP1               ;LOAD DATA
2915  020670  012777  020700  160502            MOV     #3$,@DUPTVC     ;RELOAD VECTOR
2916  020676  000452                            BR      7$              ;LEAVE
2917  020700  013777  001236  160506  3$:      MOV     TEMP1,@TXDBUF   ;MOV DATA TO BUFFER
2918  020706  105237  001236                    INCB    TEMP1           ;UPDATE DATA
2919  020712  122737  000032  001236            CMPB    #32,TEMP1               ;CHECK FOR DONE
2920  020720  001041                            BNE     7$              ;BR IF MORE TO SEND
2921  020722  012777  020732  160450            MOV     #4$,@DUPTVC     ;RELOAD VECTOR
2922  020730  000435                            BR      7$              ;RETURN
2923  020732  012777  001000  160454  4$:      MOV     #TEOM,@TXDBUF   ;PUT OUT BCC
2924  020740  012777  020750  160432            MOV     #5$,@DUPTVC     ;RELOAD VECTOR
2925  020746  000426                            BR      7$              ;RETURN
2926  020750  013777  001240  160436  5$:      MOV     TEMP2,@TXDBUF   ;LOAD DATA
2927  020756  105237  001240                    INCB    TEMP2           ;UPDATE DATA
2928  020762  122737  000100  001240            CMPB    #100,TEMP2      ;CHECK FOR FINISH
2929  020770  001015                            BNE     7$              ;BR IF MORE TO GO
2930  020772  012777  021002  160400            MOV     #6$,@DUPTVC     ;RELOAD VECTOR
2931  021000  000411                            BR      7$              ;RETURN
```

```
2932  021002  012777  001000  160404  6$:     MOV     #TEOM,@TXDBUF   ;PUSH OUT DATA BCC
2933  021010  042777  000120  160374          BIC     #SEND!TXINTE,@TXCSR     ;SHUT DOWN TRANSMITTER
2934  021016  012777  006722  160354          MOV     #NO.BTRAP,@DUPTVC       ;RESET VECTOR
2935  021024  012716  020576          7$:     MOV     #100$,(SP)      ;SETUP RETURN
2936  021030  000002                          RTI                     ;RETURN
2937
2938                                          ;RECEIVER
2939
2940  021032  017737  160346  001242  10$:     MOV     @RXCSR,TEMP3    ;SAVE CSR
2941  021040  017737  160342  001244          MOV     @RXDBUF,TEMP4   ;SAVE BUFFER
2942  021046  105737  001242                  TSTB    TEMP3           ;CHECK FOR DONE
2943  021052  100401                          BMI     11$             ;BR IF SET
2944  021054  104024                          HLT     24              ;FALSE INTERRUPT
2945  021056  005737  001244          11$:    TST     TEMP4           ;CHECK FOR ERROR
2946  021062  100001                          BPL     12$             ;BR IF NO ERROR
2947  021064  104026                          HLT     26              ;RECEIVER ERROR
2948  021066  122737  000252  001244  12$:    CMPB    #252,TEMP4      ;CHECK DATA
2949  021074  001401                          BEQ     13$             ;BR IF A MATCH
2950  021076  104022                          HLT     22              ;DATA COMPARE ERROR
2951  021100  012737  000026  001246  13$:    MOV     #26,TEMP5               ;LOAD NEXT EXPECTED
2952  021106  012777  021116  160260          MOV     #14$,@DUPRVC    ;RELOAD VECTOR
2953  021114  000531                          BR      26$             ;LEAVE
2954  021116  017737  160264  001244  14$:    MOV     @RXDBUF,TEMP4   ;GET DATA
2955  021124  005737  001244                  TST     TEMP4           ;CHECK FOR ERROR
2956  021130  100001                          BPL     15$             ;BR IF NO ERROR
2957  021132  104026                          HLT     26              ;DATA ERROR
2958  021134  123737  001246  001244  15$:    CMPB    TEMP5,TEMP4             ;CHECK DATA
2959  021142  001401                          BEQ     16$             ;BR IF A MATCH
2960  021144  104022                          HLT     22              ;DATA COMPARE ERROR
2961  021146  105237  001246          16$:    INCB    TEMP5                   ;UPDATE DATA
2962  021152  122737  000032  001246          CMPB    #32,TEMP5               ;CHECK FOR FIRST PART FINISH
2963  021160  001107                          BNE     26$             ;BR IF MORE TO GO
2964  021162  012777  021172  160204          MOV     #17$,@DUPRVC    ;SET UP NEXT VECTOR
2965  021170  000503                          BR      26$             ;LEAVE
2966  021172  017737  160210  001244  17$:    MOV     @RXDBUF,TEMP4   ;GET THE BUFFER
2967  021200  005737  001244                  TST     TEMP4           ;TEST FOR ERROR
2968  021204  100001                          BPL     .+4             ;BR IF OK
2969  021206  104026                          HLT     26              ;RECEIVER ERROR
2970  021210  012777  021220  160156          MOV     #18$,@DUPRVC    ;RELOAD THE VECTOR
2971  021216  000470                          BR      26$             ;LEAVE
2972  021220  017737  160162  001324  18$:    MOV     @RXDBUF,DATA    ;GET DATA
2973  021226  032737  010000  001324          BIT     #CRCERR,DATA    ;CHECK FOR CRC ERROR
2974  021234  001001                          BNE     19$             ;BR IF OK
2975  021236  104014                          HLT     14              ;CRC ERROR!!!!!!
2976  021240  012777  021254  160126  19$:    MOV     #20$,@DUPRVC    ;SET UP VECTOR
2977  021246  005037  001330                  CLR     MIND            ;SETUP FOR NEXT DATA
2978  021252  000452                          BR      26$             ;LEAVE
2979  021254  017737  160126  001244  20$:    MOV     @RXDBUF,TEMP4   ;GET DATA
2980  021262  005737  001244                  TST     TEMP4           ;CHECK FOR ERROR
2981  021266  100001                          BPL     21$             ;BR IF NO ERROR
2982  021270  104026                          HLT     26              ;RECEIVER ERROR
2983  021272  123737  001330  001244  21$:    CMPB    MIND,TEMP4              ;CHECK DATA
2984  021300  001401                          BEQ     22$             ;BR IF A MATCH
2985  021302  104022                          HLT     22              ;DATA ERROR
2986  021304  105237  001330          22$:    INCB    MIND            ;UPDATE SOFTWARE DATA
2987  021310  122737  000100  001330          CMPB    #100,MIND               ;CHECK FOR FINISH
```

```
2988   021316  001030                        BNE     26$             ;BR IF MORE TO GO
2989   021320  012777  021330  160046        MOV     #23$,@DUPRVC    ;RELOAD FINAL VECTOR
2990   021326  000424                        BR      26$             ;LEAVE
2991   021330  017737  160052  001244  23$:  MOV     @RXDBUF,TEMP4   ;GET DATA
2992   021336  005737  001244                TST     TEMP4           ;CHECK FOR ERROR
2993   021342  100001                        BPL     24$             ;BR IF OK
2994   021344  104026                        HLT     26              ;RECEIVER ERROR ON FIRST OCTET
2995                                                                 ;OF SECOND BCC
2996   021346  105777  160032          24$:  TSTB    @RXCSR          ;TEST DONE
2997   021352  100375                        BPL     24$             ;BR IF NOT SET
2998   021354  017737  160026  001324        MOV     @RXDBUF ,DATA   ;GET SECOND BCC OCTET
2999   021362  032737  010000  001324        BIT     #CRCERR,DATA    ;CHECK FOR BCC ERROR
3000   021370  001001                        BNE     25$             ;BR IF OK
3001   021372  104014                        HLT     14              ;BCC ERROR ON SECOND PART OF MSG
3002   021374  012716  020634          25$:  MOV     #1$,(SP)        ;SETUP TO FINISH TEST
3003   021400  000002                  26$:  RTI                     ;RETURN
3004
3005
3006                                        ;************************** TEST 30 ****************************
3007                                        ;*THIS TEST IS AN AID FOR DEBUGGING CRC
3008                                        ;*ERRORS.  A CHARACTER IS LOADED INTO THE
3009                                        ;*DUP AND PUSHED OUT BIT BY BIT WHILE
3010                                        ;*ALLOWING THE OPERATOR TO MONITOR THE CRC
3011                                        ;*CHARACTER AS IT IS GENERATED.  THE DATA CHARACTER
3012                                        ;*CAN ALSO BE CHANGED BY THE OPERATOR.
