DU11

COMBINED EXERCISER

MD-11-DDDUA-A

EP-DDDUA-A-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA

.REM *

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DDDUA-A-D
PRODUCT NAME:	DU11 OFFLINE COMBINED EXERCISER
DATE RELEASED:	21 DECEMBER 1975
MAINTAINER:	DIAGNOSTIC GROUP

*

GENERAL DESCRIPTION

THIS DIAGNOSTIC CAN CHAIN 16 DU11'S. THIS MEANS THAT
16 DEVICES CAN BE SEQUENTIALLY EXERCISED. THE DIAGNOSTIC
MAKES ONE PASS BEFORE PROCEEDING TO THE NEXT DEVICE,
AND CONTINUES EXERCISING ALL DEVICES IN THIS FASHION UNTIL
HALTED.

THIS DIAGNOSTIC CAN BE RUN WITH OR WITHOUT A
CONSOLE TELETYPE.

THE AUTOSIZER PROGRAM MUST BE RUN PRIOR TO THIS DIAGNOSTIC

TO MAKE SURE THAT THE DEVICE EXISTS IN THE SYSTEM.

1. THE DU11 OFFLINE COMBINED EXERCIZER FOR PRODUCTION TESTS
THAT THE TRANSMITTER AND RECEIVER CHIPS WORK PROPERLY,PERFORM
TESTING OF THE ASSOCIATED LOGIC ON THE OPTION,AND CHECK THE
OPERATION OF THE INTERRUPT LOGIC. IT CAN BE USED ON THE
"XOR" ,DDP,ACT 11,AND THE "CHIP TESTER".
TO USE ON THE "XOR" TESTER ,THE BRANCH AROUND THE "XOR"
CODE MUST BE PATCHED TO A "NOP". (SEE LISTINGS FOR DETAILS)

2. REQUIREMENTS

PDP-11 FAMILY STANDARD COMPUTER WITH MINIMUM 4K MEMORY

DU11 SYNCRONOUS/ISOCRONOUS OPTION

ONE CONSOLE TELETYPE OR EQUIVALENT --- OPTIONAL.

2.2 STORAGE
THE PROGRAM LOADS INTO 4K OF MEMORY W--H BOOTSTRAP

3. LOADING PROCEDURE

THE STANDARD PROCEDURE FOR LOADING ABSOLUTE BINARY TAPES
IS TO BE USED.

	STARTING ADDRESS FOR ABSOLUTE LOADER
4K	017500
8K	037500
12K	057500
16K	077500
20K	117500
24K	137500
28K	157500

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

```
4.1.1    AFTER PROGRAM LOAD (INITIAL PROGRAM START)
         ALL CONSOLE SWITCHES DOWN

4.1.2    TO MODIFY DEVICE VECTOR AND CONTROL REGISTER ADDRESSES
         AFTER PROGRAM RESTART OR TO RUN MULTIPLE DEVICES

         SW00=1

4.1.3    TO START PROGRAM AT SELECTED TEST AFTER A PROGRAM RESTART
         (ONLY IN SINGLE DEVICE TESTS AND ONLY IF THERE IS A TTY, THAT
         IS, THERE CAN BE CONVERSATION BETWEEN TTY AND THE SYSTEM)
         SW01=1

4.1.4    TO LOCK ON SELECTED TEST AFTER A PROGRAM RESTART
         (ONLY IN SINGLE DEVICE TESTS AND ONLY IF THERE IS A TTY)

         SW02=1
         NOTE1: IN GENERAL  SW01 WILL BE USED WHEN SW02=1 IS USED
         NOTE2: WITHOUT SW01=1   "LOCK ON TEST" WILL DEFAULT TO TEST 1
4.2      STARTING ADDRESS

         THE STARTING ADDRESS FOR ALL TESTS IS  000200

         THE RETARTING ADDRESS FOR ALL TESTS  IS   000200
         THE STARTING ADDRESS TO ENTER A SELECTED TEST IS   000200
         THE STARTING ADDRESS TO LOCK ON TEST IS   000200

4.3      PROGRAM AND/OR OPERATOR ACTION

4.3.1    INITIAL PROGRAM START

   EITHER   A)  ONLY IF THERE IS NO TTY ---

         4.3.1.1   LOAD PROGRAM INTO MEMORY WITH ABSOLUTE LOADER

         4.3.1.2  PATCH THE TAGS TPCSR AND TPDBR WITH 177570

         4.3.1.3   LOAD ADDRESS 000200

         4.3.1.4   CLEAR CONSOLE SWITCHES

         4.3.1.5   SET SW15=1, SW07=1,  SW06=1

         4.3.1.6   PRESS START

         IF THERE ARE NO ERRORS,PROGRAM WILL RUN
         FOR ABOUT 22 SECONDS,AT THE END OF WHICH THERE
         WILL BE "5" ON THE DISPLAY LIGHTS FOR COUPLE
         OF SECONDS AND THEN PROGRAM WILL HALT AT ADDRESS
         252 FOR END OF PASS. PRESS CONTINUE TO CONTINUE THE PROGRAM.
         IF ANY ERROR ENCOUNTERED, PROGRAM WILL HALT AT ADDRESS 14340

         TO DETERMINE TYPE OF ERROR:

         4.3.1.7   LOAD ADDRESS 017400
```

4.3.1.8 PRESS EXAMINE

CONTENT OF THIS LOCATION IS -- - RECEIVER ERROR PC

4.3.1.9 PRESS EXAMINE

CONTENT IS --- ADDRESS OF REGISTER

4.3.1.10 PRESS EXAMINE

CONTENT IS --- EXPECTED VALUE

4.3.1.11 PRESS EXAMINE

CONTENT IS --- ACTUAL VALUE

TO TEST IF THERE IS ANY MORE ERROR IN ANY OTHER TEST

4.3.1.12 LOAD ADDRESS 000200

4.3.1.13 CLEAR CONSOLE SWITCHES

4.3.1.14 SET SW15 = 1

4.3.1.15 SET SW11 = 1

4.3.1.16 SET SW07 = 1

4.3.1.17 SET SW06 = 1

4.3.1.18 PRESS START

PROGRAM WILL HALT FOR THE FIRST ERROR THAT WAS JUST OBSERVED

4.3.1.19 PRESS CONTINUE (SW15 = 1, SW11 = 1, SW07 = 1)

IF THE PROGRAM HALTS AT ADDRESS 014340:

4.3.1.20 LOAD ADDRESS 017400

4.3.1.21 PRESS EXAMINE

CONTENT IS THE NEW RECEIVER ERROR PC FOR THE 2ND ERROR.

NEXT THREE LOCATIONS CONTAIN

ADDRESS OF THE REGISTER,EXPECTED VALUE,AND ACTUAL VALUE

TESTS 4.3.1.12 THRU 4.3.1.21 CAN BE REPEATED FOR ANY

FURTHER ERROR.

OR B) IF THERE IS A TTY ---

4.3.1.1 LOAD PROGRAM INTO MEMORY WITH ABSOLUTE LOADER

4.3.1.2 LOAD ADDRESS 000200

4.3.1.3 CLEAR CONSOLE SWITCHES

4.3.1.4 PRESS START

4.3.1.5 THE PROGRAM WILL TYPE "DU11 DDDUA-A TAPE COMBINED EXERCIZER" (ONCE ONLY)
4.3.1.6 THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT IS ABOUT
TO START TESTING ,AND THEN TESTING WILL BEGIN

IF THE SYSTEM DOES NOT HAVE A TTY DO NOT

PERFORM THE FOLLOWING STEPS :

4.3.2, 4.3.3, 4.3.4, 4.3.5

4.3.2 PROGRAM RESTART WITH ALL SWITCHES DOWN

4.3.2.1 THE PROGRAM WILL TYPE "R" AND WILL COMMENCE TESTING

4.3.3 PROGRAM RESTART WITH SW00=1

4.3.3.1 LOAD ADDRESS 000200

4.3.3.2 SET SW00=1

4.3.3.3 PRESS START

4.3.3.4 THE PROGRAM WILL TYPE " 1ST DEVICE: RECEIVER CONTROL REGISTER
ADDRESS" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.5 TYPE IN THE ADDRESS OF THE FIRST RECEIVER CONTROL
REGISTER ADDRESS OF THE DU11 TO BE TESTED
FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ADDRESS IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.4

4.3.3.6 THE PROGRAM WILL TYPE "VECTOR ADDRESS-" AND WAIT FOR AN
INPUT FROM THE TELETYPE KEYBOARD

4.3.3.7 TYPE IN THE BASE RECEIVER INTERRUPT VECTOR ADDRESS
FOR THE DU11 TO BE TESTED FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ADDRESS IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.6

4.3.3.8 THE PROGRAM WILL TYPE "ARE YOU RUNNING MULTIPLE DEVICES ?" (Y OR N)-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.9 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ANSWER IS GIVEN, THE PROGRAM WILL TYPE "?" AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.8

IF A "NO" ANSWER IS GIVEN: JUMP TO SECTION 4.3.3.12
IF A "YES" ANSWER IS GIVEN:THE NEXT QUESTION IS ASKED

4.3.3.10 THE PROGRAM WILL TYPE "LAST DEVICE:RECEIVER CONTROL REGISTER ADDRESS-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.11 TYPE IN THE ADDRESS OF THE LAST RECEIVER CONTROL REGISTER ADDRESS OF THE DU11 TO BE TESTED FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?" AND WILL THEN REPEAT THE MESSAGE OF 4.3.3.10
NOTE:ALL ADDRESSES SHALL BE CONTIGUOUS

4.3.3.11.1 IF AN "OUT OF RANGE" ADDRESS IS TYPED IE. MORE THAN 16 (10) DEVICES AWAY (UPWARDS).....THE PROGRAM WILL TYPE "OUT OF RANGE:RETYPE LAST DEVICE RXCSR ADDRESS-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.11.2 TYPE IN THE ADDRESS OF THE LAST RECEIVER CONTROL REGISTER ADDRESS OF THE DU11 TO BE TESTED FOLLOWED BY A <CARRIAGE RETURN>

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?" AND WILL REPEAT THE MESSAGE OF 4.3.3.11.1

IF A DEVICE ADDRESS LOWER THAN 1ST DEVICE ADDRESS IS TYPED........... ....SCHOOLS OUT.........THERE IS NO PROTECTION FOR THIS. THE PROGRAM WILL DEFAULT TO TWO DEVICES ACTIVE (UPWARDS FROM 1ST DEVICE ADDRESS).THE SAME APPLIES TO IDENTICAL ADDRESSES TYPED FOR FIRST AND LAST DEVICE.
OBSERVE LOCATION @ ACTREG: SEE SECTION 7.2

4.3.3.12 THE PROGRAM WILL TYPE "DU PRIORITY LEVEL-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.13 TYPE IN THE APPROPRIATE DEVICE PRIORITY LEVEL OF THE DU11 OR DU11'S TO BE TESTED FOLLOWED BY A <CARRIAGE RETURN> (NOTE THAT ALL MULTIPLE DEVICES MUST BE AT THE SAME PRIORITY LEVEL). IE "5"

IF AN INCORRECT LEVEL IS TYPED ,THE PROGRAM WILL TYPE "?"
AND REPEAT THE MESSAGE OF 4.3.3.12

4.3.3.14 THE PROGRAM WILL TYPE "# OF SYNC CHARS
SELECTED (1 OR 2)-" AND WAIT FOR AN INPUT FROM THE TELETYPE
KEYBOARD

4.3.3.15 TYPE IN THE APPROPRIATE ANSWER "1" OR "2" FOLLOWED
BY A <CARRIAGE RETURN>.(NOTE:ALL MULTIPLE DEVICES MUST
BE THE SAME)

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL REPEAT THE MESSAGE OF 4.3.3.14

4.3.3.16 THE PROGRAM WILL TYPE " IS SEC XMIT JUMPER #6 IN ? (Y OR N)-"
AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.17 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED
BY A <CARRIAGE RETURN>.(NOTE THAT ALL MULTIPLE DEVICES
MUST BE THE SAME)

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL REPEAT THE MESSAGE OF 4.3.3.16

4.3.3.18 THE PROGRAM WILL TYPE "IS SEC REC JUMPER # 5 IN ?
(Y OR N)-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.19 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED
BY A <CARRIAGE RETURN>. (NOTE: ALL MULTIPLE DEVICES MUST BE THE SAME)

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL REPEAT THE MESSAGE OF 4.3.3.18

4.3.3.20 THE PROGRAM WILL TYPE "IS OPT CLR ENABLE JUMPER
4 IN ? (Y OR N)-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.3.21 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED
BY A <CARRIAGE RETURN>. (NOTE: ALL MULTIPLE DEVICES MUST BE THE SAME)

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?"
AND WILL REPEAT THE MESSAGE OF 4.3.3.20

4.3.3.22 THE PROGRAM WILL TYPE "ARE YOU RUNNING IN MAINT.
MODE EXTERNAL ? AND ......DO YOU HAVE THE EXTERNAL MODEM
BYPASS JUMPER CONNECTOR ON ? (Y OR N)-" AND WAIT FOR AN
INPUT FROM THE TELETYPE KEYBOARD

4.3.3.23 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED BY
A <CARRIAGE RETURN>. (NOTE: ALL MULTIPLE DEVICES MUST BE THE SAME)

IF AN INCORRECT ANSWER IS TYPED ,THE PROGRAM WILL TYPE "?"

AND WILL REPEAT THE MESSAGE OF 4.3.3.22

4.3.3.24 THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT HAS STARTED AND WILL COMMENCE TESTING AT TEST 1

4.3.4 PROGRAM RESTART WITH SW01=1
NOTE: THIS WILL ONLY WORK WHEN A SINGLE DEVICE IS SELECTED
...,IT WILL NOT WORK IF MULTIPLE DEVICES ARE SELECTED

IF MULTIPLE DEVICES WERE PREVIOUSLY SELECTED,LOAD 000200,
AND SELECT SW00=1 AND ANSWER "NO" TO THE MULTIPLE DEVICE QUESTION
SEE 4.3.3

4.3.4.1 LOAD 000200

4.3.4.2 SET SW01=1

4.3.4.3 PRESS START

4.3.4.4 THE PROGRAM WILL TYPE "TEST PC-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.4.5 TYPE IN THE ADDRESS OF THE TEST AT WHICH THE PROGRAM IS TO BE STARTED FOLLOWED BY A <CARRIAGE RETURN>

4.3.4.6 THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT HAS STARTED TESTING AT THE SELECTED TEST

NOTE: CARE MUST BE TAKEN WHEN THIS FEATURE IS USED
,SINCE THERE IS NO PROTECTION AGAINST SELECTING AN ADDRESS
THAT IS IN THE MIDDLE OF A TEST

4.3.5 PROGRAM RESTART WITH SW02 =1
NOTE: THIS WILL ONLY WORK WHEN A SINGLE DEVICE IS SELECTED
SEE NOTE IN 4.3.4 FOR MORE DETAILS

4.3.5.1 LOAD ADDRESS 000200

4.3.5.2 SET SW02 =1
NOTE: IT MAY BE ADVANTAGEOUS TO SET SW01=1 (OPTIONAL)

4.3.5.3 PRESS START

4.3.5.4 THE PROGRAM WILL TYPE "LOCK ON SELECTED TEST ? (Y OR N)-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

4.3.5.5 TYPE IN THE APPROPRIATE ANSWER YES OR NO FOLLOWED BY A <CARRIAGE RETURN>

IF A NO ANSWER IS GIVEN: THIS LOCK ON TEST WILL BE IGNORED
AND THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT HAS STARTED
TESTING AT TEST 1

4.3.5.6 IF A YES ANSWER WAS GIVEN:THE PROGRAM WILL ACT AS FOLLOWS...
THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT HAS STARTED

TESTING AT TEST 1 AND WILL REMAIN IN TEST 1 UNTIL HALTED
OR IF ANY KEY IS STRUCK ON THE TELETYPE ,THE PROGRAM
WILL FREEZE ON THE NEXT TEST UNTIL A KEY IS STRUCK ON
THE TELETYPE AND SO FORTH THRU THE PROGRAM. IF SW01 =1 IT
WILL PERFORM AS IN SECTION 4.3.4 ALLOWING ONE TO FREEZE
ON A SELECTED TEST RATHER THAN DEFAULTING TO TEST 1

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

```
SW15 =1         HALT ON ERROR
SW14 =1         LOOP ON CURRENT TEST
SW13 =1         INHIBIT ERROR TYPEOUT
SW11 =1         INHIBIT ITERATIONS
SW10 =1         ESCAPE TO NEXT TEST ON ERROR
SW09 =1         LOOP ON ERROR
SW06 =1         HALT ON END OF PASS
SW02 =1         LOCK ON TEST
SW01 =1         RESTART PROGRAM AT SELECTED TEST
SW00 =1         RESELECT VECTOR AND CONTROL REGISTER ADRESSES
                &PARAMETERS AFTER A PROGRAM RESTART
```

TO INHIBIT "END OF PASS" TYPEOUT - TURN TELETYPE OFF

6. ERRORS

6.1 ERROR HALTS
THERE ARE FOUR DISTINCT ERROR TYPEOUTS

6.1.1 PC+2 = ERROR PC
WHERE PC +2 IS THE ADDRESS OF THE CALL TO THE ERROR HANDLER +2

REFER TO THE ABOVE "HLT" IN DIAGNOSTIC FOR ERROR DESCRIPTION

CHECK ADDRESS @ RXCSR: TO LOCATE THE DEVICE PRESENTLY UNDER
TEST WHEN RUNNING MULTIPLE DEVICES

6.1.2 PC +2 = REGISTER ERROR PC

```
REGISTER     EXPECTED     ACTUAL
16XXXX       YYYYYY       ZZZZZZ
```

WHERE 16XXXX IS THE ADDRESS OF THE FAILING DEVICE REGISTER

WHERE YYYYYY IS THE EXPECTED CONTENTS OF THAT REGISTER

WHERE ZZZZZZ IS THE ACTUAL CONTENTS OF THAT REGISTER

6.1.3 PC +2 = RECEIVER ERROR PC

```
REGISTER     EXPECTED     ACTUAL
16XXXX       YYYYYY       ZZZZZZ
```

WHERE 16XXXX IS THE ADDRESS OF THE FAILING RECEIVER (RXDBUF) REGISTER

WHERE YYYYYY IS THE EXPECTED DATA CONTENTS OF THAT REGISTER

WHERE ZZZZZZ IS THE ACTUAL DATA CONTENTS OF THAT REGISTER

```
6.1.4   PC +2 = TRANSMITTER ERROR PC
        REGISTER        EXPECTED        ACTUAL
        16XXXX          YYYYYY          ZZZZZZ

        WHERE 16XXXX IS THE ADDRESS OF THE FAILING TRANSMITTER (TXCSR) REGISTER

        WHERE YYYYYY IS THE EXPECTED CONTENTS OF THAT REGISTER

        WHERE ZZZZZZ IS THE ACTUAL CONTENTS OF THAT REGISTER

6.1.5   ERROR DESCRIPTIONS
        SEE LISTINGS FOR DETAILS OF ERRORS

6.2     ERROR RECOVERY

6.2.1   SW15 =0
        IF THE PROGRAM IS RUN WITH SW15 =0 ,NO OPERATOR ACTION IS
        REQUIRED TO CONTINUE TESTING

6.2.2   SW15 =1
        IF THE PROGRAM IS RUN WITH SW15 =1 ,TO CONTINUE TESTING
        AFTER THE PROGRAM HAS HALTED ,PRESS THE PROCESSOR
        CONSOLE "CONTINUE SWITCH"

        NOTE: THE PC + 2 OF THE "HLT" WILL BE DISPLAYED IN THE DATA LIGHTS.

6.2.3   ILLEGAL INTERRUPTS
        IF AN INTERRUPT OCCURS TO A VECTOR ADDRESS NOT SELECTED
        DURING PROGRAM INITIALIZATION, THE PROGRAM WILL HALT IN
        THE TRAPCATCHER. THE ADDRESS AT WHICH THE PROGRAM
        HALTS IS 2 GREATER THAN THE ADDRESS TO WHICH THE INTERRUPT
        OCCURED. THE PROGRAM MUST BE RESTARTED AT 000200 TO
        RECOVER FROM THIS ERROR.

6.2.4   ADDITIONAL TROUBLESHOOTING AIDS   ERRCNT: & PASCNT:
        CHECK THESE TWO TAG LOCATIONS FOR TOTAL # OF ERRORS AND PASSES RESPECTIVELY.
        LOADING 000200 AND RESTARTING WILL CLR THESE LOCATIONS.

6.3     END OF PASS ROUTINE
        THIS TYPEOUT IS MENTIONED HERE FOR CONVENIENCE
        IT IS IN THE FORM:

        END OF PASS TAPE Y
        16XXXX = DEVICE

        WHERE Y IS THE TAPE LOADED

        WHERE 16XXXX IS THE DEVICE'S BASE REGISTER ADDRESS

        TO INHIBIT THIS TYPEOUT - TURN TELETYPE OFF

7.      RESTRICTIONS

7.1     MULTIPLE DEVICES
        UP TO 16(10) DEVICES MAY BE TESTED. HOWEVER, THEY
        MUST HAVE CONTIGUOUS ADDRESSES AND VECTORS
```

```
        NOTE: IF ALL DEVICES UNDER TEST HAVE THE SAME INTERRUPT VECTOR
        YOU CAN CHANGE  "ZERO:    ADD   #10,BASEIV      ;NEXT BLOCK
        (VECTORS)" TO "ZERO:   ADD    #0,BASEIV";
        THEREBY THE VECTOR ADDRESSES WILL NOT BE
        UPDATED AFTER EACH PASS.

7.2     DISQUALIFYING DEVICES WHEN RUNNING MULTIPLE DEVICES

        WHEN RUNNING MULTIPLE DEVICES    AN ACTIVE BIT IS SET
        FOR EACH DEVICE RUNNING UNDER TEST  IE. BIT 0 FOR
        DEVICE 0   ,BIT 15 FOR DEVICE 15
        TO DISQUALIFY DEVICES:

7.2.1   IF DEVICE 0 IS TO BE DISQUALIFIED ,SIMPLY RESTART
        PROGRAM WITH SW00 =1 AND OMIT THE FIRST DEVICE.

7.2.2   IF HOWEVER, DEVICES 1 THRU 15 OR ANY COMBINATION THEREOF
        ARE TO BE DISQUALIFIED....LOAD THE LOCATION OF ACTREG:
        OBSERVE THE ACTIVE BITS (ACTIVE =1, NONACTIVE = 0)
        AND DEPOSIT 0 WHERE THOSE DEVICES ARE TO BE DISQUALIFIED

        7.2.2.1 TO RESTART...LOAD 000200 IN SWR AND DEPRESS START....
                THE PROGRAM WILL CONTINUE WITH THE DEVICE IT WAS IN BEFORE HALTING.

        7.2.2.2 .......OR .....LOAD 000200 WITH SW00 =1 AND DEPRESS START....
                ANSWER THE QUESTION :1ST DEVICE : ETC...........
                 ....THE PROGRAM WILL CONTINUE WITH DEVICE 0

        7.2.2.3 IF ALL DEVICES ARE DISQUALIFIED BY MISTAKE   THE PROGRAM
                WILL TYPEOUT   AN ERROR MESSAGE.......LOAD & START AT 000200

7.3     CABLE DELAYS
        NOTE: EXTERNAL LOOP BACK TESTS ONLY (MODEM CABLE WITH H315 CONNECTOR ON)

7.3.1   TO PROVIDE SUFFICIENT DELAY FOR CLOCK SIGNAL OVER THE CABLE,
        LOCATION "HOLD:" MUST BE MODIFIED TO ACCOMODATE FOR FASTER MACHINES.
                PRESENTLY "HOLD:" =20  IS SUFFICIENT TIME ON AN 11/20 MACHINE.
        IF RUNNING ON AN 11/40 OR AN 11/45 "HOLD:" MUST BE PPTCHED TO 40

        BASICALLY DON'T TRY TO EXCEED 10K TO 12K RATE USING THE EIA DRIVERS

7.4     TO USE THE "XOR" TESTER ,THE BRANCH AROUND THE "XOR"
        CODE MUST BE PATCHED TO A "NOP". (SEE LISTINGS FOR DETAILS)

8.      DEFAULT PARAMETERS:
        1ST DEVICE: RECEIVER CONTROL REGISTER ADDRESS- RXCSR: 160040

        VECTOR ADDRESS-                                DURIV: 770

        ARE YOU RUNNING MULTIPLE DEVICES ?- NO         MULTD: 0

        LAST DEVICE: RECEIVER CONTROL REGISTER ADDRESS-  LASTADD: 0

        DU PRIORITY LEVEL-  LEVEL 5            DUPRT: LEVEL 5
```

```
            # OF SYNC CHARS SELECTED -  2          SYNCNO: 377

            IS SEC XMIT JUMPER # 6 IN ?-  YES      SEXMIT: 377

            IS SEC REC JUMPER # 5 IN ?-  YES       SEREC: 377

            IS OPT CLR ENABLE JUMPER # 4 IN ?-  YES OPTCLR: 377

            DO YOU HAVE THE EXTERNAL MODEM BYPASS JUMPER
            CONNECTOR ON (H315)-  YES                   JMRBY: 377
```