3013                                        ;*PUT SW09=1 TO LOCK ON BITS.  TO CONTINUE HIT
3014                                        ;*ANY KEY ON THE TTY. AFTER 16 TIMES PUT DOWN SW09 TO LEAVE
3015                                        ;;*************************************************************
3016                                ;;**********
3017                                ;         *
3018                                ; TEST 30
3019                                ;         *
3020                                ;;**********
3021
3022                                ;;*************************************************************************
3023   021402  012737  000030  001226  TST30:  MOV     #30,@#TSTNO
3024   021410  012737  002764  001216          MOV     #.EOP,NEXT
3025   021416  052777  000400  157766          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3026   021424  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3027   021430  012737  120001  007100          MOV     #CRC16,XPOLY    ;LOAD THE POLYNOMIAL
3028   021436  012737  000125  021604          MOV     #125,3$         ;LOAD DATA TO SOFTWARE BCC-CHANGE CHARACTER HERE
3029   021444  013737  021604  001252          MOV     3$,SAVR1
3030   021452  005037  007104                  CLR     CALBCC          ;CLEAR FOR SOFTWARE BCC
3031   021456  013737  007104  021606          MOV     CALBCC,4$
3032   021464  005037  001242                  CLR     TEMP3
3033   021470  005037  001244                  CLR     TEMP4           ;CLEAR BIT COUNTER
3034   021474  005037  001246                  CLR     TEMP5
3035   021500  012777  100026  157702          MOV     #DECMOD!26,@PARCSR  ;LOAD MODE AND SYNC CHARACTER
3036   021506  052777  014000  157676          BIS     #MMODE,@TXCSR   ;ENTER MAINT MODE-PROGRAM CLOCKING
3037   021514  052777  000420  157662          BIS     #RCVEN!STPSYN,@RXCSR  ;TURN ON RECEIVER
3038   021522  052777  000020  157662          BIS     #SEND,@TXCSR    ;TURN ON TRANSMITTER
3039   021530  012777  000426  157656          MOV     #TSOM!26,@TXDBUF  ;LOAD A SYNC
3040   021536  104412  000044                  PKCLK   ,36.            ;PUSH OUT 2 SYNCS
3041   021542  013777  021604  157644          MOV     3$,@TXDBUF      ;LOAD DATA
3042   021550  104412  000020                  PKCLK   ,16.            ;PUSH OUT ANOTHER SYNC
3043   021554  104412  000002          1$:     PKCLK   ,2              ;PUSH OUT A BIT
```

```
3044  021560  013737  001244  001254          MOV     TEMP4,SAVR2     ;SET UP TO TYPE
3045  021566  005237  001242                  INC     TEMP3
3046  021572  005237  001244                  INC     TEMP4           ;UPDATE BIT COUNTER
3047  021576  004537  006726          2$:     JSR     R5,SIMBCC       ;CALCULATE SOFTWARE BCC BASED ON THESE PARAMETERS
3048  021602  000001                          1                       ;SHIFTS
3049  021604  000000                  3$:     .WORD 0                 ;DATA
3050  021606  000000                  4$:     .WORD 0                 ;PREVIOUS BCC
3051  021610  004737  021706                  JSR     PC,5$           ;CHECK TO SEE IF WE SHOULD WAIT FOR SCOPING
3052  021614  000241                          CLC                     ;CLEAR FOR NEXT ROTATE
3053  021616  106037  021604                  RORB    3$              ;SET UP THE NEXT BIT
3054  021622  013737  007104  021606          MOV     CALBCC,4$       ;FOR THE SOFTWARE BCC
3055  021630  022737  000006  001244          CMP     #6,TEMP4
3056  021636  001002                          BNE     .+6
3057  021640  005077  157550                  CLR     @TXDBUF
3058  021644  022737  000014  001242          CMP     #12.,TEMP3
3059  021652  001003                          BNE     12$
3060  021654  012777  001000  157532          MOV     #TEOM,@TXDBUF
3061  021662  022737  000020  001244  12$:    CMP     #16.,TEMP4      ;ALL DONE WITH THE CHARACTER?
3062                                                                  ;INCREASE COMPARE # TO FORCE
3063                                                                  ;CRC OUT OF THE GENERATOR
3064  021670  001331                          BNE     1$              ;BR IF MORE TO GO
3065  021672  052777  000400  157512          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3066  021700  004737  004772                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3067  021704  104400                          SCOPE                   ;SCOPE THIS TEST
3068
3069  021706  032777  001000  157266  5$:     BIT     #SW09,@SWR      ;SW09=1?
3070  021714  001432                          BEQ     6$              ;BR IF NO
3071  021716  013704  007104                  MOV     CALBCC,R4       ;THE DATA CHARACTER IS ALWAYS
3072  021722  012737  000001  001256          MOV     #1,SAVR3        ;FOLLOWED BY A ZERO. THE DATA IN
3073  021730  000241                          CLC                     ;CRC SHOWS WHICH BIT OF THE 2 CHARS
3074  021732  006004                  11$:    ROR     R4              ;IS BEING GENERATED
3075  021734  006137  001256                  ROL     SAVR3
3076  021740  103374                          BCC     11$
3077  021742  105737  001246                  TSTB    TEMP5
3078  021746  001006                          BNE     10$
3079  021750  104402  023100                  TYPE    ,EM17           ;TYPE MSG
3080  021754  104402  023127                  TYPE    ,MH1            ;TYPE HEADER
3081  021760  105137  001246                  COMB    TEMP5
3082  021764  104410                  10$:    CONVRT
3083  021766  023450                          DT1
3084  021770  105777  157210          7$:     TSTB    @TKCSR          ;CHECK TTY DONE--GO SCOPE THE CRC GENERATOR
3085  021774  100375                          BPL     7$              ;BR IF NOT YET
3086  021776  017701  157204                  MOV     @TKDBR,R1       ;READ THE BUFFER
3087  022002  000207                  6$:     RTS     PC              ;RETURN
3088
3089
```

```
 3090
  (1)  022004  042377  050125  043040  EM1:    .ASCIZ  <377>/DUP FAILED TO INTERRUPT IN TIME /
  (1)  022046  042377  052101  020101  EM2:    .ASCIZ  <377>/DATA COMPARE ERROR /
  (1)  022073     377  047105  020104  EM3:    .ASCIZ  <377>/END OF MESSAGE /
  (1)  022114  041377  041503  042440  EM4:    .ASCIZ  <377>/BCC ERROR /
  (1)  022130  052377  040522  051516  EM5:    .ASCIZ  <377>/TRANSMITTER DONE /
  (1)  022153     106  044501  042514  EM6:    .ASCIZ  /FAILED TO SET /
  (1)  022172  051377  041505  044505  EM7:    .ASCIZ  <377>/RECEIVER INTERRUPT IN HALF-DUPLEX /
  (1)  022236  043377  046101  042523  EM10:   .ASCIZ  <377>/FALSE INTERRUPT /
  (1)  022260  040777  047502  052122  EM11:   .ASCIZ  <377>/ABORT SEQUENCE ERROR /
  (1)  022307     377  052123  051101  EM12:   .ASCIZ  <377>/START OF MESSAGE /
  (1)  022332  052777  042516  050130  EM13:   .ASCIZ  <377>/UNEXPECTED RECEIVER INTERRUPT /
  (1)  022372  052777  042516  050130  EM14:   .ASCIZ  <377>/UNEXPECTED TRANSMITTER INTERRUPT /
  (1)  022435     377  051124  047101  EM20:   .ASCIZ  <377>/TRANSMITTER DONE /
  (1)  022460  051377  041505  044505  EM21:   .ASCIZ  <377>/RECEIVER DONE /
  (1)  022500  052377  040522  051516  EM22:   .ASCIZ  <377>/TRANSMITTER ACTIVE /
  (1)  022525     377  042522  042503  EM23:   .ASCIZ  <377>/RECEIVER ACTIVE /
  (1)  022547     106  044501  042514  EM24:   .ASCIZ  /FAILED TO SET. /
  (1)  022567     106  044501  042514  EM25:   .ASCIZ  /FAILED TO CLEAR. /
  (1)  022611     377  054523  041516  EM26:   .ASCIZ  <377>/SYNC ERROR /
  (1)  022626  042377  052101  020101  EM27:   .ASCIZ  <377>/DATA ERROR /
  (1)  022643     377  042504  044526  EM30:   .ASCIZ  <377>/DEVICE FAILED TO INTERRUPT IN TIME /
  (1)  022710  043377  046101  042523  EM31:   .ASCIZ  <377>/FALSE INTERRUPT /
  (1)  022732  052377  040522  051516  EM32:   .ASCIZ  <377>/TRANSMITTER BCC ERROR IN DEC MODE  /
  (1)  022777     377  042522  042503  EM33:   .ASCIZ  <377>/RECEIVER BCC ERROR IN DEC MODE  /
  (1)  023041     377  042522  042503  EM15:   .ASCIZ  <377>/RECEIVER ERROR /
  (1)  023062  041377  041503  042440  EM16:   .ASCIZ  <377>/BCC ERROR!! /
  (1)  023100  041777  041522  043440  EM17:   .ASCIZ  <377>/CRC GENERATOR STATUS /
  (1)  023127     377  040504  040524  MH1:    .ASCIZ  <377>/DATA CHAR     DATA BIT IN CRC GEN.     CRC FOR THIS BIT  /
  (1)                                  .EVEN
  (1)
  (1)  023222                          .ERRTAB:
  (1)  023222  000000                          0
  (1)  023224  000000                          0
  (1)  023226  000000                          0
  (1)  023230  022004                          EM1
  (1)  023232  000000                          0                       ;HALT 1
  (1)  023234  000000                          0
  (1)
  (1)  023236  022046                          EM2
  (1)  023240  000000                          0                       ;HALT2
  (1)  023242  000000                          0
  (1)
  (1)  023244  022073                          EM3
  (1)  023246  022153                          EM6                     ;HALT 3
  (1)  023250  000000                          0
  (1)
  (1)  023252  022114                          EM4
  (1)  023254  000000                          0                       ;HALT 4
  (1)  023256  000000                          0
  (1)
  (1)  023260  022130                          EM5
  (1)  023262  022153                          EM6                     ;HALT 5
  (1)  023264  000000                          0
  (1)
  (1)  023266  022172                          EM7
```

```
(1)  023270  000000                    0                       ;HALT 6
(1)  023272  000000                    0
(1)
(1)  023274  022236                    EM10
(1)  023276  000000                    0                       ;HALT 7
(1)  023300  000000                    0
(1)
(1)  023302  022260                    EM11
(1)  023304  000000                    0                       ;HALT 10
(1)  023306  000000                    0
(1)
(1)  023310  022307                    EM12
(1)  023312  022153                    EM6                     ;HALT 11
(1)  023314  000000                    0
(1)
(1)  023316  022332                    EM13
(1)  023320  000000                    0                       ;HALT 12
(1)  023322  000000                    0
(1)
(1)  023324  022372                    EM14
(1)  023326  000000                    0                       ;HALT 13
(1)  023330  000000                    0
(1)
(1)  023332  023062                    EM16
(1)  023334  000000                    0       ;HALT14
(1)  023336  000000                    0
(1)
(1)  023340  022460                    EM21
(1)  023342  022547                    EM24    ;HALT15
(1)  023344  000000                    0
(1)
(1)
(1)  023346  022435                    EM20
(1)  023350  022547                    EM24    ;HALT16
(1)  023352  000000                    0
(1)
(1)  023354  022500                    EM22
(1)  023356  022547                    EM24    ;HALT17
(1)  023360  000000                    0
(1)
(1)  023362  022500                    EM22
(1)  023364  022567                    EM25    ;HALT20
(1)  023366  000000                    0
(1)
(1)  023370  022611                    EM26
(1)  023372  000000                    0       ;HALT21
(1)  023374  000000                    0
(1)
(1)  023376  022626                    EM27
(1)  023400  000000                    0       ;HALT22
(1)  023402  000000                    0
(1)
(1)  023404  022643                    EM30
(1)  023406  000000                    0       ;HALT23
(1)  023410  000000                    0
(1)
```

```
   (1)  023412  022710                          EM31
   (1)  023414  000000                          0          ;HALT24
   (1)  023416  000000                          0
   (1)
   (1)  023420  022525                          EM23
   (1)  023422  022547                          EM24       ;HALT25
   (1)  023424  000000                          0
   (1)
   (1)  023426  023041                          EM15
   (1)  023430  000000                          0          ;HALT26
   (1)  023432  000000                          0
   (1)
   (1)  023434  022732                          EM32
   (1)  023436  000000                          0          ;HALT 27
   (1)  023440  000000                          0
   (1)
   (1)  023442  022777                          EM33
   (1)  023444  000000                          0          ;HALT   30
   (1)  023446  000000                          0
   (1)  023450  000003                  DT1:    3
   (1)  023452     006      021                 .BYTE   6,17.
   (1)  023454  001252                          SAVR1
   (1)  023456     006      017                 .BYTE   6,15.
   (1)  023460  001254                          SAVR2
   (1)  023462     006      002                 .BYTE   6,2
   (1)  023464  001256                          SAVR3
   (1)
   (1)  023466                          CORMAX:
3091            000001                .END
```

```
ADRCNT= 003735   743*   779*   788#
BCCFBK  007102  1162*  1168*  1171*  1174  1178  1190#
BINWRD  004240   830*   831    868#
BITW  = 002000   220#
BIT0  = 000001    82#   202   1166
BIT1  = 000002    81#   201
BIT10 = 002000    72#   192    207   220   240
BIT11 = 004000    71#   191    219   239
BIT12 = 010000    70#   190    206   213   218   238
BIT13 = 020000    69#   189    217   237
BIT14 = 040000    68#   188    205   216   236   502   650
BIT15 = 100000    67#   187    204   211   215   235   499
BIT2  = 000004    80#   200    588
BIT3  = 000010    79#   199    226
BIT4  = 000020    78#   198    225
BIT5  = 000040    77#   197
BIT6  = 000100    76#   196    224
BIT7  = 000200    75#   195    223
BIT8  = 000400    74#   194    209   222   242
BIT9  = 001000    73#   193    208   212   221   241
BRW     003306   594    672#
BRX     003310   595    673#
CALBCC  007104  1184*  1191#  1519*  1520  1614*  1615  1645*  1646  1740*  1741  1969*  1970  2024*
                2025   2541*  2542   2627  3030*  3031  3054   3071
CARDET= 010000   190#
CHRCNT  004236   828*   832    848*   866#  867
CLK   = 020000   217#   987    990
CLK.A   001433   328#  1103
CLRVEC  005032  1012#
CNVRT = 104411   295#   611    613    615   617   941   943
CONVRT= 104410   293#   547    957   3082
CORMAX  023466  3090#  3091
CRCEN = 001000   212#  1428   1845   2061  2101  2144  2167  2214  2262  2324  2437  2733
CRCERR= 010000   206#  1618   1744   2635  2846  2973  2999
CRC.CC= 102010  1193#  1518   1644   1968
CRC16 = 120001  1192#  2540   3027
CREAM   001316   182#   409*  1057*  1058  1060*  1065  1066*  1067  1070*
CTS   = 020000   189#
CYCLE   006152   596    631    632   1047#
DATA    001324   249#  1466*  1467   1573*  1574  1606*  1610*  1615  1699*  1700  1732*  1736*  1741
                2010*  2011   2020*  2023*  2025  2698*  2699  2707*  2708  2842*  2846  2972*  2973
                2998*  2999
DATABP  004612   930*   933    955    958#
DATAHD  004600   929*   951    954
DECMOD= 100000   211#  2061   2101   2144  2167  2214  2262  2324  2437  2559  2689  2733  2798
                2880   3035
DEVADR  003732   741*   776    786#
DISPLA  001200   126#   422*   608*
DISPRE  000174   114#   422
DSCA  = 100000   187#
DSCB  = 000001   202#
DSINTE= 000040   197#
DSR   = 001000   193#
DTR   = 000002   201#
DT1     023450  3083   3090#
DUPACT  001310   176#   512*   536*   569*  570   1047  1052
```

DZDPDA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
DUPCR0  001500          337#    451     515
DUPCR1  001506          341#    516
DUPCR2  001514          345#
DUPCR3  001522          349#
DUPCR4  001530          353#
DUPCR5  001536          357#
DUPCR6  001544          361#
DUPCR7  001552          365#
DUPNUM  001311          177#    405     504#    537#    621
DUPRPS  001376          307#   1093#   1094#   1095    1145#
DUPRVC  001374          306#    638    1072#   1093    1143#   1896#   1903#   2017#   2379#   2396#   2495#   2618#   2654#
                       2952#   2964#   2970#   2976#   2989#
DUPSEC  001416          315#   1079#
DUPTPS  001402          309#   1097#   1098#   1146#
DUPTR0  001502          338#    459
DUPTR1  001510          342#
DUPTR2  001516          346#
DUPTR3  001524          350#
DUPTR4  001532          354#
DUPTR5  001540          358#
DUPTR6  001546          362#
DUPTR7  001554          366#
DUPTVC  001400          308#   1095#   1096#   1097    1144#   1488#   1491#   1497#   1593#   1596#   1602#   1719#   1722#
                       1728#   1935#   2036#   2042#   2518#   2665#   2833#   2837#   2915#   2921#   2924#   2930#   2934#
DUP.EN  001560          369#    444     528     1058    1067
DUP.MA  001500          182     336#    409     442     526     530     544     1060    1070
DUP0.A  001504          339#    499#    502#
DUP1.A  001512          343#
DUP2.A  001520          347#
DUP3.A  001526          351#
DUP4.A  001534          355#
DUP5.A  001542          359#
DUP6.A  001550          363#
DUP7.A  001556          367#
EM1     022004         3090#
EM10    022236         3090#
EM11    022260         3090#
EM12    022307         3090#
EM13    022332         3090#
EM14    022372         3090#
EM15    023041         3090#
EM16    023062         3090#
EM17    023100         3079    3090#
EM2     022046         3090#
EM20    022435         3090#
EM21    022460         3090#
EM22    022500         3090#
EM23    022525         3090#
EM24    022547         3090#
EM25    022567         3090#
EM26    022611         3090#
EM27    022626         3090#
EM3     022073         3090#
EM30    022643         3090#
EM31    022710         3090#
EM32    022732         3090#
```

```
EM33    022777  3090#
EM4     022114  3090#
EM5     022130  3090#
EM6     022153  3090#
EM7     022172  3090#
ERRCNT  001232   147#   411*   644    969*
ERRFLG  001341   262#   407*   606*   664*   918*   931    945*
ERRMSG  004566   928*   946    949#
ERTABO  004720   943    978#
EXIT  = 000000    83#
EXITER  004654   963    969#
FIX.OO  006670  1104   1138#
FLAG    001332   252#   440*   553    556*
HALTS   004616   914    960#
HDXEN = 000010   226#  1349
HILIM   003730   740*   767    785#
HUPPSR  001420   316#  1083*  1084
HUPRBF  001422   317#  1081*  1082*  1083
HUPRCR  001424   318#  1075*  1076*  1077
HUPTBF  001426   319#  1090*  1091*
HUPTCR  001430   320#  1086*  1087*  1088
ICOUNT  001222   143#   662    667*
INBUF   006106   710    746    877*   878    882    1034#
INIFLG  001340   261#   426    432*   441*
INSTER= 104404   285#   761    887
INSTR = 104403   283#   446    454    462    471    479    483    489    493    1110
INSTR2  003530   717    729#
LIMITS  003656   756    767#
LOBITS  003734   742*   771    787#   788
LOCK    001220   142#   666*   680    682    937
LOKFLG  001342   263#
LOLIM   003726   739*   769    784#
LPCNT   001224   144#   661*   662    665*
LSTERR  001234   148#   412*   605*   648*   915    917*
LOO.OO  001434   330#  1073*  1101
MASK.A  001432   326#  1102
MASTEK  005337   939   1023#
MCRLF   005064   695    819    935    936    944    1023#  1109   1126
MCSR    005421   447   1023#
MCSRX   005267   610   1023#
MDATA   006044   846    856    1032#
MEPASS  005105   609   1023#
MERRPC  005414   942   1023#
MERRX   005314   616   1023#
MERR2   005133  1023#  1049
MERR3   005202   566   1023#
MEXT  = 010000   230#   966    1349   1657
MH1     023127  3080   3090#
MIND    001330   251#  2977*  2983   2986*  2987
MJMPR   005453   480   1023#
MLOCK   005240   590   1023#
MMODE = 014000   229#  2687   3036
MMODEA= 004000   219#
MMODEB= 010000   218#
MNEW    005341   561   1023#
MPAR    005630   463   1023#
```

```
MPASSX  005303            614    1023#
MPOWER  005067           1008    1023#
MQM     005060            727    1023#  1132
MR      005130            597    1023#  1127
MRESET= 000400             83#    222#  1211  1245  1279  1313  1347  1397  1418  1515  1622  1639  1748
                         1764   1817   1837  1923  1962  2003  2059  2081  2099  2142  2165  2194  2212
                         2260   2322   2407  2435  2463  2557  2585  2685  2735  2796  2823  2878  2905
                         3025   3065
MSRJM   005707            494    1023#
MSTJM   005654            490    1023#
MTCN    005526            484    1023#
MTDATA= 040000            216#
MTITLE  001000            121#    431
MTOTAL  005575            472    1023#
MTSTN   005325            940    1023#  1111
MTSTPC  005226           1023#
MVEC    005440            455    1023#
MVECX   005275            612    1023#
NEXT    001216            141#    668    974  1209* 1241* 1275* 1309* 1343* 1417* 1514* 1638* 1763* 1836*
                         1961*  2058*  2098* 2141* 2211* 2259* 2315* 2426* 2537* 2684* 2725* 2788* 2866*
                         3024#
NO.ATR  006716           1148#  1214   1248  1282  1316
NO.BTR  006722           1152#  1215   1249  1283  1352  1497  1602  1728  1768  1866  2042  2319  2518
                         2665   2837   2934
OPCLRJ  001323            248#   482    497   541*
OVRRUN= 040000            205#
PARAM = 104405            287#   448    456   464   473   1112
PARAM1  003576            745#   762
PARBIT= 000000             83#
PARCSR  001410            312#  1080*  1428* 1773* 1845* 1984* 2061* 2101* 2144* 2167* 2214* 2262* 2324*
                         2437*  2559*  2689* 2738* 2798* 2880* 3035*
PARERR  003652            748    750    752   761#  768   770   772
PASCNT  001230            146#   406*   607*  608   641
PC     =%000007            47#   626*   998* 1019* 1104* 1139* 1212* 1246* 1280* 1314* 1348* 1363* 1375*
                         1392*  1398*  1419* 1440* 1455* 1516* 1547* 1562* 1623* 1640* 1673* 1688* 1749*