9.   PROGRAM DESCRIPTION

9.1   THIS PROGRAM PERFORMS THE OFFLINE COMBINED (TRANSMITTER & RECEIVER)
TIMING & INTERRUPT TESTING OF THE DEVICE
SEE LISTING FOR DETAILS

10.   FLOW CHARTS: RECEIVER FLOW,TRANSMITTER FLOW,TRANSMITTER & RECEIVER FLOW

11.   LISTINGS   *
.
.

```
        .LIST   LOC,BIN,SEQ
        .ENABLE ABS

        .ENABLE AMA
```

```
                                    .ENABLE ABS

                                    ;DU11 DDDUA-A TAPE COMBINED EXERCIZER
                                    ;COPYRIGHT 1973, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

                                            ;STARTING PROCEDURE
                                            ;LOAD PROGRAM
                                            ;LOAD ADDRESS 000200
                                            ;PRESS START
                                            ;PROGRAM WILL TYPE "DU11 DDDUA-A TAPE COMBINED EXERCIZER"
                                            ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                            ;AT THE END OF A PASS, PROGRAM WILL TYPE "END OF PASS TAPE DDDUA-A"
                                            ;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
              004000                SW11=4000               ;=1,INHIBIT ITERATIONS
              002000                SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
              001000                SW09=1000               ;=1,LOOP WITH CURRENT DATA
              000400                SW08=400                ;=1,LOOP ON ERROR
              000100                SW06=100
              000040                SW05=40
              000020                SW04=20
              000010                SW03=10
              000004                SW02=4                  ;LOCK ON TEST SELECT
              000002                SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
              000001                SW00=1                  ;RESELECT VECTOR AND CONTROL REGISTER
                                                            ;ADDRESS AFTER PROGRAM RESTART
```

```

                        ;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           ;PROGPAM COUNTER

                        ;LOCATION EQUIVALENCIES

        177570          SWR=177570      ;CONSOLE SWITCH REGISTER
        177570          LIGHTS=177570   ;PDP-11/45 DISPLAY REGISTER
        177776          PS=177776       ;PROCESSOR STATUS WORD
        001100          STACK=1100      ;START OF PROCESSOR STACK

                        ;INSTRUCTION DEFINITIONS

        005746          PUSH1SP=5746    ;DECREMENT PROCESSOR STACK 1 WORD =TST -(SP)
        005726          POP1SP=5726     ;INCREMENT PROCESSOR STACK 1 WORD =TST (SP)+
        010046          PUSHR0=10046    ;SAVE R0 ON STACK =MOV R0,-(SP)
        012600          POPR0=12600     ;RESTORE R0 FROM STACK =MOV (SP)+,R0
        024646          PUSH2SP=24646   ;DECREMENT STACK TWICE =CMP -(SP),-(SP)
        022626          POP2SP=22626    ;INCREMENT STACK TWICE =CMP (SP)+,(SP)+
                        .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

                        ;PROCESSER LEVELS
        000340          LEVEL7=340
        000300          LEVEL6=300
        000240          LEVEL5=240
        000200          LEVEL4=200
        000140          LEVEL3=140
        000100          LEVEL2=100
        000040          LEVEL1=040
```

```
        000000          LEVEL0=000
                        ;REGISTER DEFINITIONS
                                ;RXCSR BIT DEFINITIONS
        100000          DSC=BIT15               ;DATA SET CHANGE
        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          SYNSCH=BIT4             ;SYNC SEARCH
        000010          STD=BIT3                ;SEC XMIT DATA
        000004          RTS=BIT2                ;REQ TO SEND
        000002          DTR=BIT1                ;DATA TERM RDY
        000001          VOID=BIT0
                                ;RXDBUF BIT DEFINITIONS
        100000          RXERR=BIT15             ;REC ERROR
        040000          OVRRUN=BIT14            ;OVERRUN
        020000          FRMERR=BIT13            ;FRAME ERROR
        010000          PARER=BIT12             ;PARITY ERROR
                                ;PARCSR BIT DEFINITIONS
        001000          PAREN=BIT9              ;PARITY ENABLE
        000400          EVPAR=BIT8              ;EVEN PARITY SENSE
                                ;PARCSR WRD DEFINITIONS
        030000          SYNINT=30000            ;SYNC EXTERNAL MODE
        020000          SYNEXT=20000            ;SYNC INTERNAL MODE
        000000          ISYMOD=0                ;ISOC MODE
        000000          FIVE=0                  ;WORD LENGTH 5 BITS
        002000          SIX=2000                ;WORD LENGTH 6 BITS
        004000          SEVEN=4000              ;WORD LENGTH 7 BITS
        006000          EIGHT=6000              ;WORD LENGTH 8 BITS
        000000          NOPAR=0                 ;NO PARITY
        001000          ODDPAR=1000             ;ODD PARITY
        001400          EVEPAR=1400             ;EVEN PARITY
                                ;TXCSR BIT DEFINITIONS
        100000          DNA=BIT15               ;DATA NOT AVAILABLE
        040000          MTDATA=BIT14            ;MAINT DATA
        020000          CLK=BIT13               ;CLK
        002000          BITW=BIT10              ;BIT WINDOW
        000400          MRESET=BIT8             ;MASTER RESET
        000200          TXDONE=BIT7             ;XMIT DONE
        000100          TXINTE=BIT6             ;XMIT INTR ENABLE
        000040          DNAINTE=BIT5            ;DNA INTR ENAB
        000020          SEND=BIT4               ;SEND
        000010          HDXEN=BIT3              ;HDX/FDX
        000001          BREAK=BIT0              ;BREAK
                                ;TXCSR WRD DEFINITIONS
        000000          USER=0                  ;USER MODE
        004000          MINT=4000               ;MAINT INT MODE
        010000          MEXT=10000              ;MAINT EXT MODE
        014000          SYSTST=14000            ;SYSTEM TEST MODE
```

```
                                                    ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
        000000           .=0
000000  000002                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000002  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000004  000006                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000006  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000010  000012                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000012  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000014  000016                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000016  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000020  000022                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000022  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000024  000026                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000026  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000030  000032                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000032  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000034  000036                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000036  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000040  000042                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000042  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000044  000046                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000046  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000050  000052                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000052  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000054  000056                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000056  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000060  000062                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000062  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000064  000066                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000066  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000070  000072                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000072  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000074  000076                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000076  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000100  000102                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000102  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000104  000106                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000106  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000110  000112                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000112  000000                                      ;EXAMINE STACK TO FIND CAUSE
000114  000116                                      ;UNEXPECTED TRAP TO THIS LOCATION
000116  000000                                      ;EXAMINE STACK TO FIND CAUSE
000120  000122                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000122  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000124  000126                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000126  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000130  000132                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000132  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000134  000136                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000136  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000140  000142                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000142  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000144  000146                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000146  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
000150  000152                   .+2                ;UNEXPECTED TRAP TO THIS LOCATION
000152  000000                   HALT               ;EXAMINE STACK TO FIND CAUSE
```

```
000154  000156          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000156  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000160  000162          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000162  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000164  000166          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000166  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000170  000172          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000172  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000174  000176          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000176  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000200  000202          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000202  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000204  000206          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000206  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000210  000212          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000212  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000214  000216          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000216  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000220  000222          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000222  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000224  000226          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000226  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000230  000232          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000232  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000234  000236          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000236  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000240  000242          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000242  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000244  000246          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000246  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000250  000252          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000252  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000254  000256          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000256  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000260  000262          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000262  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000264  000266          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000266  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000270  000272          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000272  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000274  000276          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000276  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000300  000302          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000302  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000304  000306          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000306  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000310  000312          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000312  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000314  000316          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000316  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000320  000322          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000322  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000324  000326          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000326  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
000330  000332          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000332  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
```

H02


```
000334  000336        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000336  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000340  000342        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000342  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000344  000346        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000346  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000350  000352        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000352  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000354  000356        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000356  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000360  000362        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000362  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000364  000366        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000366  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000370  000372        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000372  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000374  000376        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000376  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000400  000402        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000402  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000404  000406        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000406  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000410  000412        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000412  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000414  000416        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000416  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000420  000422        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000422  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000424  000426        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000426  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000430  000432        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000432  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000434  000436        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000436  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000440  000442        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000442  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000444  000446        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000446  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000450  000452        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000452  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000454  000456        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000456  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000460  000462        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000462  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000464  000466        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000466  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000470  000472        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000472  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000474  000476        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000476  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000500  000502        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000502  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000504  000506        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000506  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
000510  000512        .+2          ;UNEXPECTED TRAP TO THIS LOCATION
000512  000000        HALT         ;EXAMINE STACK TO FIND CAUSE
```

I02


```
000514 000516      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000516 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000520 000522      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000522 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000524 000526      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000526 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000530 000532      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000532 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000534 000536      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000536 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000540 000542      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000542 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000544 000546      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000546 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000550 000552      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000552 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000554 000556      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000556 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000560 000562      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000562 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000564 000566      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000566 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000570 000572      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000572 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000574 000576      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000576 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000600 000602      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000602 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000604 000606      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000606 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000610 000612      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000612 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000614 000616      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000616 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000620 000622      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000622 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000624 000626      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000626 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000630 000632      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000632 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000634 000636      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000636 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000640 000642      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000642 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000644 000646      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000646 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000650 000652      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000652 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000654 000656      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000656 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000660 000662      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000662 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000664 000666      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000666 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
000670 000672      .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000672 000000      HALT       ;EXAMINE STACK TO FIND CAUSE
```

```
        000674  000676          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000676  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000700  000702          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000702  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000704  000706          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000706  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000710  000712          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000712  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000714  000716          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000716  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000720  000722          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000722  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000724  000726          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000726  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000730  000732          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000732  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000734  000736          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000736  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000740  000742          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000742  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000744  000746          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000746  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000750  000752          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000752  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000754  000756          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000756  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000760  000762          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000762  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000764  000766          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000766  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000770  000772          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000772  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
        000774  000776          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
        000776  000000          HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
                                          ;STANDARD INTERRUPT VECTORS


        000024                    .=24
000024  014414                            .PFAIL                  ;POWER FAIL HANDLER
000026  000340                            340                     ;SERVICE AT LEVEL 7
000030  014122                            .HLT                    ;ERROR HANDLER
000032  000340                            340                     ;SERVICE AT LEVEL 7
000034  014070                            .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                            340                     ;SERVICE AT LEVEL 7
        000200                    .=200
000200  000137  001246                    JMP     .START          ;GO TO START OF PROGRAM

        000250                    .=250
000250  000000                    EOPHLT: HALT                    ;THIS IS AN END OF PASS HALT.
                                                                  ;NOT AN ERROR HALT.
                                                                  ;THIS HAPPENS ONLY IF SW6
                                                                  ;IS UP.PRESS CONTINUE
                                                                  ;TO RESUME PROGRAM

000252  000207                            RTS     PC



        001100                    .=1100

                                          ;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS

001100  177560                    TKCSR:  177560                  ;TELETYPE KEYBOARD CONTROL REGISTER
001102  177562                    TKDBR:  177562                  ;TELETYPE KEYBOARD DATA BUFFER
001104  177564                    TPCSR:  177564                  ;TELEPRINTER CONTROL REGISTER
001106  177566                    TPDBR:  177566                  ;TELEPRINTER DATA BUFFER

                                          ;PROGRAM CONTROL PARAMETERS

001110  000000                    RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
001112  000000                    NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
001114  000000                    LOCK:   0                       ;ADDRESS FOR LOCK ON CURRENT DATA
001116  000000                    ICOUNT: 0       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
001120  000000                    LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
001122  000000                    TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
001124  000000                    PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
001126  000000                    ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
001130  000000                    LSTERR: 0                       ;PC OF LAST ERROR CALL

                                          ;PROGRAM VARIABLES

001132  000020                    HOLD:   20              ;TEMPORARY STORAGE=DELAY TIME FOR CABLES
001134  000000                    SHIFT:  0               ;TEMPORARY STORAGE= # OF SHIFTS PER CHAR
001136  000000                    COUNT:  0               ;TEMPORARY STORAGE= # OF TIMES A CHAR WILL BE SENT
001140  000000                    TEMP1:  0                       ;TEMPORARY STORAGE
001142  000000                    TEMP2:  0                       ;TEMPORARY STORAGE
001144  000000                    TEMP3:  0                       ;TEMPORARY STORAGE
001146  000000                    TEMP4:  0                       ;TEMPORARY STORAGE
001150  000000                    TEMP5:  0                       ;TEMPORARY STORAGE
001152  000000                    SAVR0:  0                       ;R0 STORAGE
```

```
001154 000000      SAVR1:  0      ;R1 STORAGE
001156 000000      SAVR2:  0      ;R2 STORAGE
001160 000000      SAVR3:  0      ;R3 STORAGE
001162 000000      SAVR4:  0      ;R4 STORAGE
001164 000000      SAVR5:  0      ;R5 STORAGE
001166 000000      SAVSP:  0      ;STACK POINTER STORAGE
001170 000000      SAVPC:  0      ;PROGRAM COUNTER STORAGE
```

```
                                        ;PROGRAM CONVERSATIONAL PARAMETERS
001172     377         SYNCNO: .BYTE   377      ;# OF SYNC CHARS REQ'D FOR SYNC'ZATION
001173     377         SEXMIT: .BYTE   377      ;SEC XMIT JUMPER "IN"
001174     377         SEREC:  .BYTE   377      ;SEC REC JUMPER "IN"
001175     377         OPTCLR: .BYTE   377      ;OPTIONAL JUMPER CLR "IN"
001176     000         MULTD:  .BYTE   0        ;NO MULTIPLE DEVICE FLAG
001177     377         JMRBY:  .BYTE   377      ;EXTERNAL MODEM BYPASS JUMPER "IN"
                       .EVEN

                                        ;PROGRAM MULTIPLE DEVICE PARAMETERS
001200  000000         BASEADD:        0        ;PROG CONTROLLED 1ST DEVICE ADDR
001202  000000         KEEPADD:        0        ;SAVED 1ST DEVICE ADDR
001204  000000         LASTADD:        0        ;LAST DEVICE RXCSR ADDR
001206  000000         BASEIV:         0        ;PROG CONTROLLED IV
001210  000000         KEEPIV:         0        ;SAVED INTR VECTOR
001212  000000         ACTREG:         0        ;ACTIVE REGISTER ,,,MODIFY THIS
                                                ;LOCATION TO DISQUALIFY OR QUALIFY
                                                ;DEVICES (1= RUN,,,0= DON'T RUN)
001214  000000         ROTADD:         0        ;ROTATING POINTER FOR ACTREG..POINTS
                               ;TO DEVICE PRESENTLY UNDER TEST WHEN RUNNING MULTIPLE DEVICES

                               ;PROGRAM CONTROL FLAGS

001216     000         INIFLG: .BYTE   0                ;PROGRAM INITIALIZATION FLAG
001217     000         STFLG:  .BYTE   0                ;TEST START FLAG
001220     000         ERRFLG: .BYTE   0                ;ERROR OCCURED FLAG
001221     000         LOKFLG: .BYTE   0                ;LOCK ON CURRENT TEST FLAG

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

001222                 .TRPTAB:
                       ;**********************************************************************
        104400                 SCOPE=TRAP+0     ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001222  013032         .SCOPE
                       ;**********************************************************************
        104401                 SCOP1=TRAP+1     ;CALL TO LOOP ON CURRENT DATA HANDLER
001224  013214         .SCOP1
                       ;**********************************************************************
        104402                 TYPE=TRAP+2      ;CALL TO TELETYPE OUTPUT ROUTINE
001226  013232         .TYPE
                       ;**********************************************************************
        104403                 INSTR=TRAP+3     ;CALL TO ASCII STRING INPUT ROUTINE
001230  013264         .INSTR
                       ;**********************************************************************
        104404                 INSTER=TRAP+4    ;CALL TO INPUT ERROR HANDLER
001232  013356         .INSTER
                       ;**********************************************************************
        104405                 PARAM=TRAP+5     ;CALL TO NUMERICAL DATA INPUT ROUTINE
001234  013366         .PARAM
                       ;**********************************************************************
        104406                 SAV05=TRAP+6     ;CALL TO REGISTER SAVE ROUTINE
001236  013552         .SAV05
                       ;**********************************************************************
        104407                 RES05=TRAP+7     ;CALL TO REGISTER RESTORE ROUTINE
```

```
001240  013612                              .RESOS
                                            ;*****************************************************************
        104410                                      CONVRT=TRAP+10  ;CALL TO DATA OUTPUT ROUTINE
001242  013644                              .CONVRT
                                            ;*****************************************************************
        104411                                      SETFLG=TRAP+11  ;CALL TO FLAG SET ROUTINE
001244  014022                              .SETFLG

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

001246  012737  000340  177776      .START: MOV     #340,PS                         ;LOCK OUT INTERRUPTS
001254  012706  001100                      MOV     #STACK,SP                       ;SET UP STACK
001260  012737  014414  000024              MOV     #.PFAIL,@#24                    ;SET UP POWER FAIL VECTOR
001266  005037  001120                      CLR     LPCNT   ;CLEAR # OF ITERATION COMPLETED LOCATION
001272  105037  001217                      CLRB    STFLG                           ;CLEAR START FLAG
001276  005037  001124                      CLR     PASCNT                          ;CLEAR PASS COUNT
001302  105037  001220                      CLRB    ERRFLG                          ;CLEAR ERROR FLAG
001306  005037  001126                      CLR     ERRCNT                          ;CLEAR ERROR COUNT
001312  005037  001130                      CLR     LSTERR                          ;CLEAR LAST ERROR POINTER
001316  012737  000001  001122              MOV     #1,TSTNO                        ;SET UP FOR TEST 1
001324  012737  001246  001110              MOV     #.START,RETURN                  ;SET UP FOR POWER FAIL BEFORE
                                                                                    ;TESTING STARTS
001332  105737  001216                      TSTB    INIFLG                          ;HAS INITIALIZATION BEEN PERFORMED
001336  001004                              BNE     ONCE
001340  104402  014556                      TYPE    ,MTITLE                         ;TYPE TITLE MESSAGE
001344  105137  001216                      COMB    INIFLG                          ;IF NOT SET FLAG AND DO
001350  032737  000001  177570      ONCE:   BIT     #SWDO,SWR       ;RESELECT VECTOR & CONTROL REG?
001356  001002                              BNE     1$
001360  000137  002032                      JMP     .BEGIN
001364  012700  000300              1$:     MOV     #300,RO ;RESTORE VECTOR AREA TO TRAPCATCHER
001370  012701  000302                      MOV     #302,R1 ;START AT LOCATION 300
001374  012702  000004                      MOV     #4,R2
001400  010110                      2$:     MOV     R1,(RO)
001402  005011                              CLR     (R1)
001404  060200                              ADD     R2,RO
001406  060201                              ADD     R2,R1
001410  022701  001000                      CMP     #1000,R1        ;END AT LOCATION 776
001414  002771                              BLT     2$
001416  104403                              INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
001420  014653                              MREGAD  ;MESSAGE
001422  104405                              PARAM   ;CONVERT STRING
001424  160000                              160000  ;LOW LIMIT
001426  167776                              167776  ;HIGH LIMIT
001430  016524                              DUBASE  ;STORE AT THIS LOCATION
001432     001                      .BYTE   1       ;MASK
001433     001                      .BYTE   1       ;HOW MANY TIMES + 2
001434  013737  016524  001202              MOV     DUBASE,KEEPADD  ;SAVE
001442  004737  016372                      JSR     PC,DUADDR
001446  013737  001202  001200              MOV     KEEPADD,BASEADD ;RESTORE FOR ROTATION
001454  104403                              INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
001456  014631                              MVECTO  ;MESSAGE
```

```
     001460  104405                                 PARAM   ;CONVERT STRING
     001462  000300                                 300     ;LOW LIMIT
     001464  000776                                 776     ;HIGH LIMIT
     001466  017046                                 DURIV   ;STORE AT THIS LOCATION
     001470     001                         .BYTE   1       ;MASK
     001471     004                         .BYTE   4       ;HOW MANY TIMES + 2
     001472  013737  017046  001210                 MOV     DURIV,KEEPIV    ;SAVE
     001500  013737  017046  001206                 MOV     DURIV,BASEIV    ;SET UP FOR ROTATION
     001506  104403                                 INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
     001510  014734                                 MMULT   ;MESSAGE
     001512  104411                                 SETFLG  ;SET FLAG BASED UPON INPUT STRING
     001514  001176                                 MULTD   ;THIS FLAG
     001516  105737  001176                         TSTB    MULTD   ;ARE THERE MULTIPLE DEVICES
                                                                ;ON THE SYSTEM ?
     001522  100406                                 BMI     BBB     ;YES,ASK NEXT QUESTION
     001524  005037  001212                         CLR     ACTREG
     001530  005037  001214                         CLR     ROTADD
     001534  000137  001700                         JMP     OUTMUL  ;JUMP AROUND NEXT QUESTION
     001540                             BBB:
     001540  104403                                 INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
     001542  015013                                 MLASTD  ;MESSAGE
     001544  104405                                 PARAM   ;CONVERT STRING
     001546  160000                                 160000  ;LOW LIMIT
     001550  167776                                 167776  ;HIGH LIMIT
     001552  001204                                 LASTADD ;STORE AT THIS LOCATION
     001554     001                         .BYTE   1       ;MASK
     001555     001                         .BYTE   1       ;HOW MANY TIMES + 2
                                        ;THE FOLLOWING ROUTINE SETS UP ACTREG FOR THE FIRST TIME
     001556  012737  000001  001214 1$:     MOV     #1,ROTADD       ;SET UP POINTER
     001564  005037  001212                 CLR     ACTREG  ;CLR ACTIVE REGISTER
     001570  053737  001214  001212 2$:     BIS     ROTADD,ACTREG   ;MAKE THIS DEVICE ACTIVE
     001576  000241                         CLC
     001600  006137  001214                 ROL     ROTADD  ;SET UP POINTER
     001604  103421                         BCS     3$      ;ARE YOU OUT OF RANGE ?
     001606  062737  000010  001200         ADD     #10,BASEADD     ;SET UP BASE ADDRESS
     001614  023737  001204  001200         CMP     LASTADD,BASEADD ;IS THIS THE LAST DEVICE ?
     001622  101362                         BHI     2$      ;NO DO IT AGAIN
     001624  053737  001214  001212         BIS     ROTADD,ACTREG   ;THIS ASSUMES THAT THERE ARE AT
                                                        ;LEAST TWO DEVICES WHEN YOU ANSWER YES TO
                                                        ;MULTIPLE DEVICE QUESTION
     001632  012737  000001  001214 4$:     MOV     #1,ROTADD       ;SET UP FOR LATER USE IN END OF PASS ROUTINE
     001640  013737  001202  001200         MOV     KEEPADD,BASEADD ;DITTO
     001646  000414                         BR      OUTMUL  ;CONTINUE QUESTIONS
     001650  013737  001202  001200 3$:     MOV     KEEPADD,BASEADD ;RESTORE
     001656  104403                         INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
     001660  015176                         MRANGE  ;MESSAGE
     001662  104405                         PARAM   ;CONVERT STRING
     001664  160000                         160000  ;LOW LIMIT
     001666  167776                         167776  ;HIGH LIMIT
     001670  001204                         LASTADD ;STORE AT THIS LOCATION
     001672     001                 .BYTE   1       ;MASK
     001673     001                 .BYTE   1       ;HOW MANY TIMES + 2
     001674  000137  001556                 JMP     1$      ;DO IT AGAIN
     001700                         OUTMUL:
     001700  104403                         INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
     001702  015470                         MLEVEL  ;MESSAGE
```

```
 001704  104405                                   PWRAM   ;CONVERT STRING
 001706  000004                                   4       ;LOW LIMIT
 001710  000007                                   7       ;HIGH LIMIT
 001712  016366                                   DUPRT   ;STORE AT THIS LOCATION
 001714     000                          .BYTE    0       ;MASK
 001715     001                          .BYTE    1       ;HOW MANY TIMES + 2
 001716  004737  016316                           JSR     PC,DULEV
                                                  ;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
                                                  ;BUFFER TO THE CHARACTERS "1" AND "2".
                                                  ;IF THE CHARACTER IS "1" CLEAR THE FLAG
                                                  ;IF THE CHARACTER IS "2" SET THE FLAG
 001722                                  AAA:
 001722  104403                                   INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
 001724  015515                                   MSYNC   ;MESSAGE
 001726  122737  000061  016150          3$:      CMPB    #'1,INBUF         ;IS IT "1" ?
 001734  001003                                   BNE     1$
 001736  105037  001172                           CLRB    SYNCNO  ;000
 001742  000412                                   BR      4$
 001744  122737  000062  016150          1$:      CMPB    #'2,INBUF         ;IS IT "2" ?
 001752  001004                                   BNE     2$
 001754  112737  177777  001172                   MOVB    #-1,SYNCNO        ;377
 001762  000402                                   BR      4$
 001764  104404                          2$:      INSTER  ;RETRY
 001766  000757                                   BR      3$
 001770  000240                          4$:      NOP
 001772  104403                                   INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
 001774  015563                                   MWIRE6  ;MESSAGE
 001776  104411                                   SETFLG  ;SET FLAG BASED UPON INPUT STRING
 002000  001173                                   SEXMIT  ;THIS FLAG
 002002  104403                                   INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
 002004  015631                                   MWIRE5  ;MESSAGE
 002006  104411                                   SETFLG  ;SET FLAG BASED UPON INPUT STRING
 002010  001174                                   SEREC   ;THIS FLAG
 002012  104403                                   INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
 002014  015676                                   MWIRE4  ;MESSAGE
 002016  104411                                   SETFLG  ;SET FLAG BASED UPON INPUT STRING
 002020  001175                                   OPTCLR  ;THIS FLAG
 002022  104403                                   INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
 002024  015752                                   MEXTJ   ;MESSAGE
 002026  104411                                   SETFLG  ;SET FLAG BASED UPON INPUT STRING
 002030  001177                                   JMRBY   ;THIS FLAG

                                                  ;TEST START AND RESTART

 002032  012737  000340  177776  .BEGIN:  MOV     #340,PS                  ;LOCK OUT INTERRUPTS
 002040  012706  001100                   MOV     #STACK,SP                ;SET UP STACK
 002044  005737  000042                   TST     @#42                     ;IS PROGRAM UNDER MONITOR CONTROL
 002050  001056                           BNE     3$
 002052  105737  001176                   TSTB    MULTD   ;DON'T ALLOW LOCK ON TEST IF RUNNING
                                                  ;MULTIPLE DEVICES
 002056  001407                           BEQ     5$      ;IF NO,TEST FOR LOCK ON TEST
 002060  013737  013210  013112           MOV     BRW,TTST          ;RESTORE NORMAL SCOPE LOOP
 002066  013737  013212  013114           MOV     BRX,TTST+2        ;DITTO
 002074  000444                           BR      3$                ;JUMP AROUND IF YES
 002076  032737  000004  177570  5$:      BIT     #BIT2,SWR                ;CHECK FOR LOCK ON TEST
 002104  001416                           BEQ     1$
```

```
002106  104403                            INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
002110  015425                            MLOCK   ;MESSAGE
002112  104411                            SETFLG  ;SET FLAG BASED UPON INPUT STRING
002114  001221                            LOKFLG  ;THIS FLAG
002116  105737  001221                    TSTB    LOKFLG                  ;IS LOCK ON TEST OPTION SELECTED
002122  001407                            BEQ     1$
002124  012737  000240  013112            MOV     #NOP,TTST
002132  012737  000240  013114            MOV     #NOP,TTST+2             ;SET UP TO LOCK
002140  000406                            BR      2$
002142  013737  013210  013112  1$:       MOV     BRW,TTST
002150  013737  013212  013114            MOV     BRX,TTST+2              ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
002156  032737  000002  177570  2$:       BIT     #SW01,SWR               ;IF SW01=1, GET STARTING PC
002164  001410                            BEQ     3$
002166  104403                            INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
002170  015412                            MTSTPC  ;MESSAGE
002172  104405                            PARAM   ;CONVERT STRING
002174  002224                            TST1    ;LOW LIMIT
002176  012176                            TLAST   ;HIGH LIMIT
002200  000207                            RETURN  ;STORE AT THIS LOCATION
002202     001                    .BYTE   1       ;MASK
002203     001                    .BYTE   1       ;HOW MANY TIMES + 2
002204  000403                            BR      4$
002206  012737  002224  001110  3$:       MOV     #TST1,RETURN            ;START AT TEST 1
002214  104402  015406          4$:       TYPE    ,MR                     ;TYPE R
002220  000177  176664                    JMP     @RETURN                 ;START TESTING

                                          ;;THIS TEST VERIFYS WORD LENGTH SELECT OF THE
                                          ;;RECEIVER SECTION,IT USES THE ERROR FLAGS
                                          ;;TO DETERMINE THAT IT WAS SELECTED CORRECTLY
                                          ;;(OVRRUN,RXERR)
                                          ;;MODE:ISYMOD
                                          ;;LENGTH:EIGHT
                                          ;;CHAR:125
                                          ;;
002224  012737  000001  001122  TST1:     MOV     #1,TSTNO        ;SAVE THIS
002232  012737  002466  001112            MOV     #TST2,NEXT      ;GO TO THIS TEST WHEN THRU
002240  052777  000400  014570            BIS     #MRESET,@TXCSR  ;MASTER RESET
002246  012777  000000  014556            MOV     #ISYMOD,@PARCSR ;SET THE MODE
002254  052777  000400  014554            BIS     #MRESET,@TXCSR  ;MASTER RESET

                                ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
002262  012777  064001  014546            MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                ;SET MODE , # OF BITS,PARITY SENSE,&LOAD SYNC REG
002270  012777  006J00  014534            MOV     #ISYMOD!EIGHT!NOPAR!0,@PARCSR
002276  052777  000020  014516            BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                          ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
002304  042777  020000  014524            BIC     #CLK,@TXCSR     ;POKE CLK DOWN
002312  052777  020000  014516            BIS     #CLK,@TXCSR     ;POKE CLK UP
                                ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
002320  042777  020000  014510            BIC     #CLK,@TXCSR     ;POKE CLK DOWN
002326  052777  020000  014502            BIS     #CLK,@TXCSR     ;POKE CLK UP
002334  013703  017026                    MOV     RXDBUF,R3       ;SET UP FOR ERROR MESSAGE
002340  012700  000125                    MOV     #125,R0 ;EXPECTED
002344  012737  000012  001134            MOV     #10.,SHIFT      ;# OF SHIFTS
002352  012737  001252  001140            MOV     #1252,TEMP1     ;DATA CHAR
```

```
002360  004737  016526                   JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
002364  105777  014432                   TSTB    @RXCSR  ;RXDONE ?
002370  100401                           BMI     .+4
002372  104000                           HLT     ;RXDONE SHOULD BE SET