                         1765*  1792*  1818* 1838* 1858* 1877* 1924* 1963* 1998* 2004* 2060* 2070* 2082*
                         2100*  2117*  2143* 2157* 2166* 2183* 2195* 2213* 2234* 2261* 2281* 2323* 2358*
                         2408*  2436*  2458* 2464* 2558* 2580* 2586* 2686* 2736* 2797* 2818* 2824* 2879*
                         2900*  2906*  3026* 3051* 3066* 3087*
PERFOR= 000000             83#
PKCLK = 104412            297#  2692   2694  2703  3040  3042  3043
POPRO = 012600             61#   968
POP1SP= 005726             59#
POP2SP= 022626             63#   670
PRIRTY  001320            246#   470*   534* 1242  1276  1310
PRISEC= 010000            213#  1773   1984
PS     = 177776             52#   402*   584* 1210* 1218* 1244* 1252* 1278* 1286* 1312* 1320* 1346* 1354*
                         1395*  1422*  1448* 1530* 1555* 1613* 1656* 1681* 1739* 1772* 1776* 1844* 1848*
                         1964*  1991*  2316* 2338* 2427* 2449* 2538* 2571* 2726* 2789* 2809* 2867* 2893*
PUSHRO= 010046             60#   964
PUSH1S= 005746             58#
PUSH2S= 024646             62#
PWRUP   005006           1002   1004#
QV.FLG  001343            264#   408*   620*  659
RABORT= 002000            207#  1813
RCRCIN= 040000            236#
```

```
RCRC7T= 100000           235#
RCVEN = 000020           198#  1355  1429  1536  1662  1774  1846  1868  1902  1985  2146  2168  2216
                        2264  2326  2439  2561  2688  2737  2800  2882  3037
RECACT= 004000           191#  2369  2403  2472  2594  2757
REOM  = 001000           208#  1475
RESREG  004614           956   959#
RESTRT  003116           619   623   631#
RESOS = 104407           291#  959
RETURN  001214           140#  414*  596*  598   631*  668*  671   974*  976   1009  1125*  1135*  1136
RING  = 040000           188#
RINTEN= 000100           196#  1356  1430  1537  1663  1774  1846  1985  2339  2450  2572  2810  2891
RSCM  = 000400           209#  1887  1915
RTS   = 000004           200#
RUN     001314           180#  410*  1052  1055*  1056*  1063*  1064*
RXCSR   001404           310#  635  1071*  1075  1355*  1356*  1429*  1430*  1473  1536*  1537*  1607  1662*
                        1663*  1733  1774*  1799  1810  1846*  1868*  1885  1902*  1905  1913  1985*  2021
                        2146*  2160  2168*  2186  2216*  2237  2264*  2292  2326*  2339*  2364  2381  2398
                        2403  2439*  2450*  2470  2497  2561*  2572*  2592  2620  2623  2644  2688*  2695
                        2704  2737*  2750  2800*  2810*  2882*  2891*  2940  2996  3037*
RXDBUF  001406           311#  1077*  1078*  1079  1080  1081  1466  1475  1573  1606  1609  1699  1732
                        1735  1800  1812  1886  1898  1906  1914  2010  2020  2023  2189  2240  2294
                        2365  2382  2399  2471  2498  2593  2622  2625  2645  2698  2707  2756  2842
                        2941  2954  2966  2972  2979  2991  2998
RXDERR= 100000           204#
RXDONE= 000200           195#  1801  1890  2366  2383  2477  2599
R0    =%000000            40#  442*  443*  444  515*  520  522  524  526*  527*  528  530*  531*
                         532*  533*  535*  536  537  538  539  540  541  562*  570*  571*  573*
                         575*  577  578  649  655*  669*  802  807*  820  833*  837*  847  863*
                         965*  984*  988*  991*  1065*  1071  1072  1073  1074*  1094  1096  1098  1101*
                        1118*  1119  1121  1123  1125  1129  1130  1138  1156  1163*  1165*  1166  1169*
                        1170*  1171  1174*  1177*  1179*  1181  1187*  1770*  1783*  1784  1839*  1932*  1933
                        1936*  1942*  1943  1965*  1976*  1977  1979*  2014*  2015  2364*  2366  2369  2381*
                        2383  2398*  2428*  2511  2512*  2539*  2543  2548*  2550*  2658  2659*  2730*  2752*
                        2753  2768*  2831*  2832
R1    =%000001            41#  516*  520*  521*  522*  523*  524*  574*  575  576*  577  622*  626
                         801  808*  821  825*  827  828  829  830  862*  1013*  1014  1016  1017
                        1102*  1157  1175*  1176  1177  1186*  1517*  1521  1526*  1643*  1647  1652*  1799*
                        1801  1967*  1971  2011  2365*  2373  2376  2382*  2386  2389  2391  2399*  2400
                        2429*  2487  2492*  2502  2505*  2551*  2609  2614*  2649  2652*  2731*  2769*  2770
                        2771  2794*  2843*  2844  3086*
R2    =%000002            42#  800  809*  1012*  1014*  1015*  1103*  1158  1176*  1178*  1179  1185*  1800*
                        1804  1812*  1813  1966*  2033*  2034  2470*  2472  2477  2497*  2592*  2594  2599
                        2644*  2727*  2743*  2767*
R3    =%000003            43#  704  711*  721*  724*  725  730*  799  810*  822  834*  835*  836*
                         837  846*  847*  852*  855*  861*  1138*  1771*  1807*  1808  2471*  2482  2487
                        2498*  2499  2502  2593*  2604  2609  2645*  2646  2649
R4    =%000004            44#  705  710*  714*  715*  716  723*  726  729*  737  746*  747  749
                         751  753*  754  755  776*  777*  781*  798  811*  823  831*  834  839*
                         841*  843*  860*  922*  923*  924*  925*  926*  927*  928  929  930  1885*
                        1890  1905*  1913*  2073*  2074  2077  3071*  3074*
R5    =%000005            45#  688  689*  693  698  700*  736  738*  739  740  741  742  743
                         744  745*  754*  757*  758*  759*  767  769  771  777  778*  782*  797
                         812*  824  832*  844*  859*  876*  880*  884*  920*  921*  922  924  1143
                        1144  1145  1146  1147*  1159  1160  1161  1188*  1213*  1247*  1281*  1315*  1350*
                        1424*  1522*  1532*  1580*  1648*  1658*  1706*  1766*  1840*  1864*  1886*  1887  1893
                        1898*  1899  1906*  1907  1914*  1915  1918  1972*  1980*  2317*  2430*  2544*  2552*
```

```
                    2790*   2868*   3047*
SAVACT  001312       178#    503*    509*    511     512     538*    564
SAVNUM  001313       179#    405*    505*    539*    618*    621*
SAVPC   001266       165#    793*    980
SAVPCA  001306       174#
SAVRO   001250       158#    802*    807
SAVRDA  001270       167#
SAVR1   001252       159#    801*    808    3029*   3090
SAVR1A  001272       168#
SAVR2   001254       160#    800*    809    3044*   3090
SAVR2A  001274       169#
SAVR3   001256       161#    799*    810    3072*   3075*   3090
SAVR3A  001276       170#
SAVR4   001260       162#    798*    811
SAVR4A  001300       171#
SAVR5   001262       163#    797*    812
SAVR5A  001302       172#
SAVSP   001264       164#
SAVSPA  001304       173#
SAVOS = 104406       289#    919
SCOPE = 104400       277#   1223    1257    1291    1326    1399    1459    1566    1692    1797    1863    2005    2083
                    2126    2196    2244    2290    2410    2466    2588    2711    2773    2826    2908    3067
SCOP1 = 104401       279#
SEND  = 000020       225#   1357    1433    1496    1540    1601    1666    1727    1775    1847    1986    2041    2063
                    2103    2111    2147    2170    2217    2265    2327    2352    2440    2517    2562    2664    2690
                    2702    2739    2801    2836    2883    2933    3038
SETFLG= 104413       299#    481     485     491     495
SETVEC  006674      1143#   1213    1247    1281    1315    1350    1424    1532    1580    1658    1706    1766    1840
                    1864    1980    2317    2430    2552    2790    2868
SWIFTS  001326       250#
SIMBCC  006726      1156#   1522    1648    1972    2544    3047
SMALL   004772       997#   1212    1246    1280    1314    1348    1398    1419    1516    1623    1640    1749    1765
                    1818    1838    1924    1963    2004    2060    2082    2100    2143    2166    2195    2213    2261
                    2323    2408    2436    2464    2558    2586    2686    2736    2797    2824    2879    2906    3026
                    3066
SP     =%000006      46#    403*    416*    417*    423     424     425     585*    649*    669     682*    688*    689
                     690*    700     704*    705*    706     707*    723     724     725*    726*    729     730     736*
                     737*    738     744*    781     782     793     820*    821*    822*    823*    824*    825     826*
                     859     860     861     862     863     876     885*    896*    897*    898*    899*    900*    901*
                     902*    903     915     917     920     965     975*    984     985*   1005*   1156*   1157*   1156*
                    1185    1186    1187   1224*   1258*   1292*   1327*   1396*   1402*   1478*   1498*   1603*   1611*
                    1729*   1737*   1796*   1816*   1862*   1925*   1947*   2028*   2043*   2409*   2465*   2475*   2480*
                    2485*   2490*   2494*   2507*   2519*   2587*   2597*   2602*   2607*   2612*   2616*   2642*   2666*
                    2825*   2838*   2849*   2907*   2935*   3002*