002374  017701  014426                   MOV     @RXDBUF,R1      ;ACTUAL
002400  020001                           CMP     R0,R1   ;COMPARE EXPECTED VS. ACTUAL
002402  001401                           BEQ     .+4
002404  104002                           HLT     2               ;RECEIVED DATA DID NOT MATCH
                                                                 ;EXPECTED DATA - CHECK MAINT DATA
                                                                 ;OR RECEIVER LOGIC
002406  012737  000012  001134           MOV     #10.,SHIFT      ;# OF SHIFTS
002414  012737  001252  001140           MOV     #1252,TEMP1     ;DATA CHAR
002422  004737  016526                   JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
                                 ;NOW SHIFT IN A SECOND CHARACTER WITHOUT READING RXDBUF
002426  012737  000012  001134           MOV     #10.,SHIFT      ;# OF SHIFTS
002434  012737  001252  001140           MOV     #1252,TEMP1     ;DATA CHAR
002442  004737  016526                   JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
002446  012700  140125                   MOV     #140000!125,R0  ;EXPECTED DATA PLUS
                                                                 ;RXERR & OVRRUN
002452  017701  014350                   MOV     @RXDBUF,R1      ;ACTUAL
002456  020001                           CMP     R0,R1   ;COMPARE EXP VS. ACT
002460  001401                           BEQ     .+4
002462  104002                           HLT     2               ;SPECIFICALLY LOOK AT RXERR &
                                                                 ;OVRRUN BITS...THEY BOTH SHOULD BE SET
002464  104400                           SCOPE
                                         ;;THIS TEST VERIFYS WORD LENGTH SELECT OF THE
                                         ;;RECEIVER SECTION,IT USES THE ERROR FLAGS
                                         ;;TO DETERMINE THAT IT WAS SELECTED CORRECTLY
                                         ;;(OVRRUN,RXERR)
                                         ;;MODE:ISYMOD
                                         ;;LENGTH:EIGHT
                                         ;;CHAR:252
                                         ;;
002466  012737  000002  001122   TST2:   MOV     #2,TSTNO        ;SAVE THIS
002474  012737  002730  001112           MOV     #TST3,NEXT      ;GO TO THIS TEST WHEN THRU
002502  052777  000400  014326           BIS     #MRESET,@TXCSR  ;MASTER RESET
002510  012777  000000  014314           MOV     #ISYMOD,@PARCSR ;SET THE MODE
002516  052777  000400  014312           BIS     #MRESET,@TXCSR  ;MASTER RESET
                                 ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
002524  012777  064001  014304           MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR
                                 ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
002532  012777  006000  014272           MOV     #ISYMOD!EIGHT!NOPAR!0,@PARCSR
002540  052777  000020  014254           BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                         ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
002546  042777  020000  014262           BIC     #CLK,@TXCSR     ;POKE CLK DOWN
002554  052777  020000  014254           BIS     #CLK,@TXCSR     ;POKE CLK UP
                                 ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
002562  042777  020000  014246           BIC     #CLK,@TXCSR     ;POKE CLK DOWN
002570  052777  020000  014240           BIS     #CLK,@TXCSR     ;POKE CLK UP
002576  013703  017026                   MOV     RXDBUF,R3       ;SET UP FOR ERROR MESSAGE
002602  012700  000252                   MOV     #252,R0 ;EXPECTED
002606  012737  000012  001134           MOV     #10.,SHIFT      ;# OF SHIFTS
002614  012737  001524  001140           MOV     #1524,TEMP1     ;DATA CHAR
002622  004737  016526                   JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
```

```
002626  105777  014170                  TSTB    @RXCSR  ;RXDONE ?
002632  100401                          BMI     .+4
002634  104000                          HLT     ;RXDONE SHOULD BE SET
002636  017701  014164                  MOV     @RXDBUF,R1      ;ACTUAL
002642  020001                          CMP     R0,R1   ;COMPARE EXPECTED VS. ACTUAL
002644  001401                          BEQ     .+4
002646  104002                          HLT     2               ;RECEIVED DATA DID NOT MATCH
                                                                ;EXPECTED DATA - CHECK MAINT DATA
                                                                ;OR RECEIVER LOGIC
002650  012737  000012  001134          MOV     #10.,SHIFT      ;# OF SHIFTS
002656  012737  001524  001140          MOV     #1524,TEMP1     ;DATA CHAR
002664  004737  016526                  JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
                                ;NOW SHIFT IN A SECOND CHARACTER WITHOUT READING RXDBUF
002670  012737  000012  001134          MOV     #10.,SHIFT      ;# OF SHIFTS
002676  012737  001524  001140          MOV     #1524,TEMP1     ;DATA CHAR
002704  004737  016526                  JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
002710  012700  140252                  MOV     #140000!252,R0  ;EXPECTED DATA PLUS
                                                                ;RXERR & OVRRUN
002714  017701  014106                  MOV     @RXDBUF,R1      ;ACTUAL
002720  020001                          CMP     R0,R1   ;COMPARE EXP VS. ACT
002722  001401                          BEQ     .+4
002724  104002                          HLT     2               ;SPECIFICALLY LOOK AT RXERR &
                                                                ;OVRRUN BITS...THEY BOTH SHOULD BE SET
002726  104400                          SCOPE
                                        ;;THIS TEST VERIFYS WORD LENGTH SELECT OF THE
                                        ;;RECEIVER SECTION,IT USES THE ERROR FLAGS
                                        ;;TO DETERMINE THAT IT WAS SELECTED CORRECTLY
                                        ;;(OVRRUN,RXERR)
                                        ;;MODE:SYNEXT
                                        ;;LENGTH:EIGHT
                                        ;;CHAR:252
                                        ;;
002730  012737  000003  001122  TST3:   MOV     #3,TSTNO        ;SAVE THIS
002736  012737  003156  001112          MOV     #TST4,NEXT      ;GO TO THIS TEST WHEN THRU
002744  052777  000400  014064          BIS     #MRESET,@TXCSR  ;MASTER RESET
002752  012777  020000  014052          MOV     #SYNEXT,@PARCSR ;SET THE MODE
002760  052777  000400  014050          BIS     #MRESET,@TXCSR  ;MASTER RESET

                                ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
002766  012777  064001  014042          MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
002774  012777  026000  014030          MOV     #SYNEXT!EIGHT!NOPAR!0,@PARCSR
003002  052777  000020  014012          BIS     #SYNSCH,@RXCSR  ;SET SEARCH SYNC
                                ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
003010  042777  020000  014020          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
003016  052777  020000  014012          BIS     #CLK,@TXCSR     ;POKE CLK UP
003024  013703  017026                  MOV     RXDBUF,R3       ;SET UP FOR ERROR MESSAGE
003030  012700  000252                  MOV     #252,R0 ;EXPECTED
003034  012737  000010  001134          MOV     #8.,SHIFT       ;# OF SHIFTS
003042  012737  000252  001140          MOV     #252,TEMP1      ;DATA CHAR
003050  004737  016526                  JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
003054  105777  013742                  TSTB    @RXCSR  ;RXDONE ?
003060  100401                          BMI     .+4
003062  104000                          HLT     ;RXDONE SHOULD BE SET
003064  017701  013736                  MOV     @RXDBUF,R1      ;ACTUAL
```

```
        003070  020001                                  CMP     R0,R1   ;COMPARE EXPECTED VS. ACTUAL
        003072  001401                                  BEQ     .+4
        003074  104002                                  HLT     2               ;RECEIVED DATA DID NOT MATCH
                                                                                ;EXPECTED DATA - CHECK MAINT DATA
                                                                                ;OR RECEIVER LOGIC
        003076  012737  000010  001134                  MOV     #8.,SHIFT               ;# OF SHIFTS
        003104  012737  000252  001140                  MOV     #252,TEMP1              ;DATA CHAR
        003112  004737  016526                          JSR     PC,RPOKE                ;SHIFT IN THIS CHAR
                                                ;NOW SHIFT IN A SECOND CHARACTER WITHOUT READING RXDBUF
        003116  012737  000010  001134                  MOV     #8.,SHIFT               ;# OF SHIFTS
        003124  012737  000252  001140                  MOV     #252,TEMP1              ;DATA CHAR
        003132  004737  016526                          JSR     PC,RPOKE                ;SHIFT IN THIS CHAR
        003136  012700  140252                          MOV     #140000!252,R0          ;EXPECTED DATA PLUS
                                                                                        ;RXERR & OVRRUN
        003142  017701  013660                          MOV     @RXDBUF,R1              ;ACTUAL
        003146  020001                                  CMP     R0,R1   ;COMPARE EXP VS. ACT
        003150  001401                                  BEQ     .+4
        003152  104002                                  HLT     2               ;SPECIFICALLY LOOK AT RXERR &
                                                                                ;OVRRUN BITS...THEY BOTH SHOULD BE SET
        003154  104400                                  SCOPE
                                                        ;;THIS TEST PERFORMS BINARY DATA CHECK ON THE
                                                        ;;RECEIVER
                                                        ;;LENGTH:EIGHT PLUS PARITY
                                                        ;;MODE:ISYMOD
                                                        ;;PARITY:EVEPAR
                                                        ;;
        003156  012737  000004  001122          TST4:   MOV     #4,TSTNO                ;SAVE THIS
        003164  012737  003356  001112                  MOV     #TST5,NEXT              ;GO TO THIS TEST WHEN THRU
        003172  052777  000400  013636                  BIS     #MRESET,@TXCSR          ;MASTER RESET
        003200  012777  000000  013624                  MOV     #ISYMOD,@PARCSR         ;SET THE MODE
        003206  052777  000400  013622                  BIS     #MRESET,@TXCSR          ;MASTER RESET

                                                ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
        003214  012777  064001  013614                  MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                                ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
        003222  012777  007400  013602                  MOV     #ISYMOD!EIGHT!EVEPAR!0,@PARCSR
        003230  052777  000020  013564                  BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                                        ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
        003236  042777  020000  013572                  BIC     #CLK,@TXCSR             ;POKE CLK DOWN
        003244  052777  020000  013564                  BIS     #CLK,@TXCSR             ;POKE CLK UP
                                                ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
        003252  042777  020000  013556                  BIC     #CLK,@TXCSR             ;POKE CLK DOWN
        003260  052777  020000  013550                  BIS     #CLK,@TXCSR             ;POKE CLK UP
        003266  013703  017026                          MOV     RXDBUF,R3               ;SET UP ERROR MESSAGE
        003272  005004                                  CLR     R4              ;DATA CHAR
        003274  010400                          1$:     MOV     R4,R0           ;EXPECTED
        003276  012737  000013  001134                  MOV     #11.,SHIFT              ;# OF SHIFTS
        003304  010437  001140                          MOV     R4,TEMP1                ;"TO BE SHIFTED CHARACTER"
        003310  004737  016700                          JSR     PC,EVEN8                ;CALC PARITY
        003314  000241                                  CLC
        003316  006137  001140                          ROL     TEMP1   ;GENERATE START BIT
        003322  052737  002000  001140                  BIS     #BIT10,TEMP1    ;GENERATE STOP BIT
                                                ;TEMP1 NOW HAS CHARACTER TO BE POKED INTO RECEIVER
        003330  004737  016526                          JSR     PC,RPOKE                ;SHIFT IN THIS CHAR
        003334  017701  013466                          MOV     @RXDBUF,R1              ;ACTUAL
```

```
003340  020001                                  CMP     R0,R1   ;COMPARE EXP VS ACT
003342  001401                                  BEQ     .+4
003344  104002                                  HLT     2               ;DATA CHARS SHOULD MATCH
                                                                        ;THERE SHOULD BE NO PARITY ERROR
003346  005204                                  INC     R4      ;UPGRADE NEXT CHAR
003350  105704                                  TSTB    R4      ;LAST CHAR ?
003352  001350                                  BNE     1$
003354  104400                                  SCOPE
                                                ;;THIS TEST PERFORMS BINARY DATA CHECK ON THE
                                                ;;RECEIVER
                                                ;;LENGTH:EIGHT PLUS PARITY
                                                ;;MODE:SYNEXT
                                                ;;PARITY:EVEPAR
                                                ;;
003356  012737  000005  001122          TST5:   MOV     #5,TSTNO                ;SAVE THIS
003364  012737  003526  001112                  MOV     #TST6,NEXT              ;GO TO THIS TEST WHEN THRU
003372  052777  000400  013436                  BIS     #MRESET,@TXCSR          ;MASTER RESET
003400  012777  020000  013424                  MOV     #SYNEXT,@PARCSR ;SET THE MODE
003406  052777  000400  013422                  BIS     #MRESET,@TXCSR          ;MASTER RESET

                                        ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
003414  012777  064001  013414                  MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                        ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
003422  012777  027400  013402                  MOV     #SYNEXT!EIGHT!EVEPAR!0,@PARCSR
003430  052777  000020  013364                  BIS     #SYNSCH,@RXCSR  ;SET SEARCH SYNC
                                        ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
003436  042777  020000  013372                  BIC     #CLK,@TXCSR             ;POKE CLK DOWN
003444  052777  020000  013364                  BIS     #CLK,@TXCSR             ;POKE CLK UP
003452  013703  017026                          MOV     RXDBUF,R3               ;SET UP ERROR MESSAGE
003456  005004                                  CLR     R4      ;DATA CHAR
003460  010400                          1$:     MOV     R4,R0   ;EXPECTED
003462  012737  000011  001134                  MOV     #9.,SHIFT               ;# OF SHIFTS
003470  010437  001140                          MOV     R4,TEMP1                ;"TO BE SHIFTED CHARACTER"
003474  004737  016700                          JSR     PC,EVEN8                ;CALC PARITY
                                        ;TEMP1 NOW HAS CHARACTER TO BE POKED INTO RECEIVER
003500  004737  016526                          JSR     PC,RPOKE                ;SHIFT IN THIS CHAR
003504  017701  013316                          MOV     @RXDBUF,R1              ;ACTUAL
003510  020001                                  CMP     R0,R1   ;COMPARE EXP VS ACT
003512  001401                                  BEQ     .+4
003514  104002                                  HLT     2               ;DATA CHARS SHOULD MATCH
                                                                        ;THERE SHOULD BE NO PARITY ERROR
003516  005204                                  INC     R4      ;UPGRADE NEXT CHAR
003520  105704                                  TSTB    R4      ;LAST CHAR ?
003522  001356                                  BNE     1$
003524  104400                                  SCOPE
                                                ;;THIS TEST CHECKS THE STRIP SYNC FUNCTION
                                                ;;OF THE RECEIVER LOGIC
                                                ;;MODE:SYNINT
                                                ;;LENGTH:FIVE
                                                ;;NOTE: RXDONE SHOULD NEVER ASSERT
                                                ;;CHAR: 26 (SYNC)
                                                ;;
003526  012737  000006  001122          TST6:   MOV     #6,TSTNO                ;SAVE THIS
003534  012737  003712  001112                  MOV     #TST7,NEXT              ;GO TO THIS TEST WHEN THRU
003542  052777  000400  013266                  BIS     #MRESET,@TXCSR          ;MASTER RESET
```

```
003550  012777  030000  013254           MOV     #SYNINT,@PARCSR  ;SET THE MODE
003556  052777  000400  013252           BIS     #MRESET,@TXCSR   ;MASTER RESET

                                 ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
003564  012777  064001  013244           MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                 ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
003572  012777  030026  013232           MOV     #SYNINT!FIVE!NOPAR!26,@PARCSR
003600  052777  000020  013214           BIS     #SYNSCH,@RXCSR   ;SET SYNC SEARCH
                                         ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
003606  042777  020000  013222           BIC     #CLK,@TXCSR      ;POKE CLK DOWN
003614  052777  020000  013214           BIS     #CLK,@TXCSR      ;POKE CLK UP
                                 ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
003622  042777  020000  013206           BIC     #CLK,@TXCSR      ;POKE CLK DOWN
003630  052777  020000  013200           BIS     #CLK,@TXCSR      ;POKE CLK UP
003636  052777  000400  013156           BIS     #STPSYN,@RXCSR   ;SET STRIP SYNC
003644  012737  000003  001136           MOV     #3,COUNT         ;# OF SYNC CHARS
003652  012737  000026  001140   1$:     MOV     #26,TEMP1        ;CHAR TO BE SHIFTED
003660  012737  000005  001134           MOV     #5,SHIFT         ;# OF SHIFTS
003666  004737  016526                   JSR     PC,RPOKE         ;SHIFT IN THIS CHAR
003672  105777  013124                   TSTB    @RXCSR   ;RXDONE ?
003676  100001                           BPL     .+4
003700  104000                           HLT     ;RXDONE SHOULD NOT BE ASSERTED
003702  005337  001136                   DEC     COUNT    ;# OF SYNC CHARS
003706  001361                           BNE     1$
003710  104400                           SCOPE
                                         ;;THIS TEST CHECKS THE STRIP SYNC FUNCTION
                                         ;;OF THE RECEIVER LOGIC
                                         ;;MODE:SYNINT
                                         ;;LENGTH:EIGHT
                                         ;;NOTE: RXDONE SHOULD NEVER ASSERT
                                         ;;CHAR: 26 (SYNC)
                                         ;;
003712  012737  000007  001122   TST7:   MOV     #7,TSTNO         ;SAVE THIS
003720  012737  004076  001112           MOV     #TST8,NEXT       ;GO TO THIS TEST WHEN THRU
003726  052777  000400  013102           BIS     #MRESET,@TXCSR   ;MASTER RESET
003734  012777  030000  013070           MOV     #SYNINT,@PARCSR  ;SET THE MODE
003742  052777  000400  013066           BIS     #MRESET,@TXCSR   ;MASTER RESET

                                 ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
003750  012777  064001  013060           MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                 ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
003756  012777  036026  013046           MOV     #SYNINT!EIGHT!NOPAR!26,@PARCSR
003764  052777  000020  013030           BIS     #SYNSCH,@RXCSR   ;SET SYNC SEARCH
                                         ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
003772  042777  020000  013036           BIC     #CLK,@TXCSR      ;POKE CLK DOWN
004000  052777  020000  013030           BIS     #CLK,@TXCSR      ;POKE CLK UP
                                 ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
004006  042777  020000  013022           BIC     #CLK,@TXCSR      ;POKE CLK DOWN
004014  052777  020000  013014           BIS     #CLK,@TXCSR      ;POKE CLK UP
004022  052777  000400  012772           BIS     #STPSYN,@RXCSR   ;SET STRIP SYNC
004030  012737  000003  001136           MOV     #3,COUNT         ;# OF SYNC CHARS
004036  012737  000026  001140   1$:     MOV     #26,TEMP1        ;CHAR TO BE SHIFTED
004044  012737  000010  001134           MOV     #8.,SHIFT        ;# OF SHIFTS
004052  004737  016526                   JSR     PC,RPOKE         ;SHIFT IN THIS CHAR
```

```
004056  105777  012740                      TSTB    @RXCSR  ;RXDONE ?
004062  100001                              BPL     .+4
004064  104000                              HLT     ;RXDONE SHOULD NOT BE ASSERTED
004066  005337  001136                      DEC     COUNT   ;# OF SYNC CHARS
004072  001361                              BNE     1$
004074  104400                              SCOPE
                                ;;THIS TEST PROVES THAT RXERR FREEZES THE "RECEIVER RESET"
                                ;;WHILE IN STRIP SYNC MODE
                                ;;THIS TEST FIRST PROVES THAT AUTOMATIC RESETS OCCUR WHEN
                                ;;STRIP SYNC IS SET & SYNC CHARACTERS ARE SENT
                                ;;.....BUT IF AN ERROR SHOULD OCCUR....THIS AUTOMATIC RESET
                                ;;IS DISCOMBOBULATED
                                ;;IE. FORCE PARITY ERROR WHILE STRIP SYNC IS SET
                                ;;NOTE: NORMALLY THE LOGIC RESETS THE RXDONE &ERROR FLAGS
                                ;;PROVIDING THAT ONLY GOOD SYNC CHARACTERS ARE SENT....
                                ;;BUT, IF AN RXERR OCCURS   RXDONE PLUS RXERR ARE ASSERTED
                                ;;MODE: ISOC (ISYMOD)
                                ;;LENGTH: EIGHT
                                ;;PARITY: EVEPAR
                                ;;CHARACTER EXPECTED:26
                                ;;CHARACTER SENT: SYNC CHARACTER
                                ;;NOTE: THIS TEST USES ONLY THE RECEIVER LOGIC
                                ;;
004076  012737  000010  001122  TST8:   MOV     #8,TSTNO        ;SAVE THIS
004104  012737  004500  001112          MOV     #TST9,NEXT      ;GO TO THIS TEST WHEN THRU
004112  052777  000400  012716          BIS     #MRESET,@TXCSR  ;MASTER RESET
004120  012777  000000  012704          MOV     #ISYMOD,@PARCSR ;SET THE MODE
004126  052777  000400  012702          BIS     #MRESET,@TXCSR  ;MASTER RESET

                                ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
004134  012777  064001  012674          MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                ;SET MODE ,# OF BITS,PARITY SENSE &LOAD SYNC REG
004142  012777  007426  012662          MOV     #ISYMOD!EIGHT!EVEPAR!26,@PARCSR
004150  013703  017026                  MOV     RXDBUF,R3       ;SET UP FOR ERROR MSG
004154  012737  000003  001136          MOV     #3,COUNT        ;# OF TIMES SYNC CHAR WILL BE SENT
004162  052777  000020  012632          BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                        ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
004170  042777  020000  012640          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
004176  052777  020000  012632          BIS     #CLK,@TXCSR     ;POKE CLK UP
                                ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
004204  042777  020000  012624          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
004212  052777  020000  012616          BIS     #CLK,@TXCSR     ;POKE CLK UP
004220  052777  000400  012574          BIS     #STPSYN,@RXCSR  ;SET STRIP SYNC
004226  012737  000013  001134  2$:     MOV     #11.,SHIFT      ;# OF SHIFTS
004234  012737  003054  001140          MOV     #3054,TEMP1     ;SYNC CHAR + START&STOP+ PARITY
004242  004737  016526          1$:     JSR     PC,RPOKE        ;SHIFT IN THIS CHARACTER
004246  105777  012550                  TSTB    @RXCSR  ;RXDONE = 0 ?
004252  100001                          BPL     .+4
004254  104000                          HLT     ;RXDONE SHOULD NOT BE SET
004256  005337  001136                  DEC     COUNT   ;# OF  SYNC CHARS
004262  001361                          BNE     2$      ;GO AGAIN ?
004264  012700  000026                  MOV     #26,R0  ;EXPECTED
004270  017701  012532                  MOV     @RXDBUF,R1      ;ACTUAL
                                                        ;NOTE THAT THIS IS THE FIRST TIME
                                                        ;RXDBUF IS READ.......THERE SHOULD BE
```

```
                                                         ;NO OVER RUN ERRORS
004274  020001                          CMP     R0,R1   ;COMPARE EXPECTED VS ACTUAL
004276  001401                          BEQ     .+4
004300  104002                          HLT     2       ;DATA CHARS SHOULD COMPARE
                                                        ;THERE SHOULD BE NO RXERR'S
004302  012737  000004  001136          MOV     #4,COUNT        ;# OF TIMES
004310  012700  110026                  MOV     #RXERR!PARER!26,R0      ;EXPECTED
004314  012737  002054  001140          MOV     #2054,TEMP1     ;BAD SYNC CHAR (WRONG PARITY)
004322  012737  000013  001134  3$:     MOV     #11.,SHIFT      ;# OF SHIFTS
004330  004737  016526                  JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
004334  105777  012462                  TSTB    @RXCSR  ;RXDONE = 1?
004340  100401                          BMI     .+4
004342  104000                          HLT     ;RXDONE SHOULD BE SET
004344  017701  012456                  MOV     @RXDBUF,R1      ;ACTUAL DATA
004350  020001                          CMP     R0,R1   ;COMPARE EXP VS ACT
004352  001401                          BEQ     .+4
004354  104002                          HLT     2       ;DID THE RESPECTIVE ERROR STOP THE
                                                ;AUTOMATIC RESSETING OF RXDONE & ERROR FLAGS
                                                ;.....CHECK THIS.....
004356  005337  001136                  DEC     COUNT   ;# OF SYNC CHARS
004362  001445                          BEQ     5$      ;FINISHED ? GET OUT OF TEST
004364  022737  000003  001136          CMP     #3,COUNT        ;# OF SYNC CHARS
004372  001423                          BEQ     6$      ;CHECK FRAME ERROR ?
004374  022737  000002  001136          CMP     #2,COUNT        ;# OF SYNC CHARS
004402  001426                          BEQ     7$      ;CHECK FRAME ERROR & BAD PARITY ?
                                        ;NOPE THEN IT (COUNT) MUST BE = 1  THEREFORE....
004404  012737  000013  001134          MOV     #11.,SHIFT      ;# OF SHIFTS
004412  012737  000054  001140          MOV     #54,TEMP1       ;FRAME & PARITY ERROR
004420  004737  016526                  JSR     PC,RPOKE        ;SHIFT IN THIS CHAR
                                ;NOW DON'T READ THE RXDBUF TO CREATE OVER RUN
004424  012737  000054  001140          MOV     #54,TEMP1       ;FRAME & PARITY ERROR
004432  012700  170026                  MOV     #RXERR!OVRRUN!FRMERR!PARER!26,R0        ;EXPECTED
004436  000137  004322                  JMP     3$      ;DO IT AGAIN
004442  012737  001054  001140  6$:     MOV     #1054,TEMP1     ;BAD STOP BIT  FOR FRAME ERROR
004450  012700  120026                  MOV     #RXERR!FRMERR!26,R0     ;EXPECTED
004454  000137  004322                  JMP     3$      ;DO IT AGAIN
004460  012737  000054  001140  7$:     MOV     #54,TEMP1       ;BAD STOP BIT & PARITY
004466  012700  130026                  MOV     #RXERR!FRMERR!PARER!26,R0       ;EXPECTED
004472  000137  004322                  JMP     3$      ;DO IT AGAIN
004476  104400                  5$:     SCOPE
                                        ;;THIS TEST VERIFYS WORD LENGTH SELECT OF
                                        ;;THE TRANSMITTER SECTION,IT USES THE DNA FLAG
                                        ;;AND BIT WINDOW TO DETERMINE THAT IT WAS SELECTED
                                        ;;CORRECTLY
                                        ;;NOTE: DNA COMES UP ON THE FIRST RISING BIT
                                        ;;EDGE OF THE NEXT CHARACTER IF NO NEW CHARACTER IS
                                        ;;LOADED INTO TXDBUF
                                        ;;MODE:SYNINT
                                        ;;PARITY:NO PARITY
                                        ;;LENGTH:EIGHT
                                        ;;
004500  012737  000011  001122  TST9:   MOV     #9,TSTNO        ;SAVE THIS
004506  012737  004732  001112          MOV     #TST10,NEXT     ;GO TO THIS TEST WHEN THRU
004514  052777  000400  012314          BIS     #MRESET,@TXCSR  ;MASTER RESET
004522  012777  030000  012302          MOV     #SYNINT,@PARCSR ;SET THE MODE
004530  052777  000400  012300          BIS     #MRESET,@TXCSR  ;MASTER RESET
```

```
                                        ;SET MAINTENANCE MODE & SEND
                                                ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
004536  012777  004020  012272                  MOV     #MINT!SEND,@TXCSR

                                        ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
004544  012777  036026  012260                  MOV     #SYNINT!EIGHT!NOPAR!26,@PARCSR
004552  013703  017036                          MOV     TXCSR,R3        ;SET UP FOR ERROR MSG
004556  112777  000021  012256                  MOVB    #21,@TXDBUF     ;LOAD CHAR
004564  012737  000021  001140                  MOV     #21,TEMP1       ;SHIFTED CHAR
004572  012737  000010  001134                  MOV     #8.,SHIFT       ;# OF SHIFTS
                                        ;POKE CLK TO GET INTO SYNCRONIZATION
004600  052777  020000  012230                  BIS     #CLK,@TXCSR     ;POKE CLK UP
004606  042777  020000  012222                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
004614  005000                          1$:     CLR     R0
004616  006037  001140                          ROR     TEMP1   ;FORCE CARRY
004622  103002                                  BCC     2$
004624  052700  002000                          BIS     #BITW,R0        ;EQUIV OF BIT WINDOW
004630                                  2$:
004630  052777  020000  012200                  BIS     #CLK,@TXCSR     ;POKE CLK UP
004636  042777  020000  012172                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
004644  017701  012166                          MOV     @TXCSR,R1       ;ACTUAL
004650  042701  075777                          BIC     #075777,R1      ;SAVE BITW & DNA
004654  020001                                  CMP     R0,R1   ;COMPARE EXP VS ACT
004656  001401                                  BEQ     .+4
004660  104003                                  HLT     3       ;BIT WINDOW DID NOT MATCH ACTUAL DATA
                                                                ;BIT.....ALSO CHECK DNA
004662  005337  001134                          DEC     SHIFT   ;# OF SHIFTS
004666  001352                                  BNE     1$      ;DO IT AGAIN ?
                                        ;NOW POKE CLK TO SEE DNA
004670  052777  020000  012140                  BIS     #CLK,@TXCSR     ;POKE CLK
004676  012700  100000                          MOV     #100000,R0      ;EXPECTED
004702  017701  012130                          MOV     @TXCSR,R1       ;ACTUAL
004706  042701  077777                          BIC     #77777,R1       ;SAVE DNA ONLY
004712  020001                                  CMP     R0,R1   ;COMPARE EXPECTED VS ACTUAL
004714  001401                                  BEQ     .+4
004716  104003                                  HLT     3       ;DNA SHOULD BE SET
                                                                ;IF DNA DID NOT SET ,CHECK WORD LENGTH
                                                                ;SELECT LOGIC OF THE TRANSMITTER
004720  005777  012112                          TST     @TXCSR  ;DNA ?
004724  100001                                  BPL     .+4
004726  104000                                  HLT             ;DNA SHOULD NOT BE SET
                                                                ;IT SHOULD HAVE BEEN CLEARED FROM
                                                                ;PREVIOUS READ

004730  104400                                  SCOPE
                                                ;;THIS TEST VERIFYS CHARACTER PLUS PARITY GENERATION
                                                ;;OF THE TRANSMITTER SECTION.
                                                ;;IT ALSO CHECKS DNA TIMING
                                                ;;MODE:ISYMOD
                                                ;;LENGTH:SEVEN PLUS PARITY
                                                ;;PARITY:EVEPAR
                                                ;;CHARACTER:125
                                                ;;
004732  012737  000012  001122  TST10:  MOV     #10,TSTNO       ;SAVE THIS
004740  012737  005154  001112          MOV     #TST11,NEXT     ;GO TO THIS TEST WHEN THRU
004746  052777  000400  012062          BIS     #MRESET,@TXCSR  ;MASTER RESET
```

```
004754  012777  000000  012050          MOV     #ISYMOD,@PARCSR ;SET THE MODE
004762  052777  000400  012046          BIS     #MRESET,@TXCSR  ;MASTER RESET

                                ;SET MAINTENANCE MODE & SEND
                                        ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
004770  012777  004020  012040          MOV     #MINT!SEND,@TXCSR

                                ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
004776  012777  005426  012026          MOV     #ISYMOD!SEVEN!EVEPAR!26,@PARCSR
005004  013703  017036                  MOV     TXCSR,R3        ;SET UP FOR ERROR MSG
005010  112777  000125  012024          MOVB    #125,@TXDBUF    ;LOAD DATA CHAR
005016  012737  001252  001140          MOV     #1252,TEMP1     ;TO BE SHIFTED CHAR
005024  012737  000012  001134          MOV     #10.,SHIFT      ;# OF SHIFTS
                                ;POKE CLK TO GET INTO SYNCRONIZATION
005032  052777  020000  011776          BIS     #CLK,@TXCSR     ;POKE CLK UP
005040  042777  020000  011770          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005046  005000                  1$:     CLR     R0
005050  006037  001140                  ROR     TEMP1   ;FORCE CARRY
005054  103002                          BCC     2$      ;BR IF CARRY CLR
005056  052700  002000                  BIS     #BITW,R0        ;EQUIV OF BITW
005062                          2$:
005062  052777  020000  011746          BIS     #CLK,@TXCSR     ;POKE CLK UP
005070  042777  020000  011740          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005076  017701  011734                  MOV     @TXCSR,R1       ;ACTUAL
005102  042701  075777                  BIC     #075777,R1      ;SAVE BITW & DNA
005106  020001                          CMP     R0,R1   ;COMPARE EXP VS ACT
005110  001401                          BEQ     .+4
005112  104003                          HLT     3               ;BIT WINDOW DID NOT MATCH ACTUAL DATA
                                                                ;BIT,...ALSO CHECK DNA
005114  005337  001134                  DEC     SHIFT   ;# OF SHIFTS
005120  001352                          BNE     1$      ;DO IT AGAIN ?
                                ;NOW POKE CLK TO SEE DNA
005122  052777  020000  011706          BIS     #CLK,@TXCSR     ;POKE CLK
005130  012700  000000                  MOV     #0,R0   ;EXPECTED
005134  017701  011676                  MOV     @TXCSR,R1       ;ACTUAL
005140  042701  077777                  BIC     #77777,R1       ;SAVE DNA ONLY
005144  020001                          CMP     R0,R1           ;COMPARE EXP VS ACT
005146  001401                          BEQ     .+4
005150  104003                          HLT     3               ;DNA SHOULD BE SET
                                                                ;IF DNA DID NOT SET
                                                                ;CHECK WORD LENGTH SELECT LOGIC

005152  104400                          SCOPE
                                        ;;THIS TEST VERIFYS CHARACTER PLUS PARITY GENERATION
                                        ;;OF THE TRANSMITTER SECTION.
                                        ;;IT ALSO CHECKS DNA TIMING
                                        ;;MODE:SYNINT
                                        ;;LENGTH:EIGHT PLUS PARITY
                                        ;;PARI /:EVEPAR
                                        ;;CHARACTER:125
                                        ;;
005154  012737  000013  001122  TST11:  MOV     #11,TSTNO       ;SAVE THIS
005162  012737  005376  001112          MOV     #TST12,NEXT     ;GO TO THIS TEST WHEN THRU
005170  052777  000400  011640          BIS     #MRESET,@TXCSR  ;MASTER RESET
005176  012777  030000  011626          MOV     #SYNINT,@PARCSR ;SET THE MODE
005204  052777  000400  011624          BIS     #MRESET,@TXCSR  ;MASTER RESET
```

```
                                        ;SET MAINTENANCE MODE & SEND
                                                ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
005212  012777  004020  011616                  MOV     #MINT!SEND,@TXCSR

                                        ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
005220  012777  037426  011604                  MOV     #SYNINT!EIGHT!EVEPAR!26,@PARCSR
005226  013703  017036                          MOV     TXCSR,R3        ;SET UP FOR ERROR MSG
005232  112777  000125  011602                  MOVB    #125,@TXDBUF    ;LOAD DATA CHAR
005240  012737  000125  001140                  MOV     #125,TEMP1      ;TO BE SHIFTED CHAR
005246  012737  000011  001134                  MOV     #9.,SHIFT       ;# OF SHIFTS
                                        ;POKE CLK TO GET INTO SYNCRONIZATION
005254  052777  020000  011554                  BIS     #CLK,@TXCSR     ;POKE CLK UP
005262  042777  020000  011546                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005270  005000                          1$:     CLR     R0
005272  006037  001140                          ROR     TEMP1   ;FORCE CARRY
005276  103002                                  BCC     2$      ;BR IF CARRY CLR
005300  052700  002000                          BIS     #BITW,R0        ;EQUIV OF BITW
005304                                  2$:
005304  052777  020000  011524                  BIS     #CLK,@TXCSR     ;POKE CLK UP
005312  042777  020000  011516                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005320  017701  011512                          MOV     @TXCSR,R1       ;ACTUAL
005324  042701  075777                          BIC     #075777,R1      ;SAVE BITW & DNA
005330  020001                                  CMP     R0,R1   ;COMPARE EXP VS ACT
005332  001401                                  BEQ     .+4
005334  104003                                  HLT     3               ;BIT WINDOW DID NOT MATCH ACTUAL DATA
                                                                        ;BIT,...ALSO CHECK DNA
005336  005337  001134                          DEC     SHIFT   ;# OF SHIFTS
005342  001352                                  BNE     1$      ;DO IT AGAIN ?
                                        ;NOW POKE CLK TO SEE DNA
005344  052777  020000  011464                  BIS     #CLK,@TXCSR     ;POKE CLK
005352  012700  100000                          MOV     #100000,R0      ;EXPECTED
005356  017701  011454                          MOV     @TXCSR,R1       ;ACTUAL
005362  042701  077777                          BIC     #77777,R1       ;SAVE DNA ONLY
005366  020001                                  CMP     R0,R1           ;COMPARE EXP VS ACT
005370  001401                                  BEQ     .+4
005372  104003                                  HLT     3               ;DNA SHOULD BE SET
                                                                        ;IF DNA DID NOT SET
                                                                        ;CHECK WORD LENGTH SELECT LOGIC

005374  104400                                  SCOPE
                                                ;;THIS TEST VERIFYS CHARACTER PLUS PARITY GENERATION
                                                ;;OF THE TRANSMITTER SECTION.
                                                ;;IT ALSO CHECKS DNA TIMING
                                                ;;MODE:SYNINT
                                                ;;LENGTH:EIGHT PLUS PARITY
                                                ;;PARITY:EVEPAR
                                                ;;CHARACTER:252
                                                ;;
005376  012737  000014  001122          TST12:  MOV     #12,TSTNO       ;SAVE THIS
005404  012737  005620  001112                  MOV     #TST13,NEXT     ;GO TO THIS TEST WHEN THRU
005412  052777  000400  011416                  BIS     #MRESET,@TXCSR  ;MASTER RESET
005420  012777  030000  011404                  MOV     #SYNINT,@PARCSR ;SET THE MODE
005426  052777  000400  011402                  BIS     #MRESET,@TXCSR  ;MASTER RESET

                                        ;SET MAINTENANCE MODE & SEND
                                                ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
005434  012777  004020  011374                  MOV     #MINT!SEND,@TXCSR
```

```
                                            ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
005442  012777  037426  011362                      MOV     #SYNINT!EIGHT!EVEPAR!26,@PARCSR
005450  013703  017036                              MOV     TXCSR,R3        ;SET UP FOR ERROR MSG
005454  112777  000252  011360                      MOVB    #252,@TXDBUF    ;LOAD DATA CHAR
005462  012737  000252  001140                      MOV     #252,TEMP1      ;TO BE SHIFTED CHAR
005470  012737  000011  001134                      MOV     #9.,SHIFT       ;# OF SHIFTS
                                            ;POKE CLK TO GET INTO SYNCRONIZATION
005476  052777  020000  011332                      BIS     #CLK,@TXCSR     ;POKE CLK UP
005504  042777  020000  011324                      BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005512  005000                              1$:     CLR     R0
005514  006037  001140                              ROR     TEMP1   ;FORCE CARRY
005520  103002                                      BCC     2$      ;BR IF CARRY CLR
005522  052700  002000                              BIS     #BITW,R0        ;EQUIV OF BITW
005526                                      2$:
005526  052777  020000  011302                      BIS     #CLK,@TXCSR     ;POKE CLK UP
005534  042777  020000  011274                      BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005542  017701  011270                              MOV     @TXCSR,R1       ;ACTUAL
005546  042701  075777                              BIC     #075777,R1      ;SAVE BITW & DNA
005552  020001                                      CMP     R0,R1   ;COMPARE EXP VS ACT
005554  001401                                      BEQ     .+4
005556  104003                                      HLT     3               ;BIT WINDOW DID NOT MATCH ACTUAL DATA
                                                                            ;BIT....ALSO CHECK DNA
005560  005337  001134                              DEC     SHIFT   ;# OF SHIFTS
005564  001352                                      BNE     1$      ;DO IT AGAIN ?
                                            ;NOW POKE CLK TO SEE DNA
005566  052777  020000  011242                      BIS     #CLK,@TXCSR     ;POKE CLK
005574  012700  100000                              MOV     #100000,R0      ;EXPECTED
005600  017701  011232                              MOV     @TXCSR,R1       ;ACTUAL
005604  042701  077777                              BIC     #77777,R1       ;SAVE DNA ONLY
005610  020001                                      CMP     R0,R1           ;COMPARE EXP VS ACT
005612  001401                                      BEQ     .+4
005614  104003                                      HLT     3               ;DNA SHOULD BE SET
                                                                            ;IF DNA DID NOT SET
                                                                            ;CHECK WORD LENGTH SELECT LOGIC

005616  104400                                      SCOPE
                                            ;;THIS TEST VERIFYS THAT BY SENDING ONLY ONE SYNC
                                            ;;CHARACTER (TWO SELECTED BY STRAPPING ) RECACT =0
                                            ;;THEN SEND ONE ORDINARY CHARACTER (TO BREAK UP THE SEQUENCE)
                                            ;;RECACT =0..............IT WILL TAKE TWO MORE SYNC CHARS
                                            ;;BEFORE RECACT =1
                                            ;;NOTE: THIS TEST WILL ONLY WORK WHEN TWO SYNC CHARS
                                            ;;HAS BEEN BEEN SELECTED ...OTHERWISE JUMP AROUND THIS TEST
                                            ;;MODE:SYNC INTERNAL (SYNINT)
                                            ;;PARITY: NOPAR
                                            ;;LENGTH: EIGHT
                                            ;;
                                            ;;THIS TEST CHECKS ONLY THE RECEIVER SECTION
                                            ;;
005620  012737  000015  001122              TST13:  MOV     #13,TSTNO       ;SAVE THIS
005626  012737  006122  001112                      MOV     #TST14,NEXT     ;GO TO THIS TEST WHEN THRU
005634  105737  001172                              TSTB    SYNCNO  ;TEST FOR # OF SYNC CHARS REQUIRED
005640  100127                                      BPL     2$      ;IF NOT TWO  GET OUT OF TEST
005642  052777  000400  011166                      BIS     #MRESET,@TXCSR  ;MASTER RESET
005650  012777  030000  011154                      MOV     #SYNINT,@PARCSR ;SET THE MODE
005656  052777  000400  011152                      BIS     #MRESET,@TXCSR  ;MASTER RESET
```

```

                                             ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
005664  012777  064001  011144                       MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                             ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
005672  012777  036026  011132                       MOV     #SYNINT!EIGHT!NOPAR!26,@PARCSR
005700  052777  000020  011114                       BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                                     ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
005706  042777  020000  011122                       BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005714  052777  020000  011114                       BIS     #CLK,@TXCSR     ;POKE CLK UP
                                             ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
005722  042777  020000  011106                       BIC     #CLK,@TXCSR     ;POKE CLK DOWN
005730  052777  020000  011100                       BIS     #CLK,@TXCSR     ;POKE CLK UP
005736  012737  000010  001134                       MOV     #8.,SHIFT       ;# OF SHIFTS
005744  012737  000026  001140                       MOV     #26,TEMP1       ;SYNC CHAR TO BE SHIFTED IN
005752  004737  016526                               JSR     PC,RPOKE        ;SHIFT IN THIS SYNC CHAR
005756  032777  004000  011036                       BIT     #RECACT,@RXCSR  ;RECACT = 0 ?
005764  001401                                       BEQ     .+4
005766  104000                                       HLT     ;RECACT SHOULD BE 0
005770  012737  000010  001134                       MOV     #8.,SHIFT       ;# OF SHIFTS
005776  012737  000025  001140                       MOV     #25,TEMP1       ;ANY CHARACTER
006004  004737  016526                               JSR     PC,RPOKE        ;SHIFT IN THIS CHARACTER
                                             ;YOU HAVE JUST LOST SYNCRONIZATION....................
                                             ;POKE THE CLK TWICE TO GET INTO SYNCRONIZATION
                                             ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
006010  042777  020000  011020                       BIC     #CLK,@TXCSR     ;POKE CLK DOWN
006016  052777  020000  011012                       BIS     #CLK,@TXCSR     ;POKE CLK UP
                                             ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
006024  042777  020000  011004                       BIC     #CLK,@TXCSR     ;POKE CLK DOWN
006032  052777  020000  010776                       BIS     #CLK,@TXCSR     ;POKE CLK UP
006040  012737  000002  001136                       MOV     #2,COUNT        ;# OF SYNC CHARS
006046  032777  004000  010746               1$:     BIT     #RECACT,@RXCSR  ;RECACT = 0 ?
006054  001401                                       BEQ     .+4
006056  104000                                       HLT             ;RECACT SHOULD BE  0
006060  012737  000010  001134                       MOV     #8.,SHIFT       ;# OF SHIFTS
006066  012737  000026  001140                       MOV     #26,TEMP1       ;SYNC CHAR
006074  004737  016526                               JSR     PC,RPOKE        ;SHIFT IN THIS SYNC CHAR
006100  005337  001136                               DEC     COUNT
006104  001360                                       BNE     1$      ;IS COUNT = 0 ? NO GO AGAIN
006106  032777  004000  010706                       BIT     #RECACT,@RXCSR  ;RECACT = 1 ?
006114  001001                                       BNE     .+4
006116  104000                                       HLT     ;RECACT SHOULD BE ASSERTED
006120  104400                               2$:     SCOPE
                                              ;;THIS TEST VERIFYS TX DONE FUNCTION, DONE = 1
                                              ;;ALSO   VERIFYS THAT THE TRANSMITTER CHIP IDLES "SYNC" CHARACTER
                                              ;;WHEN NO NEW CHARACTER IS LOADED INTO TXDBUF("SYNC" = BINARY COUNT PATTERN)
                                              ;;MODE:  SYNC INTERNAL
                                              ;;PARITY:  NO PARITY (NOPAR)
                                              ;;LENGTH:  EIGHT
                                              ;;
006122  012737  000016  001122               TST14:  MOV     #14,TSTNO       ;SAVE THIS
006130  012737  006420  001112                       MOV     #TST15,NEXT     ;GO TO THIS TEST WHEN THRU
006136  012737  006322  001114                       MOV     #6$,LOCK        ;SET UP FOR SCOPE LOOP
006144  012704  036000                               MOV     #SYNINT!EIGHT!NOPAR!0,R4        ;MODE ETC.
006150  052777  000400  010660               5$:     BIS     #MRESET,@TXCSR  ;MASTER RESET
006156  012777  030000  010646                       MOV     #SYNINT,@PARCSR ;SET THE MODE
```

```
006164  052777  000400  010644          BIS     #MRESET,@TXCSR   ;MASTER RESET
006172  012777  004020  010636          MOV     #MINT!SEND,@TXCSR
006200  010477  010626                  MOV     R4,@PARCSR
006204  105777  010626                  TSTB    @TXCSR           ;TXDONE?
006210  100401                          BMI     .+4
006212  104000                          HLT                      ;TXDONE SHOULD BE SET
006214  112777  000021  010620          MOVB    #21,@TXDBUF      ;LOAD ANY CHAR
                                        ;POKE   CLK TO GET INTO SYNCIRONIZATION
006222  052777  020000  010606          BIS     #CLK,@TXCSR      ;POKE CLK UP
006230  042777  020000  010600          BIC     #CLK,@TXCSR      ;POKE CLK DOWN
006236  105777  010574                  TSTB    @TXCSR
006242  100001                          BPL     .+4
006244  104000                          HLT                      ;TXDONE SHOULD BE CLR

006246  052777  020000  010562          BIS     #CLK,@TXCSR      ;POKE CLK UP
006254  042777  020000  010554          BIC     #CLK,@TXCSR      ;POKE CLK DOWN
006262  105777  010550                  TSTB    @TXCSR
006266  100401                          BMI     .+4
006270  104000                          HLT                      ;TXDONE SHOULD BE SET
006272  012737  000007  001134          MOV     #7,SHIFT
006300                          1$:
006300  052777  020000  010530          BIS     #CLK,@TXCSR      ;POKE CLK UP
006306  042777  020000  010522          BIC     #CLK,@TXCSR      ;POKE CLK DOWN
006314  005337  001134                  DEC     SHIFT
006320  001367                          BNE     1$      ;SHIFT OUT THE "21"
006322  013703  017036          6$:     MOV     TXCSR,R3         ;FOR ERROR MESSAGE
                                                ;THE BIT WINDOW IS RE GENERATED INTO
                                                ;A CHARACTER AND LEFT PRESENTED IN R1
                                                ;FOR THE COMPARE OPERATION. IF YOU WANT TO
                                                ;LOCK ON A PARTICULAR SYNC CHARACTER...
                                                ;SET SWR09=1
006326  005000                          CLR     R0
006330  150400                          BISB    R4,R0   ;EXPECT "SYNC"
006332  012737  000010  001134          MOV     #8.,SHIFT        ;# OF SHIFTS
006340  005001                          CLR     R1
006342                          3$:
006342  052777  020000  010466          BIS     #CLK,@TXCSR      ;POKE CLK UP
006350  042777  020000  010460          BIC     #CLK,@TXCSR      ;POKE CLK DOWN
006356  000241                          CLC
006360  032777  002000  010450          BIT     #BITW,@TXCSR     ;BITW = ?
006366  001401                          BEQ     2$
006370  000261                          SEC     ;SET CARRY
006372  106001                  2$:     RORB    R1      ;PICK UP CARRY
006374  005337  001134                  DEC     SHIFT
006400  001360                          BNE     3$      ;FINISH THAT CHARACTER
006402  020001                          CMP     R0,R1   ;CMP EXPECTED VS ACTUAL
006404  001401                          BEQ     .+4
006406  104003                          HLT     3       ;SYNC CHAR IS NOT CORRECT
006410  104401                          SCOP1
006412  105204                          INCB    R4      ;SET UP FOR NEXT SYNC HOLDING REG.
006414  001255                          BNE     5$      ;FINISHED WITH BINARY COUNT PATTERN ?

006416  104400                          SCOPE

                                ;;THIS TEST PROVES THAT RXERR FREEZES THE "RECEIVER
                                ;;RESET" WHILE IN STRIP SYNC MODE
```

```
                                        ;;THIS TEST FIRST PROVES THAT AUTOMATIC RESETS OCCUR
                                        ;;WHEN STRIP SYNC IS SET AND SYNC CHARACTERS ARE SENT
                                        ;;BUT IF AN ERROR SHOULD OCCUR...THIS AUTOMATIC RESET
                                        ;;IS DISCOMBOBULATED
                                        ;;IE:  FORCE OVERRUN (OVERRUN) WHILE STRIP SYNC IS SET
                                        ;;BY TRANSMITTING A DATA CHARACTER THEN TRANSMIT A SYNC CHARACTER
                                        ;;AND DON'T READ THAT DATA CHARACTER.  NOTE:  NORMALLY THE LOGIC
                                        ;;RESETS THE RXDONE & ERROR FLAGS PROVIDING THAT ONLY SYNC CHARACTERS ARE
                                        ;;STRIPPED
                                        ;;MODE:  SYNC EXTERNAL (SYNEXT)
                                        ;;LENGTH:  EIGHT
                                        ;;NOTE:  THIS TEST USES BOTH RECEIVER AND TRANSMITTER LOGIC
                                        ;;

006420  012737  000017  001122  TST15:  MOV     #15,TSTNO         ;SAVE THIS
006426  012737  006722  001112          MOV     #TST16,NEXT       ;GO TO THIS TEST WHEN THRU

006434  052777  000400  010374          BIS     #MRESET,@TXCSR    ;MASTER RESET
006442  012777  020000  010362          MOV     #SYNEXT,@PARCSR   ;SET THE MODE
006450  052777  000400  010360          BIS     #MRESET,@TXCSR    ;MASTER RESET

                                ;SET MAINTENANCE MODE & SEND
                                        ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
006456  012777  004020  010352          MOV     #MINT!SEND,@TXCSR

                                ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
006464  012777  026026  010340          MOV     #SYNEXT!EIGHT!NOPAR!26,@PARCSR
006472  112777  000026  010342          MOVB    #26,@TXDBUF       ;LOAD SYNC CHAR
006500  052777  000420  010314          BIS     #SYNSCH!STPSYN,@RXCSR   ;SET SYNC SEARCH & STRIP SYNC
```

```
006506  013703  017026                  MOV     RXDBUF,R3       ;SET UP FOR ERROR MESSAGE
006512  012737  000003  001136          MOV     #3,COUNT        ;# OF TIMES SYNC WILL BE SENT
006520  052777  020000  010310          BIS     #CLK,@TXCSR     ;POKE CLK UP
006526  042777  020000  010302          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
006534  012737  000010  001134  2$:     MOV     #8.,SHIFT       ;# OF SHIFTS
006542                          1$:
006542  052777  020000  010266          BIS     #CLK,@TXCSR     ;POKE CLK UP
006550  042777  020000  010260          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
006556  005337  001134                  DEC     SHIFT
006562  001367                          BNE     1$
006564  105777  010232                  TSTB    @RXCSR          ;RXDONE?
006570  100001                          BPL     .+4
006572  104000                          HLT                     ;RXDONE SHOULD NOT ASSERT
006574  005337  001136                  DEC     COUNT
006600  001355                          BNE     2$
006602  012700  000026                  MOV     #26,R0          ;EXPECTED
006606  017701  010214                  MOV     @RXDBUF,R1      ;ACTUAL
006612  020001                          CMP     R0,R1
006614  001401                          BEQ     .+4
006616  104002                          HLT     2               ;NOTE THAT OVERRUN SHOULD NOT OCCUR, ALSO
                                                                ;SECOND & 3RD SYNC CHARACTER CAME FROM
                                                                ;SYNC HOLDING REGISTER
006620  012737  000003  001136          MOV     #3,COUNT        ;# OF TIMES
006626  112777  000025  010206          MOVB    #25,@TXDBUF     ;LOAD ANY CHAR....HOWEVER...
                                                ;ONE MORE SYNC CHAR WILL BE SENT BEFORE
                                                ;THE "25" CHAR IS SENT (THE DNA BIT IS
                                                ;ALREADY UP)

006634  012737  000010  001134  4$:     MOV     #8.,SHIFT       ;# OF SHIFTS

006642                          3$:
006642  052777  020000  010166          BIS     #CLK,@TXCSR     ;POKE CLK UP
006650  042777  020000  010160          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
006656  005337  001134                  DEC     SHIFT
006662  001367                          BNE     3$
006664  005337  001136                  DEC     COUNT
006670  001361                          BNE     4$
006672  105777  010124                  TSTB    @RXCSR  ;RXDONE = 1 ?
006676  100401                          BMI     .+4
006700  104000                          HLT     ;RXDONE SHOULD BE SET
006702  012700  140026                  MOV     #RXERR!OVRRUN!26,R0 ;EXPECTED
006706  017701  010114                  MOV     @RXDBUF,R1      ;ACTUAL
006712  020001                          CMP     R0,R1
006714  001401                          BEQ     .+4
006716  104002                          HLT     2               ;NOTE THAT OVRRUN SHOULD OCCUR,
                                                                ;ALSO SECOND SYNC CHARACTER CAME
                                                                ;FROM SYNC HOLDING REGISTER
```