SPACNT= 004237       829*    850     853*    867#
SRD   = 002000       192#
SRJMFL  001336       254#    496
STACK = 001150        53#    403     585     975    1005    1396    1796    1862    2409    2465    2587    2825    2907
STD   = 000010       199#
STJMFL  001334       253#    492
STPSYN= 000400       194#   2326    2882    3037
SVOS    003744       797#
SWR     001202       127#    419    421*    433     437     551     559     564     569     588     650     657     678
                     691     908     913     962     970     972    1107    3069
SWREG   000176       115#    421
SWOO  = 000001        34#    437     551
```

```
SW01   = 000002              33#   1107
SW02   = 000004              32#
SW03   = 000010              30#    559
SW04   = 000020              29#
SW05   = 000040              28#
SW06   = 000100              27#
SW07   = 000200              26#
SW08   = 000400              25#    970
SW09   = 001000              24#    678   3069
SW10   = 002000              23#    972
SW11   = 004000              22#    657
SW12   = 010000              21#    691    908
SW13   = 020000              20#    913
SW14   = 040000              19#
SW15   = 100000              18#
SYSTST= 004000              231#    966   1217   1251   1285   1319   1423   1531   1775   1847   1986   2062   2102
                           2145    2169   2215   2263   2325   2438   2560   2739   2799   2881
TABORT= 002000              240#   1786
TCNFLG  001322              247#    486    487    500    540*  1344   1641
TCRCIN= 010000              238#
TCRC7T= 020000              237#
TEMP    006002              833   1030#
TEMP1   001236              153#    476    504    505    506*   544*   545    549*  1025   1159*  1182*  1420*  1484
                           1485*   1486   1528*  1589   1590*  1591   1654*  1715   1716*  1717   2189*  2190   2240*
                           2241    2273*  2275*  2288*  2622*  2626*  2627   2728*  2732*  2742   2763   2770*  2872*
                           2914*   2917   2918*  2919
TEMP2   001240              154#    467    470    511*   514*   518*   545*  1027   1160*  1164*  1421*  1467   1470*
                           1471    1529*  1574   1577*  1578   1655*  1700   1703*  1704   2274*  2294*  2295   2625*
                           2626    2635   2729*  2733*  2738   2771*  2873*  2926   2927*  2928
TEMP3   001242              155#   1161*  1163   1173*  1175   1180*  1181*  1184   1609*  1610   1618   1735*  1736
                           1744    2757   2874*  2940*  2942   3032*  3045*  3058
TEMP4   001244              156#   2756*  2760   2763   2875*  2941*  2945   2948   2954*  2955   2958   2966*  2967
                           2979*   2980   2983   2991*  2992   3033*  3044   3046*  3055   3061
TEMP5   001246              157#   2876*  2951*  2958   2961*  2962   3034*  3077   3081*
TEOM   = 001000              241#   1386   1493   1598   1724   1945   2038   2110   2351   2516   2663   2835   2923
                           2932    3060
TIMER  = 004000              239#   1359   1361   1371   1373   1388   1390   1436   1438   1451   1453   1543   1545
                           1558    1560   1669   1671   1684   1686   1788   1790   1854   1856   1873   1875   1994
                           1996    2066   2068   2113   2115   2153   2155   2179   2181   2219   2221   2230   2232
                           2277    2279   2354   2356   2442   2444   2454   2456   2564   2566   2576   2578   2746
                           2748    2814   2816   2896   2898
TKCSR   001204              132#    653    712   3084
TKDBR   001206              133#    655    714    720   3086
TLAST  = 021402             1130   3090#
TPCSR   001210              134#    696    718    910
TPDBR   001212              135#    698*   720*   912*
TRPOK   004330              899#
TRP.PC  006150             1036#
TSOM   = 000400              242#   1369   1434   1541   1667   1779   1851   1945   1989   2064   2104   2107   2148
                           2171    2174   2218   2225   2228   2266   2269   2272   2287   2328   2331   2334   2337
                           2441    2448   2563   2570   2691   2732   2802   2805   2808   2884   2887   2890   3039
TSTNO   001226              145#    413*   983   1115   1121   1123   1208*  1240*  1274*  1308*  1342*  1416*  1513*
                           1637*   1762*  1835*  1960*  2057*  2097*  2140*  2210*  2258*  2314*  2425*  2536*  2683*
                           2724*   2787*  2865*  3023*
TST1    007106             1118   1135   1208#
TST10   011360             1514   1637#
```

```
TST11   012166            1638    1762#
TST12   012502            1763    1835#
TST13   013272            1836    1960#
TST14   013732            1961    2057#
TST15   014104            2058    2097#
TST16   014300            2098    2140#
TST17   014646            2141    2210#
TST2    007212            1209    1240#
TST20   015070            2211    2258#
TST21   015334            2259    2314#
TST22   016054            2315    2425#
TST23   016536            2426    2536#
TST24   017360            2537    2683#
TST25   017556            2684    2724#
TST26   020036            2725    2787#
TST27   020376            2788    2865#
TST3    007326            1241    1274#
TST30   021402            2866    3023#   3090
TST31 = ****** U          3024
TST4    007442            1275    1308#
TST5    007560            1309    1342#
TST6    010120            1343    1416#
TST7    010560            1417    1513#
TTST    003174             591*    592*    594*    595*    651#
TWOSYN= 000000              83#
TXACT = 001000             221#   2077    2123
TXCSR   001412             313#    966*    987*    990*   1084*   1085*   1086    1211*   1217*   1222*   1245*   1251*   1256*
                          1279*   1285*   1290*   1313*   1319*   1325*   1347*   1349*   1357*   1366    1378    1384    1397*
                          1418*   1423*   1431    1433*   1443    1447*   1496*   1515*   1531*   1538    1540*   1550    1554*
                          1601*   1622*   1639*   1657*   1664    1666*   1676    1680*   1727*   1748*   1764*   1775*   1777
                          1780    1817*   1837*   1847*   1849    1852*   1869    1923*   1928    1938    1946*   1962*   1986*
                          1987    1990*   2003*   2041*   2059*   2062*   2063*   2073    2081*   2099*   2102*   2103*   2105
                          2108    2111*   2120    2123    2142*   2145*   2147*   2149    2165*   2169*   2170*   2172    2175
                          2194*   2212*   2215*   2217*   2223    2226    2260*   2263*   2265*   2267    2270    2284    2322*
                          2325*   2327*   2329    2332    2335    2340    2343    2346    2349    2352*   2407*   2435*   2438*
                          2440*   2446    2451*   2463*   2514    2517*   2557*   2560*   2562*   2568    2573*   2585*   2661
                          2664*   2685*   2687*   2690*   2702*   2735*   2739*   2740    2796*   2799*   2801*   2803    2806
                          2811*   2823*   2836*   2878*   2881*   2883*   2885    2888    2892*   2905*   2933*   3025*   3036*
                          3038*   3065*
TXDBUF  001414             314#   1088*   1089*   1090    1359    1361    1369*   1371    1373    1383*   1386*   1388    1390
                          1434*   1436    1438    1446*   1451    1453    1484*   1490*   1493*   1541*   1543    1545    1553*
                          1558    1560    1589*   1595*   1598*   1667*   1669    1671    1679*   1684    1686    1715*   1721*
                          1724*   1779*   1782*   1786*   1788    1790    1851*   1854    1856    1871*   1873    1875    1931*
                          1941*   1945*   1989*   1994    1996    2032*   2038*   2064*   2066    2068    2104*   2107*   2110*
                          2113    2115    2148*   2151*   2153    2155    2171*   2174*   2177*   2179    2181    2218*   2219
                          2221    2225*   2228*   2230    2232    2266*   2269*   2272*   2277    2279    2287*   2328*   2331*
                          2334*   2337*   2342*   2345*   2348*   2351*   2354    2356    2441*   2442    2444    2448*   2454
                          2456    2511*   2516*   2563*   2564    2566    2570*   2576    2578    2658*   2663*   2691*   2693*
                          2742*   2746    2748    2802*   2805*   2808*   2814    2816    2832*   2835*   2884*   2887*   2890*
                          2896    2898    2913*   2917*   2923*   2926*   2932*   3039*   3041*   3057*   3060*
TXDLAT= 100000             215#
TXDONE= 000200             223#   2074
TXINTE= 000100             224#   1217    1251    1285    1319    1447    1496    1554    1601    1680    1727    1852    1946
                          1990    2041    2451    2517    2573    2664    2811    2836    2892    2933
TYPDAT  004602             934     952     955#
TYPE  = 104402             281#    431     543     561     566     590     597     609     610     612     614     616     695
```

```
                    708    727    819    856    935    936    939    940    942    944    948    953   1007
                   1049   1109   1126   1127   1132   3079   3080
TYPMSG  004502      932    935#
USER  = 000000      228#
WRDCNT  004234      827*   857*   865#
WRKO.F  004570      947    950#
XBX     004374      909    911    913#
XCSR    003130      611    633#
XERR    003152      617    642#
XHEAD   005742      543   1023#
XPASS   003144      615    639#
XPOLY   007100     1169   1180   1189#  1518*  1644*  1968*  2540*  3027*
XSTATQ  005770      548   1023#
XTSTN   004726      941    981#
XVEC    003136      613    636#
$E    = 000032        1#  1209   1210#  1241   1242#  1275   1276#  1309   1310#  1343   1344#  1417   1418#
                   1514   1515#  1638   1639#  1763   1764#  1836   1837#  1961   1962#  2058   2059#  2098
                   2099#  2141   2142#  2211   2212#  2259   2260#  2315   2316#  2426   2427#  2537   2538#
                   2684   2685#  2725   2726#  2788   2789#  2866   2867#  3024   3025#
$ENDAD  003104      109    429    625#   960
$N    = 000030        1#  1196   1202   1210#  1228   1234   1242#  1262   1268   1276#  1296   1302   1310#
                   1330   1336   1344#  1405   1410   1418#  1502   1507   1515#  1626   1631   1639#  1752
                   1756   1764#  1822   1829   1837#  1950   1954   1962#  2046   2051   2059#  2086   2091
                   2099#  2129   2134   2142#  2199   2204   2212#  2247   2252   2260#  2301   2308   2316#
                   2414   2419   2427#  2524   2530   2538#  2672   2677   2685#  2714   2718   2726#  2777
                   2781   2789#  2853   2859   2867#  3007   3017   3025#  3090#
$RAYO = 177777      1#  1196#  1197   1200#  1228#  1229   1232#  1262#  1263   1266#  1296#  1297   1300#
                   1330#  1331   1334#  1405#  1406   1408#  1502#  1503   1505#  1626#  1627   1629#  1752#
                   1753   1755#  1822#  1823   1827#  1950#  1951   1953#  2046#  2047   2049#  2086#  2087
                   2089#  2129#  2130   2132#  2199#  2200   2202#  2247#  2248   2250#  2301#  2302   2307#
                   2414#  2415   2418#  2524#  2525   2529#  2672#  2673   2675#  2714#  2715   2717#  2777#
                   2778   2780#  2853#  2854   2857#  3007#  3008   3016#
$Y    = 000014    268#   277    279#   281#   283#   285#   287#   289#   291#   293#   295#   297#   299#
                   301#
.     = 023466     94#    95     98#   105#   110#   113#   116#   120#   122#   176#   177#   178#   179#
                   180#   181#   247#   248#   253#   254#   335#   337#   338#   339#   341#   342#   343#
                   345#   346#   347#   349#   350#   351#   353#   354#   355#   357#   358#   359#   361#
                   362#   363#   365#   366#   367#   438    568   1031#  1033#  1035#  1036#  1051   1385
                  1468   1476   1524#  1525#  1575   1608   1616   1619   1650#  1651#  1701   1734   1742
                  1745   1974#  1975#  2012   2022   2026   2968   3056
.BEGIN  002660      584#
.CNVRT  004034      296    820#
.CONVR  004030      294    819#
.EOP    002764      605#  3024
.ERRTA  023222      927   3090#
.HLT    004350      101    908#
.INSTE  003516      286    725#
.INSTR  003412      284    704#
.INST1  003432      708#   728
.MSG    003434      706*   709#
.PARAM  003536      288    736#
.PFAIL  004776       99    404    1002#  1006
.PKCLK  004734      298    984#
.RESOS  003776      292    807#
.SAVOS  003736      290    793#
.SCOPE  003160      278    648#
```

CROSS REFERENCE TABLE -- USER SYMBOLS

.SCOP1	003312	280	678#	
.SETFL	004242	300	876#	888
.START	001562	117	402#	414
.TRPSR	004316	103	896#	
.TRPTA	001344	276#	901	
.TYPE	003336	282	688#	

```
DUPEND      1#    599
DUPFRN      1#
HLT        64#   1149   1153   1324   1368   1380   1401   1445   1458   1469   1477   1552   1565   1576   1617
         1620    1678   1691   1702   1743   1746   1795   1803   1806   1815   1861   1880   1889   1892   1895
         1901    1909   1911   1917   1920   1930   1940   2001   2013   2027   2076   2079   2122   2125   2162
         2188    2192   2239   2243   2286   2297   2361   2368   2371   2375   2378   2385   2388   2393   2395
         2402    2405   2461   2474   2479   2484   2489   2501   2504   2583   2596   2601   2606   2611   2629
         2637    2648   2651   2697   2701   2706   2710   2755   2759   2762   2765   2821   2848   2903   2944
         2947    2950   2957   2960   2969   2975   2982   2985   2994   3001
QQQ         1#
$ABORT      1#   1751
$AUTOS      1#
$BINCT      1#   1404   1501   1625
$BUFFE      1#   1029
$CLKDN      1#    990
$CLKUP      1#    987
$CLRVE      1#   1010
$CYCLE      1#   1038
$DECST      1#   2050   2090   2133   2203   2251   2321   2434   2556   2795   2877
$DEC1       1#   2046
$DEC10      1#   2714
$DEC2       1#   2086
$DEC20      1#   2671
$DEC21      1#   2776
$DEC22      1#   2852
$DEC23      1#   3006
$DEC3       1#   2247
$DEC4       1#   2199
$DEC5       1#   2301
$DEC6       1#   2129
$DEC7       1#   2414   2524
$DEFAU      1#    526
$DELY       1#
$EOP        1#    599
$ERRTA      1#
$FINI       1#   3090
$GETFL      1#    479    483    489    493
$GETPA      1#    446    454    462    471   1110
$HDPLX      1#   1330
$HEADE      1#
$MSG        1#   1023
$MSGA      3090#
$ONES       1#   1949
$PFAIL      1#   1000
$POKCK      1#    984
$PRIOR      1#   1195   1227   1261   1295
$QUEST      1#    437
$RESET      1#   1211   1245   1279   1313   1347   1397   1418   1515   1622   1639   1748   1764   1817   1837
         1922    1962   2002   2059   2080   2099   2142   2164   2193   2212   2260   2322   2406   2435   2462
         2557    2584   2685   2734   2796   2822   2878   2904   3025   3065
$SCOPE      1#    645
$SCOP2      1#   3069
$SETFL      1#    870
$SETVE      1#   1143
$SHUTD      1#   1821
$SIMBC      1#   1156
```

DZDPDA.P11 CROSS REFERENCE TABLE -- MACRO NAMES

```
$SMALL      1#   995
$SYNC       1#  2172  2223  2226  2267  2270  2329  2332  2335  2446  2568  2803  2806  2885  2888
$TRPDE      1#   277   279   281   283   285   287   289   291   293   295   297   299
$TSTN       1#  1202  1234  1268  1302  1336  1410  1507  1631  1756  1829  1954  2051  2091  2134
         2204   2252  2308  2419  2530  2677  2718  2781  2859  3017
$VARIA      1#   119
$WAIT       1#  1358  1370  1387  1435  1449  1542  1556  1668  1682  1787  1853  1872  1992  2065
         2112   2152  2178  2219  2229  2276  2353  2442  2453  2564  2575  2745  2812  2895
$XZ         1#  1196  1200  1228  1232  1262  1266  1296  1300  1330  1334  1405  1408  1502  1505
         1626   1629  1752  1755  1822  1827  1950  1953  2046  2049  2086  2089  2129  2132  2199
         2202   2247  2250  2301  2307  2414  2418  2524  2529  2672  2675  2714  2717  2777  2780
         2853   2857  3007  3016
```

DZDPDA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
ADC     1165
ADCB    1056  1064
ADD      521   523   549   690   707   778   826   836   885   901   924   927   985  1057  1066
        1094  1096  1098
ASL      757   758   759   899   923   925
BCC      519  3076
BEQ      427   430   488   498   501   507   546   552   560   589   623   651   660   679   681
         717   748   756   851   909   916   932   938   947   952   956   961   973   989  1108
        1167  1345  1362  1374  1391  1439  1454  1468  1546  1561  1575  1616  1619  1642  1672
        1687  1701  1742  1745  1791  1805  1857  1876  1894  1900  1908  1919  1997  2012  2026
        2069  2116  2124  2156  2182  2191  2222  2233  2242  2280  2296  2357  2377  2390  2392
        2401  2445  2457  2488  2503  2567  2579  2610  2628  2650  2700  2709  2749  2764  2817
        2847  2899  2949  2959  2984  3070
BGT      752
BHI      768
BIC      571   835   877   900   926   966   990  1171  1177  1178  1180  1496  1601  1727  1902
        1946  2041  2111  2352  2517  2664  2702  2836  2933
BICB     715   753
BIS      499   502   987  1179  1181  1211  1245  1279  1313  1347  1349  1355  1356  1357  1397
        1418  1423  1428  1429  1430  1433  1447  1515  1531  1536  1537  1540  1554  1622  1639
        1657  1662  1663  1666  1680  1748  1764  1773  1774  1775  1779  1786  1817  1837  1845
        1846  1847  1851  1852  1868  1923  1962  1984  1985  1986  1990  2003  2059  2062  2063
        2081  2099  2102  2103  2142  2145  2146  2147  2165  2168  2169  2170  2194  2212  2215
        2216  2217  2260  2263  2264  2265  2322  2325  2326  2327  2339  2407  2435  2438  2439
        2440  2450  2451  2463  2516  2557  2560  2561  2562  2572  2573  2585  2663  2685  2687
        2690  2732  2733  2735  2737  2739  2796  2799  2800  2801  2810  2811  2823  2878  2881
        2882  2883  2891  2892  2905  3025  3036  3037  3038  3065
BISB     754
BIT      437   551   559   588   650   657   678   691   908   913   970   972  1107  1166  1359
        1361  1371  1373  1388  1390  1436  1438  1451  1453  1475  1543  1545  1558  1560  1618
        1669  1671  1684  1686  1744  1788  1790  1801  1813  1854  1856  1873  1875  1887  1890