G04


```
                                                   ;SUMMARY:  THE OVRRUN STOPPED
                                                   ;THE AUTOMATIC RESETTING OF
                                                   ;RXDONE & ERROR FLAGS.....CHECK THIS
006720  104400                              SCOPE
                                  ;;THIS TEST VERIFYS THAT BY DROPPING SYNSCN
                                  ;;IN THE MIDDLE OF A CHARACTER, SYNC CHARACTER SEQUENCE
                                  ;;IS NEEDED BEFORE RECACT, RXDONE ASSERT AGAIN.
                                  ;;ALSO NOTE:  SINCE RECACT IS DEPENDENT ON MATCH DETECT,
                                  ;;AND IF SYNSCH IS DROPPED IN THE MIDDLE OF
                                  ;;A SYNC CHARACTER AND THEN RAISED AGAIN ;RXDONE SHOULD
                                  ;;NOT ASSERT UNTIL NEW SYNC CHARACTER SEQUENCE
                                  ;;MODE:  SYNC INTERNAL (SYNINT)
                                  ;;LENGTH:  EIGHT
                                  ;;

006722  012737  000020  001122    TST16:  MOV     #16,TSTNO        ;SAVE THIS
006730  012737  007452  001112            MOV     #TST17,NEXT      ;GO TO THIS TEST WHEN THRU

006736  105737  001172                    TSTB    SYNCNO           ;TWO SYNC CHARACTERS SELECTED  ?
006742  100002                            BPL     .+6     ;IF ANSWER WAS NO  DO THIS TEST
006744  000137  007450                    JMP     4$      ;IF ANSWER WAS YES  JUMP OVER THIS TEST
006750  052777  000400  010060            BIS     #MRESET,@TXCSR  ;MASTER RESET
006756  012777  030000  010046            MOV     #SYNINT,@PARCSR ;SET THE MODE
006764  052777  000400  010044            BIS     #MRESET,@TXCSR  ;MASTER RESET

                                  ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
006772  012777  064001  010036            MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR

                                  ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
007000  012777  036026  010024            MOV     #SYNINT!EIGHT!NOPAR!26,@PARCSR

007006  052777  000020  010006            BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                          ;POKE CLK TO GET RECEIVER INTO SYNCROIZATION....
007014  042777  020000  010014            BIC     #CLK,@TXCSR      ;POKE CLK DOWN
007022  052777  020000  010006            BIS     #CLK,@TXCSR      ;POKE CLK UP
                                  ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
007030  042777  020000  010000            BIC     #CLK,@TXCSR      ;POKE CLK DOWN
007036  052777  020000  007772            BIS     #CLK,@TXCSR      ;POKE CLK UP
007044  012737  000002  001136            MOV     #2,COUNT         ;# OF TIMES
007052  013703  017026                    MOV     RXDBUF,R3        ;FOR ERROR MESSAGE
007056  012737  000010  001134    1$:     MOV     #8.,SHIFT        ;# OF SHIFTS
007064  012737  000026  001140            MOV     #26,TEMP1        ;SYNC CHAR.
007072  004737  016526                    JSR     PC,RPOKE
007076  005337  001136                    DEC     COUNT
007102  001403                            BEQ     2$
                                  ;TEST TO SEE HOW MANY SYNC CHARACTERS NEEDED
007104  105737  001172                    TSTB    SYNCNO
007110  100762                            BMI     1$

007112  032777  004000  007702    2$:     BIT     #RECACT,@RXCSR  ;RECACT=1?
007120  001001                            BNE     .+4
007122  104000                            HLT                      ;RECACT SHOULD BE SET
007124  012737  000004  001134            MOV     #4,SHIFT         ;# OF SHIFTS
007132  012737  000026  001140            MOV     #26,TEMP1        ;SYNC CHAR.
007140  004737  016526                    JSR     PC,RPOKE
007144  032777  004000  007650            BIT     #RECACT,@RXCSR  ;RECACT=1?
```

```
007152  001001                              BNE     .+4
007154  104000                              HLT                     ;RECACT SHOULD  STILL BE SET
007156  042777  000020  007636              BIC     #SYNSCH,@RXCSR  ;DROP SEARCH SYNC
007164  032777  004000  007630              BIT     #RECACT,@RXCSR  ;RECACT=0?
007172  001401                              BEQ     .+4
007174  104000                              HLT                     ;RECACT SHOULD NOT BE SET
                                    ;NOW SHIFT TWO BITS TO ALLOW SEARCH SYNC =0 TO TAKE
                                    ;EFFECT IN THE LOGIC(THIS ALLOWS THE RECEIVER CHIP TO SEE
                                    ;THE DROPPING OF SEARCH SYNC)...MATCH DETECT IN THE  REC.CHIP SHOULD ALSO DROP
007176  012737  000002  001134              MOV     #2,SHIFT        ;# OF SHIFTS
007204  004737  016526                      JSR     PC,RPOKE
007210  052777  000020  007604              BIS     #SYNSCH,@RXCSR  ;SET SEARCH SYNC
007216  032777  004000  007576              BIT     #RECACT,@RXCSR
007224  001401                              BEQ     .+4
007226  104000                              HLT     ;RECACT =0 ?
007230  105777  007566                      TSTB    @RXCSR
007234  100001                              BPL     .+4
007236  104000                              HLT     ;RXDONE = 0 ?
007240  012737  000002  001134              MOV     #2,SHIFT        ;# OF SHIFTS
007246  004737  016526                      JSR     PC,RPOKE
007252  032777  004000  007542              BIT     #RECACT,@RXCSR  ;RECACT=0?
007260  001401                              BEQ     .+4
007262  104000                              HLT                     ;RECACT SHOULD NOT BE SET
007264  105777  007532                      TSTB    @RXCSR          ;RXDONE=0?
007270  100001                              BPL     .+4
007272  104000                              HLT                     ;RXDONE SHOULD NOT BE ASSERTED
007274  012700  000026                      MOV     #26,R0          ;EXPECTED
007300  017701  007522                      MOV     @RXDBUF,R1      ;ACTUAL
007304  020001                              CMP     R0,R1
007306  001401                              BEQ     .+4
007310  104002                              HLT     2               ;CHARACTERS SHOULD BE MATCHED
007312  012737  000002  001136              MOV     #2,COUNT        ;# OF TIMES OF SYNC CHARS.
                                            ;TEST TO SEE HOW MANY SYNC CHARS  NEEDED
007320  105737  001172                      TSTB    SYNCNO
007324  100402                              BMI     3$      ;WILL IT BE TWO OR ONE ?
007326  005337  001136                      DEC     COUNT   ;IT WAS ONLY ONE NEEDED
007332  012737  000010  001134      3$:     MOV     #8.,SHIFT       ;#OF SHIFTS
007340  012737  000026  001140              MOV     #26,TEMP1       ;SYNC CHAR
007346  004737  016526                      JSR     PC,RPOKE
007352  005337  001136                      DEC     COUNT   ;IS IT THE LAST SYNC CHAR ?
007356  001365                              BNE     3$      ;GO AGAIN AND SHIFT IN ANOTHER SYNC CHAR
007360  032777  004000  007434              BIT     #RECACT,@RXCSR  ;RECACT=1?
007366  001001                              BNE     .+4
007370  104000                              HLT     ;RECACT SHOULD BE ASSERTED
007372  105777  007424                      TSTB    @RXCSR          ;RXDONE=0?
007376  100001                              BPL     .+4
007400  104000                              HLT                     ;RXDONE SHOULD NOT BE ASSERTED
007402  012737  000010  001134              MOV     #8.,SHIFT       ;#OF SHIFTS
007410  012737  000025  001140              MOV     #25,TEMP1       ;ANY CHARACTER
007416  004737  016526                      JSR     PC,RPOKE
007422  105777  007374                      TSTB    @RXCSR          ;RXDONE=1?
007426  100401                              BMI     .+4
007430  104000                              HLT                     ;RXDONE SHOULD NOW BE ASSERTED
007432  012700  000025                      MOV     #25,R0          ;EXPECTED
007436  017701  007364                      MOV     @RXDBUF,R1      ;ACTUAL
007442  020001                              CMP     R0,R1
```

```
007444	001401				BEQ	.+4
007446	104002				HLT	2		;CHARACTERS SHOULD BE MATCHED
							;IF THIS  FAILS THEN CHECK THAT THE CORRECT
							;RECEIVER CHIP IS BEING USED...WHAT IS
							;HAPPENING IS THAT MATCH DETECT IS ASSERTING
							;BEFORE A NEW SYNC CHARACTER SEQUENCE
							;TRANSPIRES  THUS RXDONE ASSERTS TOO SOON
							;AND OVER RUN OCCURS SINCE THE RECEIVER WAS NOT READ
							;CONCLUSION::: IF OLDER RECEIVER CHIP
							;IS BEING USED THEN REPLACE IT WITH A NEW
							;RECEIVER CHIP IF 1 SYNC CHARACTER SEQUENCE IS DESIRED

007450	104400			4$:	SCOPE
				;;THIS TEST VERIFYS THAT DSC CAUSES AN INTERRUPT
				;;THIS TEST ONLY WORKS IN MAINT EXTERNAL MODE
				;;INTERRUPT VECTOR:  DURIV

007452	012737	000021	001122	TST17:	MOV	#17,TSTNO	;SAVE THIS
007460	012737	007652	001112		MOV	#TST18,NEXT	;GO TO THIS TEST WHEN THRU
007466	105737	001177			TSTB	JMRBY		;IN MAINT EXTERNAL?
007472	100066				BPL	3$		;IF ANSWER NO JUMP AROUND TEST
007474	052777	000400	007334		BIS	#MRESET,@TXCSR	;MASTER RESET
007502	012777	007526	007336		MOV	#4$,@DURIV	;SET UP TRAPCATCHER
007510	013777	016366	007332		MOV	DUPRT,@DURIS	;
007516	013737	016370	177776		MOV	LESS1,PS	;ALLOW INTERRUPTS
007524	000423				BR	1$	;JUMP AROUND INTERRUPT SVC ROUTINE
				;THE FOLLOWING IS THE INTERRUPT SVC ROUTINE
007526	012737	000340	177776	4$:	MOV	#LEVEL7,PS	;DON'T ALLOW ANYMORE INTERRUPTS
007534	005777	007262			TST	@RXCSR		;DSC=1?
007540	100401				BMI	.+4
007542	104000				HLT			;FALSE INTERRUPT
007544	042777	000040	007250		BIC	#DSINTE,@RXCSR	;CLEAR INTERRUPT ENABLE
007552	012716	007642			MOV	#2$,(SP)	;SET UP RETURN LOCATION
007556	013777	017050	007262		MOV	DURIS,@DURIV	;RESTORE TRAPCATCHER
007564	012777	000000	007256		MOV	#0,@DURIS	;
007572	000002				RTI

007574	052777	000040	007220	1$:	BIS	#DSINTE,@RXCSR	;SET INTERRUPT ENABLE
007602	052777	000002	007212		BIS	#DTR,@RXCSR	;TRY TO CAUSE INTERRUPT
007610	005000				CLR	R0
007612	005200				INC	R0		;WAIT FOR INTERRUPT
007614	001376				BNE	.-2
007616	013777	017050	007222		MOV	DURIS,@DURIV	;RESTORE TRAPCATCHER
007624	012777	000000	007216		MOV	#0,@DURIS	;

007632	042777	000040	007162		BIC	#DSINTE,@RXCSR	;CLEAR INTERRUPT ENABLE
007640	104000				HLT			;INTERRUPT FAILED TO OCCUR
007642	012737	000340	177776	2$:	MOV	#LEVEL7,PS
007650	104400			3$:	SCOPE
				;;THIS TEST VERIFYS THAT TWO INTERRUPTS THAT TRAP TO
				;;THE SAME VECTOR ARE BOTH EXECUTED
				;;INTERRUPT VECTOR:  DURIV
				;;THIS TEST ONLY WORKS IN MAINT EXTERNAL MODE

007652	012737	000022	001122	TST18:	MOV	#18,TSTNO	;SAVE THIS
007660	012737	010316	001112		MOV	#TST19,NEXT	;GO TO THIS TEST WHEN THRU
007666	105737	001177			TSTB	JMRBY		;IN MAINT. EXTERNAL?
```

```
007672  100402                                  BMI     .+6             ;IF ANSWER WAS YES DO THIS TEST
007674  000137  010314                          JMP     1$              ;IF ANSWER WAS NO JUMP AROUND TEST
007700  052777  000400  007130                  BIS     #MRESET,@TXCSR  ;MASTER RESET
007706  012777  020000  007116                  MOV     #SYNEXT,@PARCSR ;SET THE MODE
007714  052777  000400  007114                  BIS     #MRESET,@TXCSR  ;MASTER RESET
                                        ;SET MAINT DATA,CLK,BREAK,&MAINTENANCE MODE
007722  012777  064001  007106                  MOV     #MTDATA!CLK!MINT!BREAK,@TXCSR
                                        ;SET MODE ,# OF BITS,PARITY SENSE,&LOAD SYNC REG
007730  012777  026026  007074                  MOV     #SYNEXT!EIGHT!NOPAR!26,@PARCSR
007736  052777  000020  007056                  BIS     #SYNSCH,@RXCSR  ;SET SEARCH SYNC
                                        ;POKE CLK TO GET LOGIC INTO SYNCRONIZATION
007744  042777  020000  007064                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
007752  052777  020000  007056                  BIS     #CLK,@TXCSR     ;POKE CLK UP
007760  012777  010004  007060                  MOV     #2$,@DURIV      ;SET UP TRAPCATCHER
007766  013777  016366  007054                  MOV     DUPRT,@DURIS    ;
007774  013737  016370  177776                  MOV     LESS1,PS        ;ALLOW INTERRUPT
010002  000457                                  BR      3$      ;JUMP AROUND SVC ROUTINE
                                        ;THE FOLLOWING IS THE 1ST INTERRUPT SVC ROUTINE
010004  012737  000340  177776          2$:     MOV     #LEVEL7,PS      ;DON'T ALLOW ANY MORE INTERRUPTS
010012  105777  007004                          TSTB    @RXCSR          ;RXDONE = 1 ?
010016  100401                                  BMI     .+4
010020  104000                                  HLT                     ;FALSE INTERRUPT
010022  012716  010306                          MOV     #5$,(SP)        ;SET UP RETURN LOCATION
010026  012777  010110  007012                  MOV     #4$,@DURIV      ;SET UP TRAPCATCHER FOR SECOND
                                                                        ;INTERRUPT
010034  052777  000002  006760                  BIS     #DTR,@RXCSR     ;TRY TO CAUSE SECOND INTERRUPT
010042  017701  006760                          MOV     @RXDBUF,R1      ;JUST READ RXDBUF TO CLR RXDONE
                                                                        ;TO ALLOW SECOND INTERRUPT
010046  013737  016370  177776                  MOV     LESS1,PS        ;ALLOW INTERRUPT
010054  005000                                  CLR     R0
010056  005200                                  INC     R0              ;WAIT FOR INTERRUPT
010060  001376                                  BNE     .-2
010062  042777  000140  006732                  BIC     #RINTEN!DSINTE,@RXCSR   ;CLR INTR ENABLES
010070  104000                                  HLT                     ;2ND INTERRUPT FAILED TO OCCUR
010072  013777  017050  006746          6$:     MOV     DURIS,@DURIV    ;RESTORE TRAPCATCHER
010100  012777  000000  006742                  MOV     #0,@DURIS       ;
010106  000002                                  RTI
                                        ;THE FOLLOWING IS THE 2ND INTERRUPT SVC ROUTINE
010110  012737  000340  177776          4$:     MOV     #LEVEL7,PS      ;DON'T ALLOW ANYMORE INTERRUPTS
010116  005777  006700                          TST     @RXCSR          ;DSC = 1 ?
010122  100401                                  BMI     .+4
010124  104000                                  HLT                     ;FALSE INTERRUPT
010126  042777  000140  006666                  BIC     #RINTEN!DSINTE,@RXCSR   ;CLR BOTH INTR ENABLES
010134  012716  010072                          MOV     #6$,(SP)        ;SET UP RETURN LOCATION
010140  000002                                  RTI
010142  052777  000140  006652          3$:     BIS     #RINTEN!DSINTE,@RXCSR ;SET INTERRUPT ENABLES
010150  012737  000010  001134                  MOV     #8.,SHIFT       ;# OF SHIFTS
010156  012737  000025  001140                  MOV     #25,TEMP1
                                        ;THE FOLLOWING POKES THE MAINT DATA BASED UPON THE
                                        ;INFORMATION CONTAINED IN TEMP1 AND IT IS
```

```
                                        ;SHIFTED IN BY THE CONTENTS OF SHIFT
010164  042777  040000  006644  8$:     BIC     #MTDATA,@TXCSR
010172  000241                          CLC
010174  006037  001140                  ROR     TEMP1   ;FORCE CARRY
010200  103003                          BCC     7$
010202  052777  040000  006626          BIS     #MTDATA,@TXCSR
010210  042777  020000  006620  7$:     BIC     #CLK,@TXCSR
010216  052777  020000  006612          BIS     #CLK,@TXCSR
010224  005337  001134                  DEC     SHIFT
010230  001355                          BNE     8$
                                        ;1ST INTERRUPT SHOULD NOW OCCUR
010232  005000                          CLR     R0
010234  005200                          INC     R0              ;WAIT FOR INTERRUPT
010236  001376                          BNE     .-2             ;
010240  013777  017050  006600          MOV     DURIS,@DURIV    ;RESTORE TRAPCATCHER
010246  012777  000000  006574          MOV     #0,@DURIS       ;
010254  013703  017026                  MOV     RXDBUF,R3       ;FOR ERROR MESSAGE
010260  012700  000025                  MOV     #25,R0          ;EXPECTED
010264  017701  006536                  MOV     @RXDBUF,R1
010270  042777  000140  005524          BIC     #RINTEN!DSINTE,@RXCSR   ;CLR INTERRUPT ENABLES
010276  020001                          CMP     R0,R1
010300  001401                          BEQ     .+4
010302  104002                          HLT     2               ;CHARACTERS SHOULD COMPARE
010304  104000                          HLT                     ;INTERRUPT FAILED TO OCCUR

010306  012737  000340  177776  5$:     MOV     #LEVEL7,PS      ;DON'T ALLOW ANY MORE INTERRUPTS
010314  104400                  1$:     SCOPE

                                        ;;THIS TEST VERIFYS THAT DNA CAUSES AN INTERRUPT
                                        ;;MODE:  SYNC EXTERNAL
                                        ;;INTERRUPT VECTOR:  DUTIV

010316  012737  000023  001122  TST19:  MOV     #19,TSTNO       ;SAVE THIS
010324  012737  010612  001112          MOV     #TST20,NEXT     ;GO TO THIS TEST WHEN THRU

010332  052777  000400  006476          BIS     #MRESET,@TXCSR  ;MASTER RESET
010340  012777  020000  006464          MOV     #SYNEXT,@PARCSR ;SET THE MODE
010346  052777  000400  006462          BIS     #MRESET,@TXCSR  ;MASTER RESET

                                        ;SET MAINTENANCE MODE & SEND
                                                ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
010354  012777  004020  006454          MOV     #MINT!SEND,@TXCSR

                                        ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
010362  012777  026026  006442          MOV     #SYNEXT!EIGHT!NOPAR!26,@PARCSR
010370  112777  000025  006444          MOVB    #25,@TXDBUF     ;LOAD CHARACTER
010376  012737  000010  001134          MOV     #8.,SHIFT
                                        ;POKE CLK TO GET INTO SYNCRONIZATION
010404  052777  020000  006424          BIS     #CLK,@TXCSR     ;POKE CLK UP
010412  042777  020000  006416          BIC     #CLK,@TXCSR     ;POKE CLK DOWN

010420                                  1$:
010420  052777  020000  006410          BIS     #CLK,@TXCSR     ;POKE CLK UP
010426  042777  020000  006402          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
010434  005337  001134                  DEC     SHIFT           ;LAST SHIFT?
010440  001367                          BNE     1$
```

```
010442  012777  010512  006402          MOV     #2$,@DUTIV      ;SET UP TRAPCATCHER
010450  013777  016366  006376          MOV     DUPRT,@DUTIS    ;
010456  013737  016370  177776          MOV     LESS1,PS        ;ALLOW INTERRUPTS
010464  052777  000040  006344          BIS     #DNAINTE,@TXCSR ;ENABLE INTERRUPT
                                        ;NOW POKE CLK TO GET DNA
010472  052777  020000  006336          BIS     #CLK,@TXCSR     ;POKE CLK
010500  005000                          CLR     R0
010502  005200                          INC     R0              ;WAIT FOR INTERRUPT
010504  001376                          BNE     .-2
010506  104000                          HLT                     ;INTERRUPT FAILED TO OCCUR
010510  000423                          BR      3$      ;JUMP AROUND SVC ROUTINE
                                ;THE FOLLOWING IS THE INTERRUPT SERVICE ROUTINE
010512  012737  000340  177776  2$:     MOV     #LEVEL7,PS      ;DON'T ALLOW ANYMORE INTERRUPTS
010520  005777  006312                  TST     @TXCSR          ;DNA?
010524  100401                          BMI     .+4
010526  104000                          HLT                     ;FALSE INTERRUPT
010530  042777  000040  006300          BIC     #DNAINTE,@TXCSR ;CLR INTR ENABLE
010536  012716  010602                  MOV     #4$,(SP)        ;SET UP RETURN LOCATION
010542  013777  017054  006302          MOV     DUTIS,@DUTIV    ;RESTORE TRAPCATCHER
010550  012777  000000  006276          MOV     #0,@DUTIS       ;
010556  000002                          RTI

010560  013777  017054  006264  3$:     MOV     DUTIS,@DUTIV    ;RESTORE TRAPCATCHER
010566  012777  000000  006260          MOV     #0,@DUTIS       ;

010574  042777  000040  006234          BIC     #DNAINTE,@TXCSR ;CLR INTERRUPT ENABLE
010602  012737  000340  177776  4$:     MOV     #LEVEL7,PS      ;RESTORE NO INTERRUPT STATUS
010610  104400                          SCOPE
                                ;;THIS TEST VERIFYS THAT TXDONE CAUSES ONLY ONE INTERRUPT
                                ;;PROVIDING THAT TXCSR IS NOT READ
                                ;;AND TXDBUF IS NOT LOADED (WRITTEN)
                                ;;THIS TEST CHECKS THE ONCE ONLY FLIP/FLOP (V2)
                                ;;OF THE INTERRUPT CONTROL LOGIC
                                ;;INTERRUPT VECTOR: DUTIV
                                ;;NOTE: TXDONE = 1    AFTER A MASTER RESET
                                ;;
010612  012737  000024  001122  TST20:  MOV     #20,TSTNO       ;SAVE THIS
010620  012737  011006  001112          MOV     #TST21,NEXT     ;GO TO THIS TEST WHEN THRU
010626  052777  000400  006202          BIS     #MRESET,@TXCSR  ;MASTER RESET
010634  012777  010676  006210          MOV     #1$,@DUTIV      ;SET UP TRAPCATCHER
010642  013777  016366  006204          MOV     DUPRT,@DUTIS    ;
010650  013737  016370  177776          MOV     LESS1,PS        ;ALLOW INTERRUPTS
010656  052777  000100  006152          BIS     #TXINTE,@TXCSR  ;ENABLE INTR ENABLE
010664  005000                          CLR     R0
010666  005200                          INC     R0
010670  001376                          BNE     .-2
010672  104000                          HLT     ;INTERRUPT FAILED TO OCCUR
010674  000427                          BR      4$
                                ;THE FOLLOWING IS THE INTR SVC ROUTINE
010676  012737  000340  177776  1$:     MOV     #LEVEL7,PS      ;DON'T ALLOW ANYMORE INTR
010704  012716  010746                  MOV     #3$,(SP)        ;SET UP RETURN LOCATION
010710  012777  010720  006134          MOV     #2$,@DUTIV      ;SET UP TRAPCATCHER TO
                                                        ;PROVE THAT THE INTERRUPT DOES NOT OCCUR
                                                        ;TWICE (AFTER RTI 'ING FROM THIS
                                                        ;SVC ROUTINE
010716  000002                          RTI
```

```
                                        ;THE FOLLOWING INTERRUPT SVC ROUTINE WILL CATCH THE SECOND INTR
010720  012737  000340  177776  2$:     MOV     #LEVEL7,PS      ;DON'T ALLOW INTER
010726  012716  010754                  MOV     #4$,(SP)        ;SET UP RETURN LOCATION
010732  105777  006100                  TSTB    @TXCSR  ;TXDONE = 1?
010736  100401                          BMI     .+4
010740  104000                          HLT     ;TXDONE SHOULD BE SET
010742  104000                          HLT     ;THE INTERRUPT WAS TAKEN TWICE.........
                                                ;CHECK OUT THE V2 FLIP/FLOP LOGIC
                                                ;IN THE INTERRUPT CONTROL LOGIC

010744  000002                          RTI
010746  005000                  3$:     CLR     R0      ;ALLOW TIME TO CATCH SECOND
010750  005200                          INC     R0              ;IF IT WERE TO OCCUR
010752  001376                          BNE     .-2     ;
010754  013777  017054  006070  4$:     MOV     DUTIS,@DUTIV    ;RESTORE TRAPCATCHER
010762  012777  000000  006064          MOV     #0,@DUTIS       ;
010770  042777  000100  006040          BIC     #TXINTE,@TXCSR  ;CLR INTERRUPT ENABLE
010776  012737  000340  177776          MOV     #LEVEL7,PS      ;RESTORE NO INTERRUPT STATUS
011004  104400                          SCOPE
                                ;;THIS TEST VERIFYS CTP MODE (IE SYSTST MODE)
                                ;;IT BASICALLY CHECKS THE EXISTANCE OF
                                ;;THE FREE RUNNING OSCILLATOR
                                ;;MODE: SYNINT
                                ;;LENGTH: EIGHT
                                ;;THIS TEST USES BOTH THE RECEIVER & TRANSMITTER LOGIC
                                ;;
011006  012737  000025  001122  TST21:  MOV     #21,TSTNO       ;SAVE THIS
011014  012737  011650  001112          MOV     #TST22,NEXT     ;GO TO THIS TEST WHEN THRU
011022  052777  000400  006006          BIS     #MRESET,@TXCSR  ;MASTER RESET
011030  012777  030000  005774          MOV     #SYNINT,@PARCSR ;SET THE MODE
011036  052777  000400  005772          BIS     #MRESET,@TXCSR  ;MASTER RESET

                                ;SET MAINTENANCE MODE & SEND
                                        ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
011044  012777  014020  005764          MOV     #SYSTST!SEND,@TXCSR

                                ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
011052  012777  036026  005752          MOV     #SYNINT!EIGHT!NOPAR!26,@PARCSR
011060  052777  000420  005734          BIS     #SYNSCH!STPSYN,@RXCSR   ;SET SEARCH SYNC &
                                                ;STRIP SYNC SO THAT RXDONE ASSERTS
                                                ;WHEN CHAR "25" ARRIVES AND NOT BEFORE...
                                                ; ,,,THEREFORE,SET STRIP SYNC
                                        ;...........WAIT FOR SYNSCH TO BE
                                        ;CLOCKED IN BY SYSTST CLK
011066  005037  001150                  CLR     TEMP5
011072  005002                          CLR     R2
011074  005202                          INC     R2      ;WAIT
011076  001376                          BNE     .-2
011100  005237  001150                  INC     TEMP5
011104  022737  000003  001150          CMP     #3,TEMP5
011112  002367                          BGE     .-20    ;GO BACK TO CLR R2 AND WAIT SOME MORE
011114  012777  011334  005724          MOV     #2$,@DURIV      ;SET UP TRAPCATCHER
011122  013777  016366  005720          MOV     DUPRT,@DURIS    ;
011130  012777  011430  005714          MOV     #3$,@DUTIV      ;
011136  013777  016366  005710          MOV     DUPRT,@DUTIS    ;
011144  013737  016370  177776          MOV     LESS1,PS        ;ALLOW INTERRUPTS
011152  013703  017026                  MOV     RXDBUF,R3       ;SET UP FOR ERROR MSG
```

```
011156  012700  000025                    MOV     #25,R0   ;EXPECTED CHAR
011162  012737  000002  001136            MOV     #2,COUNT         ;# OF SYNC CHARS TO GET INTO
                                                  ;SYNCRONIZATION
011170  105737  001172                    TSTB    SYNCNO   ;TEST TO SEE HOW MANY SYNC CHARS NEEDED
011174  100402                            BMI     9$
011176  005337  001136                    DEC     COUNT    ;MAKE IT ONE LESS
011202  052777  000100  005612   9$:      BIS     #RINTEN,@RXCSR  ;SET INTERRUPT ENABLES
011210  052777  000100  005620            BIS     #TXINTE,@TXCSR  ;
011216  000137  011234                    JMP     8$       ;THE FIRST XMIT INTERRUPT SHOULD COME
                                                  ;FROM TXDONE = 1 AFTER A MASTER RESET
011222  112777  000026  005612   1$:      MOVB    #26,@TXDBUF      ;LOAD SYNC CHAR
011230  005037  001150                    CLR     TEMP5
011234  005002                   8$:      CLR     R2       ;WAIT FOR INTERRUPT
011236  005202                            INC     R2
011240  001376                            BNE     .-2      ;
011242  005237  001150                    INC     TEMP5
011246  022737  000003  001150            CMP     #3,TEMP5
011254  002367                            BGE     8$
011256  012737  000340  177776            MOV     #LEVEL7,PS       ;PREVENT INTERRUPTS
011264  042777  000100  005544            BIC     #TXINTE,@TXCSR   ;CLR INTR ENABLES
011272  042777  000100  005522            BIC     #RINTEN,@RXCSR   ;
011300  013777  017050  005540            MOV     DURIS,@DURIV     ;RESTORE TRAPCATCHER
011306  012777  000000  005534            MOV     #0,@DURIS        ;
011314  013777  017054  005530            MOV     DUTIS,@DUTIV     ;
011322  012777  000000  005524            MOV     #0,@DUTIS
011330  104000                            HLT     ;TXDONE INTERRUPT FAILED TO OCCUR
                                                  ;WATCH OUT HERE::: THIS FAILURE MAY
                                                  ;ALSO BE CAUSED BY TRANSMIT DATA NOT
                                                  ;BEING CLOCKED OUT.  I.E.  TXDONE
                                                  ;NOT RE-ASCERTING SO THAT THE 2ND
                                                  ;SYNC CHARACTER CAN BE LOADED

011332  000542                            BR      7$       ;GET OUT OF THE TEST

                                 ;THE FOLLOWING IS THE RECEIVER INTERRUPT SVC ROUTINE
011334  012737  000340  177776   2$:      MOV     #LEVEL7,PS       ;PREVENT INTERRUPTS
011342  017704  005454                    MOV     @RXCSR,R4        ;SAVE
011346  017701  005454                    MOV     @RXDBUF,R1       ;ACTUAL
011352  013777  017050  005466            MOV     DURIS,@DURIV     ;RESTORE TRAPCATCHER
011360  012777  000000  005462            MOV     #0,@DURIS        ;
011366  013777  017054  005456            MOV     DUTIS,@DUTIV     ;
011374  012777  000000  005452            MOV     #0,@DUTIS        ;
011402  012716  011564                    MOV     #4$,(SP)         ;SET UP RETURN LOCATION
011406  042777  000100  005406            BIC     #RINTEN,@RXCSR   ;CLR INTERRUPT ENABLES
011414  042777  000100  005414            BIC     #TXINTE,@TXCSR   ;
011422  013705  001136                    MOV     COUNT,R5         ;SAVE COUNT
011426  000002                            RTI
                                 ;END OF RECEIVER INTERRUPT SVC ROUTINE
                                 ;....THE FOLLOWING IS THE XMITTER INTERRUPT SVC ROUTINE