        1915  1994  1996  2066  2068  2074  2077  2113  2115  2123  2153  2155  2179  2181  2219
        2221  2230  2232  2277  2279  2354  2356  2366  2369  2383  2403  2442  2444  2454  2456
        2472  2477  2564  2566  2576  2578  2594  2599  2635  2746  2748  2757  2814  2816  2846
        2896  2898  2973  2999  3069
BITB     771  1052
BLO      770
BLOS     565
BLT      750
BMI     1367  1379  1444  1551  1677  1929  1939  2121  2187  2238  2285  2696  2705  2751  2943
BNE      438   445   529   554   558   579   587   619   658   663   692   699   722   772   780
         845   849   854   858   879   883   914   934   971   992  1018  1048  1053  1059  1069
        1106  1120  1122  1124  1131  1183  1243  1277  1311  1360  1365  1372  1377  1389  1394
        1437  1442  1452  1457  1472  1476  1487  1527  1544  1549  1559  1564  1579  1592  1653
        1670  1675  1685  1690  1705  1718  1785  1789  1794  1802  1809  1814  1855  1860  1874
        1879  1888  1891  1916  1934  1944  1978  1995  2000  2016  2035  2067  2072  2075  2078
        2114  2119  2154  2159  2180  2185  2220  2231  2236  2278  2283  2289  2355  2360  2367
        2370  2384  2404  2443  2455  2460  2473  2478  2493  2506  2513  2549  2565  2577  2582
        2595  2600  2615  2636  2653  2660  2744  2747  2754  2758  2772  2815  2820  2845  2897
        2902  2920  2929  2963  2974  2988  3000  3056  3059  3064  3078
BPL      434   654   694   697   713   719   911   963  1385  1432  1474  1539  1608  1665  1734
        1778  1781  1811  1850  1870  1988  2022  2106  2109  2150  2161  2173  2176  2224  2227
        2268  2271  2293  2330  2333  2336  2341  2344  2347  2350  2374  2387  2447  2483  2500
        2515  2569  2605  2621  2624  2647  2662  2741  2761  2804  2807  2886  2889  2946  2956
        2968  2981  2993  2997  3085
BR       420   428   510   525   550   568   593   652   656   728   760   762   881   888  1051
```

CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
        1061  1128  1133  1489  1492  1594  1597  1720  1723  1881  1904  1910  1912  1937  2037
        2163  2298  2380  2394  2397  2476  2481  2486  2491  2508  2598  2603  2608  2613  2617
        2643  2655  2834  2916  2922  2925  2931  2953  2965  2971  2978  2990
CLC      513   517   838   840   842  1054  1062  1172  3052  3073
CLR      406   411   412   443   527   556   562   576   605   648   665   666   745  1015  1162
        1222  1256  1290  1325  1354  1383  1421  1446  1448  1517  1529  1553  1555  1643  1655
        1679  1681  1770  1771  1776  1839  1848  1936  1965  1966  1979  1991  2273  2274  2338
        2428  2429  2449  2539  2541  2550  2551  2571  2728  2729  2730  2731  2768  2794  2809
        2831  2872  2873  2874  2875  2876  2893  2977  3030  3032  3033  3034  3057
CLRB     407   408   606   664   855   880   918
CMP      423   429   444   528   577   578   662   767   769   915   960  1016  1017  1058  1067
        1119  1121  1123  1130  1615  1741  1784  1808  1933  1943  1977  2015  2025  2034  2627
        2753  2844  3055  3058  3061
CMPB     564   716   747   749   751   755   878   882  1242  1276  1310  1467  1486  1574  1591
        1700  1717  1804  1893  1899  1907  1918  2011  2190  2241  2295  2376  2389  2391  2400
        2487  2502  2609  2649  2699  2708  2763  2919  2928  2948  2958  2962  2983  2987
COM     1168  1170  1614  1740  2024
COMB     432   440  3081
DEC      506   721   844   857   988   991  1182  1363  1375  1392  1440  1455  1547  1562  1673
        1688  1792  1858  1877  1998  2070  2117  2157  2183  2234  2281  2288  2358  2458  2580
        2743  2818  2900
DECB     618   779   848   853
EMT       64
HALT      95   563   567   572   967  1003  1050
INC      607   661   969  1076  1078  1082  1085  1087  1089  1091  1783  1807  1932  1942  1976
        2014  2033  2492  2614  2752  2843  3045  3046
INCB    1470  1485  1526  1577  1590  1652  1703  1716  2505  2512  2548  2652  2659  2769  2918
        2927  2961  2986
JMP      117   435   439   555   598   632   671   903   976  1009  1136  1624  1750
JSR      626  1104  1212  1213  1246  1247  1280  1281  1314  1315  1348  1350  1398  1419  1424
        1516  1522  1532  1580  1623  1640  1648  1658  1706  1749  1765  1766  1818  1838  1840
        1864  1924  1963  1972  1980  2004  2060  2082  2100  2143  2166  2195  2213  2261  2317
        2323  2408  2430  2436  2464  2544  2552  2558  2586  2686  2736  2790  2797  2824  2868
        2879  2906  3026  3047  3051  3066
MOV      402   403   404   409   413   414   416   417   418   421   422   424   425   442   470
         515   516   520   522   524   526   530   531   532   533   535   544   545   573   574
         575   584   585   591   592   594   595   596   608   622   631   649   655   667   668
         669   682   688   689   700   704   705   706   710   711   720   723   724   725   726
         729   730   736   737   738   739   740   741   744   746   776   777   781   782   793
         797   798   799   800   801   802   807   808   809   810   811   812   820   821   822
         823   824   825   827   830   831   833   834   846   859   860   861   862   863   876
         896   898   902   917   920   922   928   929   930   965   974   975   984  1002  1005
        1006  1012  1013  1014  1060  1065  1070  1071  1072  1073  1074  1075  1077  1079  1080
        1081  1083  1084  1086  1088  1090  1093  1095  1097  1101  1102  1103  1118  1125  1135
        1138  1143  1144  1156  1157  1158  1159  1160  1161  1163  1169  1174  1175  1176  1184
        1185  1186  1187  1208  1209  1210  1217  1218  1224  1240  1241  1244  1251  1252  1258
        1274  1275  1278  1285  1286  1292  1308  1309  1312  1319  1320  1327  1342  1343  1346
        1358  1369  1370  1386  1387  1395  1396  1402  1416  1417  1420  1422  1434  1435  1450
        1466  1478  1488  1490  1491  1493  1497  1498  1513  1514  1518  1519  1520  1521  1528
        1530  1541  1542  1557  1573  1593  1595  1596  1598  1602  1603  1609  1611  1613  1637
        1638  1644  1645  1646  1647  1654  1656  1667  1668  1683  1699  1719  1721  1722  1724
        1728  1729  1735  1737  1739  1762  1763  1772  1782  1787  1796  1799  1800  1812  1816
        1835  1836  1844  1853  1862  1871  1872  1885  1886  1896  1898  1903  1905  1906  1913
        1914  1925  1931  1935  1941  1945  1947  1960  1961  1964  1967  1968  1969  1970  1971
        1989  1993  2010  2017  2028  2032  2036  2038  2042  2043  2057  2058  2061  2064  2065
        2073  2097  2098  2101  2104  2107  2110  2112  2140  2141  2144  2148  2151  2152  2167
```

```
        2171  2174  2177  2178  2189  2210  2211  2214  2218  2225  2228  2229  2258  2259  2262
        2266  2269  2272  2275  2276  2287  2294  2314  2315  2316  2324  2328  2331  2334  2337
        2342  2345  2348  2351  2353  2364  2365  2379  2381  2382  2396  2398  2399  2409  2425
        2426  2427  2437  2441  2448  2453  2465  2470  2471  2475  2480  2485  2490  2494  2495
        2497  2498  2507  2511  2518  2519  2536  2537  2538  2540  2542  2543  2559  2563  2570
        2575  2587  2592  2593  2597  2602  2607  2612  2616  2618  2622  2625  2642  2644  2645
        2654  2658  2665  2666  2683  2684  2688  2689  2691  2693  2698  2707  2724  2725  2726
        2727  2738  2742  2756  2767  2787  2788  2789  2798  2802  2805  2808  2813  2825  2832
        2833  2835  2837  2838  2842  2849  2865  2866  2867  2880  2884  2887  2890  2895  2907
        2913  2914  2915  2917  2921  2923  2924  2926  2930  2932  2934  2935  2940  2941  2951
        2952  2954  2964  2966  2970  2972  2976  2979  2989  2991  2998  3002  3023  3024  3027
        3028  3029  3031  3035  3039  3041  3044  3054  3060  3071  3072  3086
MOVB     405   410   441   503   504   505   511   512   534   536   537   538   539   540   541
         569   570   620   621   698   714   742   743   828   829   832   837   847   852   884
         912   945  1145  1146  1484  1589  1606  1610  1715  1732  1736  2020  2023  2240  2626
        2770  2771
NOP      591   592   627   628   629   630   997  1219  1220  1221  1253  1254  1255  1287  1288
        1289  1321  1322  1323  1494  1495  1599  1600  1725  1726  2039  2040
RESET    624  1004
ROL     3075
ROLB     509  1055  1063
ROR      839   841   843  1164  1173  3074
RORB     514   518  3053
RTI      683   701   731   783   803   813   864   886   977   993  1150  1154  1225  1259  1293
        1328  1403  1480  1499  1585  1604  1612  1711  1730  1738  1819  1897  1926  1948  2018
        2029  2044  2411  2496  2520  2619  2667  2839  2850  2936  3003
RTS      998  1019  1139  1147  1188  3087
SEC      508
SUB      897   921
TRAP     277   279   281   283   285   287   289   291   293   295   297   299
TST      419   553   557   586   680   933   937   946   951   955   962  1105  1129  2373  2386
        2482  2499  2604  2646  2760  2945  2955  2967  2980  2992
TSTB     426   433   487   497   500   653   659   693   696   712   718   850   910   931  1047
        1344  1366  1378  1384  1431  1443  1471  1473  1538  1550  1578  1607  1641  1664  1676
        1704  1733  1777  1780  1810  1849  1869  1928  1938  1987  2021  2105  2108  2120  2149
        2160  2172  2175  2186  2223  2226  2237  2267  2270  2284  2292  2329  2332  2335  2340
        2343  2346  2349  2446  2514  2568  2620  2623  2661  2695  2704  2740  2750  2803  2806
        2885  2888  2942  2996  3077  3084
.ASCIZ   121  1023  3090
.BLKB    176   177   178   179   180   247   248
.BLKW    253   254   337   338   339   341   342   343   345   346   347   349   350   351   353
         354   355   357   358   359   361   362   363   365   366   367  1036  1524  1525  1650
        1651  1974  1975
.BYTE    261   262   263   264   326   328   452   453   460   461   468   469   477   478   634
         637   640   643   979   982  1024  1026  1116  1117  1216  1250  1284  1318  1353  1427
        1535  1583  1661  1709  1769  1843  1867  1983  2320  2433  2555  2793  2871  3090
.ENABL     1
.END    3091
.ENDC   1197  1200  1201  1210  1229  1232  1233  1242  1244  1263  1266  1267  1276  1278  1297
        1300  1301  1310  1312  1331  1334  1335  1344  1366  1378  1395  1406  1408  1409  1418
        1420  1443  1458  1473  1479  1501  1503  1505  1506  1515  1517  1528  1550  1565  1584
        1625  1627  1629  1630  1639  1643  1654  1676  1691  1710  1751  1753  1755  1756  1764
        1795  1823  1827  1828  1837  1861  1880  1951  1953  1954  1962  2001  2047  2049  2050
        2059  2073  2087  2089  2090  2099  2120  2130  2132  2133  2142  2160  2186  2200  2202
        2203  2212  2223  2237  2248  2250  2251  2260  2284  2302  2307  2308  2316  2361  2415
        2418  2419  2427  2428  2442  2446  2461  2497  2509  2525  2529  2530  2538  2550  2564
```

```
                2568    2583    2644    2656    2673    2675    2676    2685    2715    2717    2718    2726    2750    2778    2780
                2781    2789    2821    2854    2857    2858    2867    2903    3008    3016    3017    3025    3091
.EQUIV            64
.EVEN            181     267    1023    1028    3090
.IF             1196    1197    1200    1209    1210    1228    1229    1232    1241    1242    1262    1263    1266    1275    1276
                1296    1297    1300    1309    1310    1330    1331    1334    1343    1358    1370    1397    1405    1406    1408
                1417    1420    1435    1450    1473    1501    1502    1503    1505    1514    1517    1542    1557    1580    1584
                1606    1626    1627    1629    1638    1641    1643    1668    1683    1706    1710    1732    1752    1753    1755
                1763    1787    1822    1823    1827    1836    1853    1872    1950    1951    1953    1961    1993    2046    2047
                2049    2058    2065    2086    2087    2089    2098    2112    2129    2130    2132    2141    2152    2178    2199
                2200    2202    2211    2219    2229    2247    2248    2250    2259    2276    2301    2302    2307    2315    2353
                2414    2415    2417    2418    2426    2427    2428    2434    2442    2453    2497    2507    2524    2525    2527
                2529    2537    2538    2539    2556    2564    2575    2620    2654    2672    2673    2675    2684    2714    2715
                2717    2725    2746    2777    2778    2780    2788    2813    2853    2854    2857    2866    2895    3007    3008
                3016    3024    3091
.IFF            1209    1210    1241    1242    1275    1276    1309    1310    1343    1344    1358    1370    1387    1417    1418
                1435    1450    1514    1515    1542    1557    1638    1639    1668    1683    1763    1764    1787    1836    1837
                1853    1872    1961    1962    1993    2058    2059    2065    2098    2099    2112    2141    2142    2152    2178
                2211    2212    2223    2229    2259    2260    2276    2315    2316    2353    2418    2426    2427    2442    2446
                2453    2507    2527    2537    2538    2556    2568    2575    2656    2684    2685    2725    2726    2750    2788
                2789    2813    2866    2867    2895    3024
.IFT            1358    1370    1387    1435    1450    1542    1557    1668    1683    1787    1853    1872    1993    2065    2112
                2152    2178    2219    2229    2276    2353    2417    2428    2434    2442    2453    2497    2507    2527    2539
                2556    2564    2575    2620    2654    2746    2813    2895
.IIF            1209    1210    1222    1241    1242    1256    1275    1276    1290    1309    1310    1324    1343    1344    1417
                1418    1423    1424    1428    1514    1515    1531    1532    1536    1638    1639    1657    1662    1763    1764
                1836    1837    1961    1962    2051    2058    2059    2061    2062    2063    2091    2098    2099    2101    2102
                2103    2134    2141    2142    2144    2145    2146    2174    2204    2211    2212    2214    2215    2216    2225
                2228    2252    2259    2260    2262    2263    2264    2269    2272    2315    2316    2322    2324    2325    2326
                2331    2334    2337    2426    2427    2434    2435    2437    2438    2439    2442    2448    2494    2495    2537
                2538    2556    2557    2559    2560    2561    2564    2570    2616    2617    2684    2685    2725    2726    2788
                2789    2796    2798    2799    2800    2805    2808    2866    2867    2878    2880    2881    2882    2887    2890
                3024    3025
.IRP             277     279     281     283     285     287     289     291     293     295     297     299     337    1196    1202
                1209    1228    1234    1241    1262    1268    1275    1296    1302    1309    1330    1336    1343    1405    1410
                1417    1502    1507    1514    1626    1631    1638    1752    1756    1763    1822    1829    1836    1950    1954
                1961    2046    2051    2058    2086    2091    2098    2129    2134    2141    2199    2204    2211    2247    2252
                2259    2301    2308    2315    2414    2419    2426    2524    2530    2537    2672    2677    2684    2714    2718
                2725    2777    2781    2788    2853    2859    2866    3007    3017    3024    3090
.LIST              1      14      35      83      85      95     119     121     279     281     283     285     287     289     291
                 293     295     297     299     301     375     394     543     599    1023    1140    1195    1196    1200    1210
                1227    1228    1232    1242    1261    1262    1266    1276    1295    1296    1300    1310    1330    1334    1344
                1404    1405    1408    1418    1501    1502    1505    1515    1625    1626    1629    1639    1751    1752    1755
                1764    1821    1822    1827    1837    1949    1950    1953    1962    2046    2049    2059    2086    2089    2099
                2129    2132    2142    2199    2202    2212    2247    2250    2260    2301    2307    2316    2414    2418    2427
                2524    2529    2538    2671    2672    2675    2685    2714    2717    2726    2776    2777    2780    2789    2852
                2853    2857    2867    3006    3007    3016    3025    3090
.MACRO             1    3090
.NLIST             1      14      35      83      85      95     119     121     279     281     283     285     287     289     291
                 293     295     297     299     301     375     394     543     599    1023    1140    1195    1196    1200    1210
                1227    1228    1232    1242    1261    1262    1266    1276    1295    1296    1300    1310    1330    1334    1344
                1404    1405    1408    1418    1501    1502    1505    1515    1625    1626    1629    1639    1751    1752    1755
                1764    1821    1822    1827    1837    1949    1950    1953    1962    2046    2049    2059    2086    2089    2099
                2129    2132    2142    2199    2202    2212    2247    2250    2260    2301    2307    2316    2414    2418    2427
                2524    2529    2538    2671    2672    2675    2685    2714    2717    2726    2776    2777    2780    2789    2852
                2853    2857    2867    3006    3007    3016    3025    3090
```

```
.PAGE      1     35     85    183    257    304    332    375    394    599   1038   1140   3090
.REM     375
.REPT     95
.SBTTL     1     35     85    119    394    599   1195   1227   1261   1295   1330   1404   1501   1625   1751
        1821   1949   2046   2086   2129   2199   2247   2301   2414   2524   2671   2714   2776   2852   3006
.TITLE    14
.WORD    246    249    250    251    252   2546   2547   3049   3050
```

```
ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

*,DZDPDA.SEQ/SOL/CRF/PAGNUM/NL:TOC=DZDPXX.MAC,DZDPDA.P11
RUN-TIME: 28 41 6 SECONDS
RUN-TIME RATIO: 193/77=2.5
CORE USED:  26K  (51 PAGES)
```

Spooler runtime 13 Seconds, 56 KCS, 352 disk reads, 3 disk writes, 80 pages