011430  005337  001136           3$:      DEC     COUNT
011434  100403                            BMI     5$
011436  012716  011222                    MOV     #1$,(SP)         ;SET UP RETURN LOCATION
                                                  ;(LOAD SYNC CHARACTER AGAIN)
011442  000002                            RTI
011444  012716  011452           5$:      MOV     #6$,(SP)         ;SET UP RETURN LOCATION
011450  000002                            RTI
```

```
                                                 ;END OF XMITTER INTERRUPT SVC ROUTINE
011452  112777  000025  005362  6$:     MOVB    #25,@TXDBUF       ;LOAD CHARACTER
011460  042777  000100  005350          BIC     #TXINTE,@TXCSR    ;CLR INTR ENABLE
011466  005037  001150                  CLR     TEMP5
011472  005002                  10$:    CLR     R2          ;WAIT FOR INTERRUPT(RECEIVER)
011474  005202                          INC     R2
011476  001376                          BNE     .-2         ;
011500  005237  001150                  INC     TEMP5
011504  022737  000003  001150          CMP     #3,TEMP5
011512  002367                          BGE     10$
011514  012737  000340  177776          MOV     #LEVEL7,PS        ;PREVENT INTERRUPTS
011522  042777  000100  005272          BIC     #RINTEN,@RXCSR    ;CLR INTR ENABLE
011530  013777  017050  005310          MOV     DURIS,@DURIV      ;RESTORE TRAPCATCHER
011536  012777  000000  005304          MOV     #0,@DURIS         ;
011544  013777  017054  005300          MOV     DUTIS,@DUTIV      ;
011552  012777  000000  005274          MOV     #0,@DUTIS         ;
011560  104000                          HLT     ;RECEIVER INTR FAILED TO OCCUR
011562  000426                          BR      7$          ;GET OUT OF TEST
011564  020001                  4$:     CMP     R0,R1
011566  001401                          BEQ     .+4
011570  104002                          HLT     2           ;CHARACTERS DID NOT MATCH
011572  013703  017022                  MOV     RXCSR,R3          ;SET UP FOR ERROR MSG
011576  012700  000200                  MOV     #200,R0 ;EXPECTED RXDONE
011602  010401                          MOV     R4,R1   ;ACTUAL
011604  042701  177577                  BIC     #177577,R1        ;SAVE ONLY RXDONE
011610  020001                          CMP     R0,R1
011612  001401                          BEQ     .+4
011614  104001                          HLT     1           ;FALSE INTERRUPT
011616  020527  177777                  CMP     R5,#-1  ;WAS COUNT =-1   WHEN RECEIVER
                                                                  ;INTERRUPTED ?

011622  001401                          BEQ     .+4
011624  104000                          HLT     ;IF R5 IS GREATER THAN -1..........
                                                ;THEN EITHER THE # OF SYNC STRAP IS WRONG
                                                ;OR RXDONE IS OCCURING TOO SOON
011626  023727  001136  177777          CMP     COUNT,#-1
011634  001401                          BEQ     .+4
011636  104000                          HLT     ;IF THIS TEST FAILS,BUT THE ABOVE TEST
                                                ;DOESN'T.....IT MAY BE THAT CLEARING
                                                ;TXINTE IN THE RECEIVER SVC ROUTINE
                                                ;IS NOT STOPPING TXDONE INTERRUPTS
011640  012737  000340  177776  7$:     MOV     #LEVEL7,PS        ;INHIBIT INTERRUPTS
011646  104400                          SCOPE
                                        ;;THIS TEST VERIFYS MATCH DETECT & DATA RDY
                                        ;;FLAGS FOR EVERY POSSIBLE MATCH CHARACTER
                                        ;;BY OBSERVING RECACT BIT
                                        ;;IT WILL TAKE TWO SYNC # CHARACTERS TO GET RECACT BIT
                                        ;;#:      DEPENDENT ON MONITOR ............
                                        ;;IF ONE SYNC STRAP IS SELECTED ,IT WILL
                                        ;;ONLY TAKE ONE SYNC CHARACTER BEFORE RECACT TO
                                        ;;ASSERT
                                        ;;MODE: SYNC INTERNAL
                                        ;;LENGTH: FIVE
                                        ;;SYNC CHARACTER FOR MATCH: B/C
                                        ;;THIS TEST USES THE TRANSMITTER AND RECEIVER CHIPS
                                        ;;
011650  012737  000026  001122  TST22:  MOV     #22,TSTNO         ;SAVE THIS
```

```
011656  012737  012176  001112          MOV     #TST23,NEXT     ;GO TO THIS TEST WHEN THRU
011664  012737  012000  001114          MOV     #5$,LOCK        ;SET UP FOR SCOPE LOOP
011672  052777  000400  005136          BIS     #MRESET,@TXCSR  ;MASTER RESET
011700  013703  017026                  MOV     RXDBUF,R3       ;SET UP FOR ERROR MESSAGE
                                ;SET SYNC INTERNAL,FIVE,NO PARITY,0 SYNC REGISTER
011704  012704  030000                  MOV     #SYNINT!FIVE!NOPAR,R4   ;CREATE PARAMETERS
011710  012777  004020  005120  6$:     MOV     #MINT!SEND,@TXCSR       ;SET SEND & MAINT INTER
011716  010477  005110                  MOV     R4,@PARCSR      ;LOAD CSR
011722  052777  000020  005072          BIS     #SYNSCH,@RXCSR  ;SET SYNC SEARCH
                                ;POKE CLK TO GET INTO SYNCRONIZATION
                                ;BOTH THE LOGIC & RECEIVER
011730  052777  020000  005100          BIS     #CLK,@TXCSR     ;POKE CLK UP
011736  042777  020000  005072          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
011744  110477  005072                  MOVB    R4,@TXDBUF      ;LOAD DATA CHARACTER
                                ;POKE CLK TO GET TRANSMITTER & RECEIVER INTO SYNCRONIZATION
011750  052777  020000  005060          BIS     #CLK,@TXCSR     ;POKE CLK UP
011756  042777  020000  005052          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
011764  032777  004000  005030          BIT     #RECACT,@RXCSR  ;RECACT ?
011772  001401                          BEQ     .+4
011774  104000                          HLT     ;RECACT SHOULD NOT BE SET
011776  000404                          BR      4$
012000  010477  005026          5$:     MOV     R4,@PARCSR      ;LOAD PARCSR WITH PARAMETERS
012004  110477  005032                  MOVB    R4,@TXDBUF      ;LOAD SYNC CHAR
012010  012737  000002  001136  4$:     MOV     #2,COUNT        ;# OF SYNC CHARS
012016  005777  005014          2$:     TST     @TXCSR  ;DNA ?
012022  100001                          BPL     .+4     ;BR IF NOT SET
012024  104000                          HLT     ;DNA SHOULD NOT BE SET OR....
                                                ;IT SHOULD BE CLEARED FROM PREVIOUS READ

012026  012737  000005  001134          MOV     #5,SHIFT        ;# OF SHIFTS
012034                          1$:
012034  052777  020000  004774          BIS     #CLK,@TXCSR     ;POKE CLK UP
012042  042777  020000  004766          BIC     #CLK,@TXCSR     ;POKE CLK DOWN
012050  005337  001134                  DEC     SHIFT   ;# OF SHIFTS
012054  001367                          BNE     1$
012056  005337  001136                  DEC     COUNT   ;# OF SYNC CHARS
012062  001403                          BEQ     3$
                                ;TEST SYNCNO TO SEE HOW MANY SYNC CHARACTERS NEEDED
012064  105737  001172                  TSTB    SYNCNO
012070  100752                          BMI     2$      ;TWO SYNC CHARACTERS..
012072  032777  004000  004722  3$:     BIT     #RECACT,@RXCSR  ;RECACT ?
012100  001001                          BNE     .+4
012102  104000                          HLT             ;RECACT FAILED TO SET,POSSIBLE
                                                        ;THAT THE RECEIVER FAILED TO MATCH
                                                        ;THE SYNC CHARACTER

012104  017701  004716                  MOV     @RXDBUF,R1      ;SAVE ACTUAL
012110  010400                          MOV     R4,R0           ;SAVE EXPECTED
012112  042700  177400                  BIC     #177400,R0      ;CLR UPPER BYTE
012116  020001                          CMP     R0,R1   ;DO THEY COMPARE ?
012120  001401                          BEQ     .+4
012122  104002                          HLT     2               ;IF RECACT FAILED ALONG WITH THIS
                                                        ;...IT PROBABLY IS A TRANSMITTER ERROR
                                                        ;HOWEVER,...IF ONLY THIS FAILED IT
                                                        ;PROBABLY IS A RECEIVER ERROR

012124  104401                          SCOP1
                                ;POKE CLK TO SEE DNA...DNA COMES UP ON THE FIRST
                                ;BIT OF THE NEXT CHARACTER IF NO CHARACTER IS LOADED INTO
```

```
                                        ;TXDBUF
012126  052777  020000  004702                  BIS     #CLK,@TXCSR     ;POKE CLK UP
012134  005777  004676                          TST     @TXCSR  ;DNA?
012140  100401                                  BMI     .+4
012142  104000                                  HLT     ;DNA DID NOT ASSERT
                                                ;SET UP CONDITIONS FOR NEXT SYNC CHARACTER MATCH
012144  052777  000400  004664                  BIS     #MRESET,@TXCSR  ;MASTER RESET
012152  032777  000020  004642                  BIT     #SYNSCH,@RXCSR  ;SYNC SEARCH = 0 ?
012160  001401                                  BEQ     .+4
012162  104000                                  HLT     ;SYNC SEARCH SHOULD BE NOT SET
012164  005204                                  INC     R4
012166  122704  000040                          CMPB    #40,R4  ;IS THIS THE LAST CHARACTER ?
012172  001246                                  BNE     6$      ;NO
012174  104400                                  SCOPE
                                                ;;THIS TEST PERFORMS A BINARY COUNT DATA PATTERN ON
                                                ;;BOTH THE TRANSMITTER AND RECEIVER LOGIC
                                                ;;MODE:SYNC EXTERNAL (SYNEXT)
                                                ;;LENGTH:EIGHT PLUS PARITY
                                                ;;PARITY:ODDPAR
                                                ;;MAINT. MODE:MEXT
                                                ;;
012176  012737  000027  001122  TST23:          MOV     #23,TSTNO       ;SAVE THIS
012204  012737  012456  001112                  MOV     #.EOP,NEXT      ;GO TO THIS TEST WHEN THRU
012212  105737  001177                          TSTB    JMRBY   ;JUMP AROUND TEST ?
012216  100116                                  BPL     4$      ;YES ?
012220  052777  000400  004610                  BIS     #MRESET,@TXCSR  ;MASTER RESET
012226  012777  020000  004576                  MOV     #SYNEXT,@PARCSR ;SET THE MODE
012234  052777  000400  004574                  BIS     #MRESET,@TXCSR  ;MASTER RESET

                                        ;SET MAINTENANCE MODE & SEND
                                                ;NOTE:BIT WINDOW&CLK ARE CLEARED (MTDATA=0)
012242  012777  010020  004566                  MOV     #MEXT!SEND,@TXCSR

                                        ;SET MODE,# OF BITS,PARITY SENSE,& LOAD SYNC REG
012250  012777  027026  004554                  MOV     #SYNEXT!EIGHT!ODDPAR!26,@PARCSR
012256  013703  017026                          MOV     RXDBUF,R3       ;SETUP FOR ERROR MSG
012262  005004                                  CLR     R4      ;FOR DATA CHAR CREATION
012264  110477  004552                          MOVB    R4,@TXDBUF      ;LOAD CHARACTER
012270  052777  000020  004524                  BIS     #SYNSCH,@RXCSR  ;SET SEARCH SYNC
                                        ;GET INTO SYNCRONIZATION
012276  052777  020000  004532                  BIS     #CLK,@TXCSR     ;POKE CLK UP
                                                ;WAIT FOR CABLE & DRIVER DELAYS
012304  013702  001132                          MOV     HOLD,R2 ;WAIT THIS AMT
012310  005302                                  DEC     R2      ;WAIT
012312  001376                                  BNE     .-2
                                                ;EXIT...
012314  042777  020000  004514                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
                                                ;WAIT FOR CABLE & DRIVER DELAYS
012322  013702  001132                          MOV     HOLD,R2 ;WAIT THIS AMT
012326  005302                                  DEC     R2      ;WAIT
012330  001376                                  BNE     .-2
                                                ;EXIT...
012332  012737  000011  001134  1$:             MOV     #9.,SHIFT       ;# OF SHIFTS
012340  010400                                  MOV     R4,R0   ;EXPECTED
012342                          2$:
012342  052777  020000  004466                  BIS     #CLK,@TXCSR     ;POKE CLK UP
```

```
                                                ;WAIT FOR CABLE & DRIVER DELAYS
012350  013702  001132                          MOV     HOLD,R2 ;WAIT THIS AMT
012354  005302                                  DEC     R2      ;WAIT
012356  001376                                  BNE     .-2
                                                ;EXIT...
012360  042777  020000  004450                  BIC     #CLK,@TXCSR     ;POKE CLK DOWN
                                                ;WAIT FOR CABLE & DRIVER DELAYS
012366  013702  001132                          MOV     HOLD,R2 ;WAIT THIS AMT
012372  005302                                  DEC     R2      ;WAIT
012374  001376                                  BNE     .-2
                                                ;EXIT...
012376  005337  001134                          DEC     SHIFT   ;# OF SHIFTS
012402  022737  000003  001134                  CMP     #3,SHIFT        ;TIME TO LOAD NEXT CHAR ?
012410  001003                                  BNE     3$      ;NO ?
012412  005204                                  INC     R4      ;GENERATE NEXT CHAR
012414  110477  004422                          MOVB    R4,@TXDBUF      ;LOAD NEXT CHARACTER
012420  005737  001134                  3$:     TST     SHIFT   ;IS IT 0 ?
012424  001346                                  BNE     2$
012426  105777  004370                          TSTB    @RXCSR  ;RXDONE = 1 ?
012432  100401                                  BMI     .+4
012434  104000                                  HLT     ;RXDONE SHOULD BE SET
012436  017701  004364                          MOV     @RXDBUF,R1      ;ACTUAL
012442  020001                                  CMP     R0,R1   ;COMPARE EXP VS ACT
012444  001401                                  BEQ     .+4
012446  104002                                  HLT     2       ;CHARACTERS SHOULD COMPARE
                                                ;CHECK OUT MODEM BYPASS JUMPER
012450  105704                                  TSTB    R4      ;LAST CHARACTER ?
012452  001327                                  BNE     1$      ;NO
012454  104400                          4$:     SCOPE


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

012456  012700  000062                  .EOP:   MOV     #50.,R0
012462  000005                          2$:     RESET
012464  005300                                  DEC     R0
012466  001375                                  BNE     2$
012470  032737  000100  177570                  BIT     #SW06,SWR
012476  001402                                  BEQ     1$
012500  004737  000250                          JSR     PC,EOPHLT
012504  104402                          1$:     TYPE                            ;TYPE NAME OF TEST
012506  015353                                  MEPASS
012510  104410  012742                          CONVRT  ,OUTCRY
012514  104402  015074                          TYPE    ,DEVICE
012520  105737  001176                          TSTB    MULTD   ;ARE YOU RUNNING MULTIPLE DEVICES ?
012524  001511                                  BEQ     CCC     ;NO,JUMP AROUND
012526  005737  001212                          TST     ACTREG  ;ARE ANY DEVICES ACTIVE ?
012532  001007                                  BNE     RUNIT   ;YES
012534  104402  015106                          TYPE    ,MCOW   ;NO
012540  013700  001212                          MOV     ACTREG,R0       ;DISPLAY ACTREG
012544  000000                                  HALT            ;SELECT SOMETHING TO RUN @ ACTREG:
                                                ;SELECT SWITCHES & HIT CONTINUE (PUT SW00 =1)
```

```
012546  000137  001246                  JMP     .START  ;START OVER AGAIN......YOU DESELECTED EVERYTHING
012552  062737  000010  001200  RUNIT:  ADD     #10,BASEADD     ;NEXT BLOCK (ADDRESSES)
012560  062737  000010  001206  ZERO:   ADD     #10,BASEIV      ;NEXT BLOCK (VECTORS)
012566  000241                          CLC
012570  006137  001214                  ROL     ROTADD  ;UP DATE ROTATING POINTER
012574  103410                          BCS     2$              ;IS IT THE LAST DEVICE
                                                                ;TO BE TESTED IN THIS PASS ?
012576  033737  001214  001212          BIT     ROTADD,ACTREG   ;TEST THIS DEVICE FOR ACTIVE STATUS
012604  001762                          BEQ     RUNIT   ;IF NOT ACTIVE, TRY NEXT ADDRESS
012606  004737  012646                  JSR     PC,REPLAY       ;CALCULATE NEW PARAMETERS
012612  000137  013012                  JMP     RESTRT  ;YES IT WAS ACTIVE,TEST THIS DEVICE
012616  012737  000001  001214  2$:     MOV     #1,ROTADD       ;OK!,NOW SET UP  ROTATING
                                                        ;POINTER FOR NEXT MULTIPLE PASS
012624  013737  001202  001200          MOV     KEEPADD,BASEADD ;RESTORE BASE ADDRESS
012632  013737  001210  001206          MOV     KEEPIV,BASEIV   ;RESTORE BASE INTERRUPT VECTORS
012640  004737  012646                  JSR     PC,REPLAY       ;CALC NEW PARAMETERS
012644  000441                          BR      CCC     ;JUMP AROUND REPLAY
012646  013737  001200  016524  REPLAY: MOV     BASEADD,DUBASE  ;SET UP FOR NEW ADDRESSES
012654  004737  016372                  JSR     PC,DUADDR       ;CREATE NEW ADDRESSES
012660  013737  001206  017046          MOV     BASEIV,DURIV    ;CREATE DURIV
012666  062737  000002  001206          ADD     #2,BASEIV
012674  013737  001206  017050          MOV     BASEIV,DURIS    ;CREATE DURIS
012702  062737  000002  001206          ADD     #2,BASEIV
012710  013737  001206  017052          MOV     BASEIV,DUTIV    ;CREATE DUTIV
012716  062737  000002  001206          ADD     #2,BASEIV
012724  013737  001206  017054          MOV     BASEIV,DUTIS    ;CREATE DUTIS
012732  013737  017046  001206          MOV     DURIV,BASEIV    ;RESTORE
012740  000207                          RTS     PC

012742  000001                  OUTCRY: 1
012744     006     002                  .BYTE   6,2
012746  017022                          RXCSR

012750                          CCC:
012750  005037  001130                  CLR     LSTERR                  ;CLEAR LAST ERROR PC
012754  005037  001220                  CLR     ERRFLG                  ;CLEAR ERROR FLAG
012760  005237  001124                  INC     PASCNT                  ;UPDATE PASS COUNT
012764  013737  001124  177570          MOV     PASCNT,LIGHTS           ;DISPLAY PASS COUNT
012772  013701  000042                  MOV     @#42,R1                 ;CHECK FOR ACT-11 OR DDP
012776  001405                          BEQ     RESTRT                  ;IF NOT, CONTINUE TESTING
013000  000005                          RESET
013002  004711                  LOGICAL:        JSR     PC,(R1)
013004  000240                          NOP
013006  000240                          NOP
013010  000240                          NOP
013012  012737  000340  177776  RESTRT: MOV     #340,PS ;PREVENT INTERRUPTS (PRIO: 7)
013020  012737  002224  001110          MOV     #TST1,RETURN
013026  000137  002224                  JMP     TST1

                                        ;SCOPE LOOP AND INTERATION HANDLER

013032                          .SCOPE:
                                ;**** START OF CODE FOR THE X OR TESTER ******
013032  000424                          BR      4$      ;IF RUNNING ON THE X OR TESTER CHANGE
                                                        ;THIS INSTRUCTION TO A "NOP"(NOP=240)
013034  013746  000004                  MOV     @#4,-(SP)       ;SAVE CONTENTS OF ERROR VECTOR
```

```
013040 012737 013060 000004          MOV     #1$,@#4          ;SET FOR TIME OUT
013046 005737 177060                 TST     @#177060         ;TIME OUT ON  X OR ?
013052 012637 000004                 MOV     (SP)+,@#4        ;RESTORE ERROR VECTOR
013056 000404                        BR      2$      ;GO TO NEXT TEST
013060 022626                1$:     CMP     (SP)+,(SP)+      ;CLEAR THE STACK AFTER A TIMEOUT
013062 012637 000004                 MOV     (SP)+,@#4        ;RESTORE ERROR VECTOR
013066 000403                        BR      3$      ;LOOP ON PRESENT TEST
013070 013737 001112 001110  2$:     MOV     NEXT,RETURN      ;SET UP NEXT TEST IN RETURN
013076 013716 001110         3$:     MOV     RETURN,(SP)      ;SET UP STACK FOR RTI
013102 000002                        RTI
013104                       4$:     ;**** END OF CODE FOR THE  X OR TESTER ******
013104 032737 040000 177570          BIT     #SW14,SWR        ;LOOP ON CURRENT TEST ?
013112 001407                TTST:   BEQ     1$
013114 000432                        BR      3$
013116 105777 165756                 TSTB    @TKCSR  ;TEST TTY FLAG
013122 100027                        BPL     3$
013124 017700 165752                 MOV     @TKDBR,R0        ;CLR DONE BIT
013130 000412                        BR      2$      ;IF A TTY KEY IS STRUCK GO TO NEXT TEST
013132 032737 004000 177570  1$:     BIT     #SW11,SWR        ;INHIBIT ITERATIONS ?
013140 001006                        BNE     2$
013142 005237 001120                 INC     LPCNT
013146 023737 001120 001116          CMP     LPCNT,ICOUNT     ;CHECK FOR ITERATION CNT FINISH
013154 101412                        BLOS    3$
013156 105037 001220         2$:     CLRB    ERRFLG
013162 005037 001120                 CLR     LPCNT
013166 012737 000005 001116          MOV     #5,ICOUNT        ;SET UP ITERATION COUNT
013174 013737 001112 001110          MOV     NEXT,RETURN      ;SET UP NEXT TEST IN RETURN
013202 013716 001110         3$:     MOV     RETURN,(SP)      ;SET UP STACK FOR RTI
013206 000002                        RTI
013210 001407                BRW:    1407    ;RESTORE "BEQ 1$" INSTRUCTION
013212 000432                BRX:    432     ;RESTORE "BR 3$" INSTRUCTION

                                     ;CHECK FOR FREEZE ON CURRENT DATA

013214 032737 001000 177570  .SCOP1: BIT     #SW09,SWR
013222 001402                        BEQ     1$
013224 013716 001114                 MOV     LOCK,(SP)
013230 000002                1$:     RTI

                                     ;TELETYPE OUTPUT ROUTINE

013232 017605 000000         .TYPE:  MOV     @(SP),R5
013236 062716 000002                 ADD     #2,(SP)
013242 105715                1$:     TSTB    (R5)    ;LOOK FOR "0"
013244 001406                        BEQ     3$
013246 105777 165632         2$:     TSTB    @TPCSR  ;TEST DONE BIT
013252 100375                        BPL     2$
013254 112577 165626                 MOVB    (R5)+,@TPDBR     ;TYPE CHAR
013260 000770                        BR      1$      ;DO IT AGAIN UNTIL "0" IS SEEN
013262 000002                3$:     RTI

                                     ;ASCII STRING INPUT ROUTINE

013264 017637 000000 013300  .INSTR: MOV     @(SP),.MSG       ;PICK UP MESSAGE
013272 062716 000002                 ADD     #2,(SP)          ;JUMP AROUND MESSAGE FOR RTI
013276 104402                .INST1: TYPE
```

```
013300  000000                   .MSG:   0
013302  012704  016150                   MOV     #INBUF,R4       ;GET STARTING LOC OF INBUF
013306  012703  000007                   MOV     #7,R3           ;MAX # OF CHARS
013312  105777  165562           1$:     TSTB    @TKCSR  ;TTY FLAG
013316  100375                           BPL     1$
013320  117714  165556                   MOVB    @TKDBR,(R4)     ;TAKE CHAR
013324  142714  000200                   BICB    #200,(R4)       ;STRIP
013330  122427  000015                   CMPB    (R4)+,#15       ;CHECK FOR CR
013334  001413                           BEQ     INSTR2
013336  105777  165542           2$:     TSTB    @TPCSR  ;TEST FLAG
013342  100375                           BPL     2$
013344  117777  165532  165534           MOVB    @TKDBR,@TPDBR   ;ECHO CHARACTER
013352  005303                           DEC     R3      ;DID YOU TYPE TOO MANY CHARS ?
013354  001356                           BNE     1$
013356  104402                   .INSTE: TYPE
013360  015257                           MQM     ;?
013362  000745                           BR      .INST1  ;RETRY
013364  000002                   INSTR2: RTI

                                         ;CONVERT ASCII STRING TO OCTAL

013366  011605                   .PARAM: MOV     (SP),R5 ;PUT CONTENTS OF SP INTO R5
013370  012537  013542                   MOV     (R5)+,LOLIM     ;PUT LOW LIMIT INTO LOLIM
013374  012537  013544                   MOV     (R5)+,HILIM     ;PUT HIGH LIMIT INTO HILIM
013400  012537  013546                   MOV     (R5)+,DEVADR    ;PUT STORE LOC INTO DEVADR
013404  112537  013550                   MOVB    (R5)+,LOBITS    ;PUT MASK INTO LOBITS
013410  112537  013551                   MOVB    (R5)+,ADRCNT    ;PUT COUNT INTO ADRCNT
013414  010516                           MOV     R5,(SP) ;RESTORE RETURN ADDR ON STACK FOR RTI
013416  005005                   PARAM1: CLR     R5
013420  012704  016150                   MOV     #INBUF,R4
013424  122714  000015                   CMPB    #15,(R4)        ;CR ?
013430  001420                           BEQ     PARERR  ;YOU TYPED CR TOO SOON !
013432  121427  000060           1$:     CMPB    (R4),#60        ;LOW LIMIT ASCII 0
013436  002415                           BLT     PARERR
013440  121427  000067                   CMPB    (R4),#67        ;HIGH LIMIT ASCII 7
013444  003012                           BGT     PARERR
013446  142714  000060                   BICB    #60,(R4)        ;CONVERT TO OCTAL
013452  152405                           BISB    (R4)+,R5        ;STORE AWAY ITS AN OK CHAR
013454  122714  000015                   CMPB    #15,(R4)        ;CR ?
013460  001406                           BEQ     LIMITS  ;NOW CHECK FOR HIGH &LOW LIMIT CONDS
013462  006305                           ASL     R5      ;ALLOCATE ROOM FOR NEXT CHAR
013464  006305                           ASL     R5
013466  006305                           ASL     R5
013470  000760                           BR      1$
013472  104404                   PARERR: INSTER  ;RETRY
013474  000750                           BR      PARAM1

                                         ;TEST TO SEE IF NUMBER IS WITHIN LIMITS

013476  020537  013544           LIMITS: CMP     R5,HILIM
013502  101373                           BHI     PARERR  ;THE # IS TOO HIGH
013504  020537  013542                   CMP     R5,LOLIM
013510  103770                           BLO     PARERR  ;THE # IS TOO LOW
013512  133705  013550                   BITB    LOBITS,R5       ;TEST BY MASKINGTHE #
013516  001365                           BNE     PARERR
```

```
                                               ;STORE NUMBER AT SPECIFIED ADDRESS

013520  013704  013546                 MOV     DEVADR,R4          ;GET STARTING ADDR OF
013524  010524                 1$:     MOV     R5,(R4)+           ;STORE AT THIS ADDR
013526  062705  000002                 ADD     #2,R5
013532  105337  013551                 DECB    ADRCNT  ;HOW MANY TIMES + 2 ?
013536  001372                         BNE     1$
013540  000002                         RTI
013542  000000                 LOLIM:  0
013544  000000                 HILIM:  0
013546  000000                 DEVADR: 0
013550  000000                 LOBITS: 0
        013551                 ADRCNT=LOBITS+1

                                               ;SAVE PC OF TEST THAT FAILED AND R0-R5

013552  016637  000004  001170         .SAVO5: MOV     4(SP),SAVPC

                                               ;SAVE R0-R5

013560  010537  001164                 SVO5:   MOV     R5,SAVR5
013564  010437  001162                         MOV     R4,SAVR4
013570  010337  001160                         MOV     R3,SAVR3
013574  010237  001156                         MOV     R2,SAVR2
013600  010137  001154                         MOV     R1,SAVR1
013604  010037  001152                         MOV     R0,SAVR0
013610  000002                                 RTI

                                               ;RESTORE R0-R5

013612  013700  001152                 .RESO5: MOV     SAVR0,R0
013616  013701  001154                         MOV     SAVR1,R1
013622  013702  001156                         MOV     SAVR2,R2
013626  013703  001160                         MOV     SAVR3,R3
013632  013704  001162                         MOV     SAVR4,R4
013636  013705  001164                         MOV     SAVR5,R5
013642  000002                                 RTI

                                       ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER

013644  104402                 .CONVR: TYPE
013646  015263                         MCRLF   ;CR  LF
013650  017601  000000                 MOV     @(SP),R1           ;PICK UP DATA POINTER
013654  062716  000002                 ADD     #2,(SP)  ;SET UP SP FOR RTI
013660  012137  014014                 MOV     (R1)+,WRDCNT       ;PICK UP # OF WORDS FROM TABLE
013664  112137  014016         1$:     MOVB    (R1)+,CHRCNT       ;PICK UP # OF CHARS FROM TABLE
013670  112137  014017                 MOVB    (R1)+,SPACNT       ;PICK UP # OF SPACES FROM TABLE
013674  013137  014020                 MOV     @(R1)+,BINWRD      ;PICK UP ADDRESS OF  MSG
                                                                  ;FROM TABLE
013700  013704  014020         2$:     MOV     BINWRD,R4          ;SAVE
013704  113705  014016                 MOVB    CHRCNT,R5          ;SAVE
013710  012700  016212                 MOV     #TEMP,R0           ;STARTING ADDRESS OF TEMP BLOCK
013714  010403                 3$:     MOV     R4,R3              ;SAVE
013716  042703  177770                 BIC     #177770,R3         ;CLR OUT UPPER BITS .. SAVE CHAR
013722  062703  000260                 ADD     #260,R3 ;CONVERT TO ASCII
013726  110320                         MOVB    R3,(R0)+           ;STORE AWAY
```

```
013730  006204                                    ASR     R4      ;SHIFT FOR NEXT #
013732  006204                                    ASR     R4      ;DITTO
013734  006204                                    ASR     R4      ;DITTO
013736  005305                                    DEC     R5      ;DEC CHAR COUNT
013740  001365                                    BNE     3$      ;DO IT AGAIN ?
013742  012703  016254                            MOV     #MDATA,R3       ;STARTING ADDRESS OF MDATA BLOCK
013746  114023                            4$:     MOVB    -(R0),(R3)+     ;REVERSE THE ORDER OF NUMBERS
013750  105337  014016                            DECB    CHRCNT  ;DEC CHAR COUNT
013754  001374                                    BNE     4$      ;DO IT AGAIN ?
013756  105737  014017                            TSTB    SPACNT  ;HOW MANY SPACES ?
013762  001405                                    BEQ     6$      ;TYPE # IF BR =0
013764  112723  000240                    5$:     MOVB    #240,(R3)+      ;"SPACE" IN ASCII
013770  105337  014017                            DECB    SPACNT  ;DEC # OF SPACE COUNT
013774  001373                                    BNE     5$      ;DO IT AGAIN ?
013776  105013                            6$:     CLRB    (R3)    ;INSERT "0" FOR TTY OUTPUT ROUTINE
014000  104402                                    TYPE
014002  016254                                    MDATA   ;THIS MESSAGE
014004  005337  014014                            DEC     WRDCNT  ;HOW MANY #'S ?
014010  001325                                    BNE     1$      ;DO THIS ROUTINE AGAIN IF NOT EQUAL TO 0
014012  000002                                    RTI     ;RETURN TO PROGRAM
014014  000000                            WRDCNT: 0
014016  000000                            CHRCNT: 0
        014017                            SPACNT=CHRCNT+1
014020  000000                            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

014022  017605  000000                    .SETFLG:MOV     @(SP),R5
014026  122737  000116  016150                    CMPB    #'N,INBUF       ;IS IT "N" ?
014034  001002                                    BNE     1$
014036  105015                                    CLRB    (R5)    ;000
014040  000406                                    BR      2$
014042  122737  000131  016150            1$:     CMPB    #'Y,INBUF       ;IS IT "Y" ?
014050  001005                                    BNE     3$
014052  112715  177777                            MOVB    #-1,(R5)        ;377
014056  062716  000002                    2$:     ADD     #2,(SP)
014062  000002                                    RTI
014064  104404                            3$:     INSTER  ;RETRY
014066  000755                                    BR      .SETFLG
                                                          ;TRAP DISPATCH SERVICE
                                                          ;ARGUMENT OF TRAP IS EXTRACTED
                                                          ;AND USED AS OFFSET TO OBTAIN POINTER
                                                          ;TO SELECTED SUBROUTINE

014070  011646                                    .TRPSR: MOV     (SP),-(SP)              ;GET PC OF RETURN
014072  162716  000002                                    SUB     #2,(SP)                 ;=PC OF TRAP
014076  017616  000000                                    MOV     @(SP),(SP)              ;GET TRP
014102  006316                                    TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
014104  042716  177001                                    BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
014110  062716  001222                                    ADD     #.TRPTAB,(SP)           ;POINTER TO SUBROUTINE ADDRESS
014114  017616  000000                                    MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
014120  000136                                            JMP     @(SP)+                  ;GO TO SUBROUTINE
```

```
                                                 ;ERROR HANDLER

014122  032737  020000  177570  .HLT:    BIT     #SW13,SWR        ;INHIBIT ERROR TYPE OUT ?
014130  001074                           BNE     HALTS
014132  021637  001130                   CMP     (SP),LSTERR
014136  001404                           BEQ     1$
014140  011637  001130                   MOV     (SP),LSTERR
014144  105037  001220                   CLRB    ERRFLG
014150  104406                  1$:      SAVOS
014152  011605                           MOV     (SP),R5
014154  162705  000002                   SUB     #2,R5
014160  012702  017400                   MOV     #ERRST,R2
014164  010522                           MOV     R5,(R2)+
014166  011504                           MOV     (R5),R4
014170  006304                           ASL     R4
014172  061504                           ADD     (R5),R4
014174  006304                           ASL     R4
014176  042704  177001                   BIC     #177001,R4
014202  062704  016772                   ADD     #.ERRTAB,R4
014206  012437  014252                   MOV     (R4)+,ERRMSG
014212  012437  014264                   MOV     (R4)+,DATAHD
014216  011437  014316                   MOV     (R4),DATABP
014222  105737  001220                   TSTB    ERRFLG
014226  001403                           BEQ     TYPMSG
014230  005737  014316                   TST     DATABP
014234  001014                           BNE     TYPDAT
014236  104410                  TYPMSG:  CONVRT
014240  014406                           ERTABO
014242  112737  177777  001220           MOVB    #-1,ERRFLG
014250  104402                           TYPE
014252  000000                  ERRMSG:  0
014254  005737  014264                   TST     DATAHD
014260  001402                           BEQ     TYPDAT
014262  104402                           TYPE
014264  000000                  DATAHD:  0
014266  005737  014316          TYPDAT:  TST     DATABP
014272  001412                           BEQ     RESREG
014274  013722  001160                   MOV     SAVR3,(R2)+
014300  013722  001152                   MOV     SAVR0,(R2)+
014304  013722  001154                   MOV     SAVR1,(R2)+
014310  012722  000000                   MOV     #0,(R2)+
014314  104410                           CONVRT
014316  000000                  DATABP:  0
014320  104407                  RESREG:  RESOS
014322  005737  177570          HALTS:   TST     SWR
014326  100005                           BPL     EXITER
014330  010046                           PUSHR0
014332  016600  000002                   MOV     2(SP),R0
014336  000000                           HALT
014340  012600                           POPR0
014342  005237  001126          EXITER:  INC     ERRCNT
014346  032737  000400  177570           BIT     #SW08,SWR        ;LOOP ON ERROR ?
014354  001007                           BNE     1$
014356  032737  002000  177570           BIT     #SW10,SWR        ;ESCAPE TO NEXT ON ERROR ?
014364  001407                           BEQ     2$
014366  013737  001112  001110           MOV     NEXT,RETURN      ;SET UP FOR NEXT TEST
```

```
014374  012706  001100          1$:     MOV     #STACK,SP       ;REINITIALIZE SP
014400  000177  164504                  JMP     @RETURN
014404  000002                  2$:     RTI
014406  000001                  ERTAB0: 1
014410     006     002                  .BYTE   6,2
014412  001170                          SAVPC
                                        ;ENTER HERE ON POWER FAILURE


014414  010046                  .PFAIL: MOV     R0,-(SP)                ;SAVE R0-R5 ON PROCESSOR STACK
014416  010146                          MOV     R1,-(SP)
014420  010246                          MOV     R2,-(SP)
014422  010346                          MOV     R3,-(SP)
014424  010446                          MOV     R4,-(SP)
014426  010546                          MOV     R5,-(SP)
014430  013746  000024                  MOV     24,-(SP)
014434  010637  001166                  MOV     SP,SAVSP                ;SAVE STACK POINTER
014440  012737  014452  000024          MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
014446  000000                          HALT                            ;HALT ON POWER DOWN NORMAL
014450  000777                          BR      .

                                        ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED

014452  013706  001166          RESTAR: MOV     SAVSP,SP                ;RESTORE STACK POINTER
014456  012605                          MOV     (SP)+,R5                ;RESTORE R0-R5
014460  012604                          MOV     (SP)+,R4
014462  012603                          MOV     (SP)+,R3
014464  012602                          MOV     (SP)+,R2
014466  012601                          MOV     (SP)+,R1
014470  012600                          MOV     (SP)+,R0
014472  012737  014414  000024          MOV     #.PFAIL,24              ;SET UP FOR POWER FAILURE
014500  012737  000340  177776          MOV     #340,PS
014506  012706  001100                  MOV     #STACK,SP
014512  005037  016212                  CLR     TEMP
014516  005237  016212                  INC     TEMP
014522  001375                          BNE     .-4
014524  104410                          CONVRT
014526  014550                          PFTAB
014530  104402                          TYPE
014532  015266                          MPFAIL
014534  005037  001220                  CLR     ERRFLG
014540  005037  001130                  CLR     LSTERR
014544  000177  164340                  JMP     @RETURN
014550  000001                  PFTAB:  1
014552     006     002                  .BYTE   6,2
014554  000207                          RETURN
014556  005015  042012  030525  MTITLE: .ASCIZ  <15><12><12>/DU11 DDDUA-A TAPE COMBINED EXERCIZER /<15><12>
014564  020061  042104  052504
014572  026501  020101  040524
014600  042520  041440  046517
014606  044502  042516  020104
014614  054105  051105  044503
014622  042532  020122  005015
014630     000
014631     015  053012  041505  MVECTO: .ASCIZ  <15><12>/VECTOR ADDRESS-/
014636  047524  020122  042101
```

```
014644  051104  051505  026523
014652     000
014653     015  030412  052123  MREGAD: .ASCIZ  <15><12>/1ST DEVICE: RECEIVER CONTROL REGISTER ADDRESS-/
014660  042040  053105  041511
014666  035105  051040  041505
014674  044505  042526  020122
014702  047503  052116  047522
014710  020114  042522  044507
014716  052123  051105  040440
014724  042104  042522  051523
014732  000055
014734  005015  051101  020105  MMULT:   .ASCIZ  <15><12>/ARE YOU RUNNING MULTIPLE DEVICES ? (Y OR N)-/
014742  047531  020125  052522
014750  047116  047111  020107
014756  052515  052114  050111
014764  042514  042040  053105
014772  041511  051505  037440
015000  024040  020131  051117
015006  047040  026451     000
015013     015  046012  051501  MLASTD: .ASCIZ  <15><12>/LAST DEVICE:RECEIVER CONTROL REGISTER ADDRESS-/
015020  020124  042504  044526
015026  042503  051072  041505
015034  044505  042526  020122
015042  047503  052116  047522
015050  020114  042522  044507
015056  052123  051105  040440
015064  042104  042522  051523
015072  000055
015074  042075  053105  041511  DEVICE: .ASCIZ  /=DEVICE  /
015102  020105  000040
015106  005015  047510  020127  MCOW:    .ASCIZ  <15><12>/HOW NOW BROWN COW? ...SELECT SOMETHING TO RUN @ACTREG/
015114  047516  020127  051102
015122  053517  020116  047503
015130  037527  027040  027056
015136  042523  042514  052103
015144  051440  046517  052105
015152  044510  043516  052040
015160  020117  052522  020116
015166  040500  052103  042522
015174  000107
015176  005015  052517  020124  MRANGE: .ASCIZ  <15><12>/OUT OF RANGE:RETYPE LAST DEVICE RXCSR ADDRESS-/
015204  043117  051040  047101
015212  042507  051072  052105
015220  050131  020105  040514
015226  052123  042040  053105
015234  041511  020105  054122
015242  051503  020122  042101
015250  051104  051505  026523
015256     000
015257     040  037440     000  MQM:     .ASCIZ  /  ?/
015263     015  000012          MCRLF:   .ASCIZ  <15><12>
015266  020040  047520  042527  MPFAIL: .ASCIZ  /  POWER FAILURE, PROGRAM RESTART AT TEST IN PROGRESS/
015274  020122  040506  046111
015302  051125  026105  050040
015310  047522  051107  046501
015316  051040  051505  040524
```

```
015324  052122  040440  020124
015332  042524  052123  044440
015340  020116  051120  043517
015346  042522  051523     000
015353     015  042412  042116   MEPASS: .ASCIZ  <15><12>/END OF PASS TAPE DDDUA-A/
015360  047440  020106  040520
015366  051523  052040  050101
015374  020105  042104  052504
015402  026501  000101
015406  005015  000122           MR:     .ASCIZ  <15><12>/R/
015412  005015  042524  052123   MTSTPC: .ASCIZ  <15><12>/TEST PC-/
015420  050040  026503     000
015425     015  046012  041517   MLOCK:  .ASCIZ  <15><12>/LOCK ON SELECTED TEST? (Y OR N)-/
015432  020113  047117  051440
015440  046105  041505  042524
015446  020104  042524  052123
015454  020077  054450  047440
015462  020122  024516  000055
015470  005015  052504  050040   MLEVEL: .ASCIZ  <15><12>/DU PRIORITY LEVEL-/
015476  044522  051117  052111
015504  020131  042514  042526
015512  026514     000
015515     015  021412  047440   MSYNC:  .ASCIZ  <15><12>/# OF SYNC CHARS SELECTED ( 1 OR 2)-/
015522  020106  054523  041516
015530  041440  040510  051522
015536  051440  046105  041505
015544  042524  020104  020050
015552  020061  051117  031040
015560  026451     000
015563     015  044412  020123   MWIRE6: .ASCIZ  <15><12>/IS SEC XMIT JUMPER #6 IN? (Y OR N)-/
015570  042523  020103  046530
015576  052111  045040  046525
015604  042520  020122  033043
015612  044440  037516  024040
015620  020131  051117  047040
015626  026451     000
015631     015  044412  020123   MWIRE5: .ASCIZ  <15><12>/IS SEC REC JUMPER #5 IN? (Y OR N)-/
015636  042523  020103  042522
015644  020103  052512  050115
015652  051105  021440  020065
015660  047111  020077  054450
015666  047440  020122  024516
015674  000055
015676  005015  051511  047440   MWIRE4: .ASCIZ  <15><12>/IS OPT CLR ENABLE JUMPER #4 IN? (Y OR N)-/
015704  052120  041440  051114
015712  042440  040516  046102
015720  020105  052512  050115
015726  051105  021440  020064
015734  047111  020077  054450
015742  047440  020122  024516
015750  000055
015752  005015  051101  020105   MEXTJ:  .ASCII  <15><12>/ARE YOU RUNNING IN MAINT MODE EXTERNAL?/
015760  047531  020125  052522
015766  047116  047111  020107
015774  047111  046440  044501
016002  052116  046440  042117
```

```
016010  020105  054105  042524
016016  047122  046101     077                 .ASCII  <15><12><1>/AND ..... DO YOU HAVE THE EXTERNAL MODEM BYPASS/
016023     015  000412  047101
016030  020104  027056  027056
016036  020056  047504  054440
016044  052517  044040  053101
016052  020105  044124  020105
016060  054105  042524  047122
016066  046101  046440  042117
016074  046505  041040  050131
016102  051501     123
016105     015  000412  052512                 .ASCIZ  <15><12><1>/JUMPER CONNECTOR ON ?(Y OR N)-/
016112  050115  051105  041440
016120  047117  042515  052103
016126  051117  047440  020116
016134  024077  020131  051117
016142  047040  026451     000
        016150                         .EVEN

                                               ;BUFFERS FOR INPUT-OUTPUT

016150  000000                         INBUF:  0
        016212                         .=.+40
016212  000000                         TEMP:   0
        016254                         .=.+40
016254  000000                         MDATA:  0
        016315                         .=.+40
                                       ;*******************************************************
                                               ;UTILITIES
                                       ;*******************************************************

                                       ;THIS UTILITY CALCULATES PRIORITY LEVEL
016316  006337  016366                 DULEV:  ASL     DUPRT   ;SHIFT LEFT
016322  006337  016366                         ASL     DUPRT   ;
016326  006337  016366                         ASL     DUPRT   ;
016332  006337  016366                         ASL     DUPRT   ;
016336  006337  016366                         ASL     DUPRT   ;
016342  013737  016366  016370                 MOV     DUPRT,LESS1     ;MOVE THIS TO LESS1
016350  162737  000001  016370                 SUB     #1,LESS1        ;CREATE LESS1
016356  042737  000037  016370                 BIC     #37,LESS1       ;CLEAR TNZVC
016364  000207                                 RTS     PC
016366  000240                         DUPRT:  LEVEL5
016370  000200                         LESS1:  LEVEL4  ;LEVEL TO ALLOW INTERRUPTS

                                               ;NEW DU ADDRESSES
016372  013737  016524  017022         DUADDR: MOV     DUBASE,RXCSR    ;XXX0
016400  005237  016524                         INC     DUBASE
016404  013737  016524  017024                 MOV     DUBASE,HRXCSR   ;XXX1
016412  005237  016524                         INC     DUBASE
016416  013737  016524  017026                 MOV     DUBASE,RXDBUF   ;XXX2
016424  013737  016524  017032                 MOV     DUBASE,PARCSR   ;XXX2
016432  005237  016524                         INC     DUBASE
016436  013737  016524  017030                 MOV     DUBASE,HRXDBUF  ;XXX3
016444  013737  016524  017034                 MOV     DUBASE,HPARCSR  ;XXX3
016452  005237  016524                         INC     DUBASE
016456  013737  016524  017036                 MOV     DUBASE,TXCSR    ;XXX4
```

```
        016464  005237  016524                          INC     DUBASE
        016470  013737  016524  017040                  MOV     DUBASE,HTXCSR   ;XXX5
        016476  005237  016524                          INC     DUBASE
        016502  013737  016524  017042                  MOV     DUBASE,TXDBUF   ;XXX6
        016510  005237  016524                          INC     DUBASE
        016514  013737  016524  017044                  MOV     DUBASE,HTXDBUF  ;XXX7
        016522  000207                                  RTS     PC
        016524  000000                          DUBASE: 0

                                                        ;THIS UTILITY POKES THE MAINT DATA BASED UPON THE
                                                        ;INFORMATION CONTAINED IN TEMP1 AND IT IS
                                                        ;SHIFTED IN BY THE CONTENTS OF SHIFT
        016526  042777  040000  000302  RPOKE:  BIC     #MTDATA,@TXCSR
        016534  005037  001142                          CLR     TEMP2
        016540  006037  001140                          ROR     TEMP1   ;FORCE CARRY
        016544  006037  001142                          ROR     TEMP2   ;PICK UP CARRY IN BIT 15
        016550  006237  001142                          ASR     TEMP2   ;SHIFT INTO BIT 14
        016554  042737  100000  001142                  BIC     #BIT15,TEMP2    ;CLR BIT 15
        016562  053777  001142  000246                  BIS     TEMP2,@TXCSR    ;POKE MAINT DATA
        016570  042777  020000  000240                  BIC     #CLK,@TXCSR     ;POKE CLK
        016576  052777  020000  000232                  BIS     #CLK,@TXCSR     ;
        016604  005337  001134                          DEC     SHIFT
        016610  001346                                  BNE     RPOKE
        016612  000207                                  RTS     PC
                                                        ;THIS ROUTINE CALCULATES ODD PARITY FOR AN 8 BIT CHAR

        016614  013737  001140  001142  ODD8:   MOV     TEMP1,TEMP2     ;SAVE TEMP1
        016622  005037  001144                          CLR     TEMP3
        016626  012727  000010                          MOV     #8.,(PC)+
        016632  000000                          4$:     0
        016634  006037  001142                  1$:     ROR     TEMP2
        016640  005537  001144                          ADC     TEMP3
        016644  005337  016632                          DEC     4$
        016650  001371                                  BNE     1$
        016652  006037  001144                          ROR     TEMP3
        016656  103404                                  BCS     2$
        016660  052737  000400  001140                  BIS     #BIT8,TEMP1     ;SET ODD PARITY
        016666  000403                                  BR      3$
        016670  042737  000400  001140  2$:     BIC     #BIT8,TEMP1     ;CLR EVEN PARITY
                                                        ;TEMP1 NOW HAS ODD PARITY CHARACTER

        016676  000207                          3$:     RTS     PC

                                                ;THIS ROUTINE CALCULATES EVEN PARITY FOR AN 8 BIT CHARACTER

        016700  013737  001140  001142  EVEN8:  MOV     TEMP1,TEMP2     ;SAVE TEMP1
        016706  005037  001144                          CLR     TEMP3
        016712  012727  000010                          MOV     #8.,(PC)+
        016716  000000                          4$:     0
        016720  006037  001142                  1$:     ROR     TEMP2
        016724  005537  001144                          ADC     TEMP3
        016730  005337  016716                          DEC     4$
        016734  001371                                  BNE     1$
        016736  006037  001144                          ROR     TEMP3
        016742  103004                                  BCC     2$
        016744  052737  000400  001140                  BIS     #BIT8,TEMP1     ;SET EVEN PARITY
        016752  000403                                  BR      3$
        016754  042737  000400  001140  2$:     BIC     #BIT8,TEMP1     ;CLR ODD PARITY
                                                        ;TEMP1 NOW HAS EVEN PARITY CHARACTER
```

```
016762  000207                          3$:     RTS     PC

016764  062716  000002          TRPREG: ADD     #2,(SP) ;ALLOW IT TO "CRUNCH" INTO HLT BAC
                                                        ;IN MAIN PART OF THE PROGRAM

016770  000002                                  RTI
                                                ;ERROR HLT TABLE

016772  017056                  .ERRTAB:                EM0     ;HLT 0  BIT ERROR (GENERAL)
016774  000000                                          0
016776  000000                                          0
017000  017072                                          EM1     ;HLT 1  REGISTER ERROR
017002  017243                                          DH1
017004  017264                                          DT1
017006  017134                                          EM2     ;HLT 2  RECEIVER ERROR
017010  017243                                          DH1
017012  017264                                          DT1
017014  017176                                          EM3     ;HLT 3  TRANSMITTER ERROR
017016  017243                                          DH1
017020  017264                                          DT1
                                                ;DEFAULT DU ADDRESSES

017022  160040                          RXCSR:  160040
017024  160041                          HRXCSR: 160041
017026  160042                          RXDBUF: 160042
017030  160043                          HRXDBUF:        160043
017032  160042                          PARCSR: 160042
017034  160043                          HPARCSR:        160043
017036  160044                          TXCSR:  160044
017040  160045                          HTXCSR: 160045
017042  160046                          TXDBUF: 160046
017044  160047                          HTXDBUF:        160047
                                                ;DEFAULT DU VECTORS

017046  000330                          DURIV:  330     ;REC INTR VECTOR
017050  000332                          DURIS:  332     ;REC INTR STATUS
017052  000334                          DUTIV:  334     ;XMIT INTR VECTOR
017054  000336                          DUTIS:  336     ;XMIT INTR STATUS
                                                ;ERROR MESSAGES

017056  036440  042440  051122  EM0:    .ASCIZ  / = ERROR PC/
017064  051117  050040  000103
017072  036440  051040  043505  EM1:    .ASCIZ  / = REGISTER ERROR PC/<15><12><1>/REGISTER  /
017100  051511  042524  020122
017106  051105  047522  020122
017114  041520  005015  051001
017122  043505  051511  042524
017130  020122  000040
017134  036440  051040  041505  EM2:    .ASCIZ  / = RECEIVER ERROR PC/<15><12><1>/REGISTER  /
017142  044505  042526  020122
017150  051105  047522  020122
017156  041520  005015  051001
017164  043505  051511  042524
017172  020122  000040
017176  036440  052040  040522  EM3:    .ASCIZ  / = TRANSMITTER ERROR PC/<15><12><1>/REGISTER  /
017204  051516  044515  052124
017212  051105  042440  051122
017220  051117  050040  006503
017226  000412  042522  044507
017234  052123  051105  020040
017242     000
```

```
                                                  ;DATA HEADERS FOR ERROR MESSAGES
017243     105  050130  041505   DH1:     .ASCIZ  /EXPECTED  ACTUAL/
017250  042524  020104  040440
017256  052103  040525  000114
                                 .EVEN
                                          ;DATA TABLES FOR ERROR MESSAGES
017264  000003                   DT1:     3
017266     006     004                    .BYTE   6,4
017270  001160                            SAVR3   ;REGISTER
017272     006     004                    .BYTE   6,4
017274  001152                            SAVR0   ;EXPECTED DATA
017276     006     002                    .BYTE   6,2
017300  001154                            SAVR1   ;ACTUAL DATA
        017400                   .=017400
017400  000020                   ERRST:   .BLKW 20
        000001                   .END
```

```
AAA       001722     1278#
ACTREG    001212     1114#   1226*   1240*   1241*   1248*   2849    2852    2862
ADRCNT= 013551       2993*   3028*   3035#
BASEAD    001200     1109#   1208*   1245*   1246    1252*   1254*   2856*   2868*   2872
BASEIV    001206     1112#   1218*   2857*   2869*   2874    2875*   2876    2877*   2878    2879*   2880    2881*
BBB       001540     1225    1229#
BINWRD    014020     3070*   3072    3102#
BITW    = 002000      765#   1788    1846    1899    1952    2090
BIT0    = 000001      713#    741     772
BIT1    = 000002      712#    740
BIT10   = 002000      703#    731     765    1543
BIT11   = 004000      702#    730
BIT12   = 010000      701#    729     746
BIT13   = 020000      700#    728     745     764
BIT14   = 040000      699#    727     744     763
BIT15   = 100000      698#    726     743     762    3432
BIT2    = 000004      711#    739    1321
BIT3    = 000010      710#    738     771
BIT4    = 000020      709#    737     770
BIT5    = 000040      708#    736     769
BIT6    = 000100      707#    735     768
BIT7    = 000200      706#    734     767
BIT8    = 000400      705#    733     749     766    3450    3452    3467    3469
BIT9    = 001000      704#    732     748
BREAK   = 000001      772#   1364    1419    1474    1524    1568    1607    1643    1689    1997    2209    2358
BRW       013210     1318    1332    2940#
BRX       013212     1319    1333    2941#
CARDET= 010000        729#
CCC       012750     2848    2871    2888#
CHPCNT    014016     3068*   3073    3086*   3100#   3101
CLK     = 020000      764#   1364    1370    1371    1373    1374    1419    1425    1426    1428    1429    1474    1480
                     1481    1524    1530    1531    1533    1534    1568    1574    1575    1607    1613    1614    1616
                     1617    1643    1649    1650    1652    1653    1689    1697    1698    1700    1701    1783    1784
                     1790    1791    1801    1841    1842    1848    1849    1859    1894    1895    1901    1902    1912
                     1947    1948    1954    1955    1965    1997    2003    2004    2006    2007    2020    2021    2023
                     2024    2059    2060    2065    2066    2072    2073    2087    2088    2138    2139    2142    2143
                     2167    2168    2209    2216    2217    2219    2220    2358    2364    2365    2413    2414    2455
                     2456    2459    2460    2468    2698    2699    2702    2703    2717    2718    2744    2783    2789
                     2798    2804    3434    3435
CONVRT= 104410       1157#   2845    3161    3176    3227
COUNT     001136     1085#   1619*   1626*   1655*   1662*   1694*   1709*   1720*   1734*   1736    1738    2025*   2032*
                     2137*   2149*   2158*   2171*   2221*   2226*   2270*   2274*   2278*   2576*   2590*   2620    2624*
                     2665    2710*   2721*
CTS     = 020000      728#
DATABP    014316     3156*   3159    3170    3177#
DATAHD    014264     3155*   3166    3169#
DEVADR    013546     2991*   3025    3033#
DEVICE    015074     2846    3275#
DH1       017243     3481    3484    3487    3528#
DNA     = 100000      762#
DNAINT= 000040        769#   2466    2479    2488
DSC     = 100000      726#
DSINTE= 000040        736#   2325    2331    2339    2386    2398    2402    2426
DSR     = 001000      732#
DTR     = 000002      740#   2332    2379
DT1       017264     3482    3485    3488    3533#
```

```
DUADDR   016372    1207   2873   3404#
DUBASE   016524    1203   1206   2872*  3404   3405*  3406   3407*  3408   3409   3410*  3411   3412   3413*
                   3414   3415*  3416   3417*  3418   3419*  3420   3422#
DULEV    016316    1273   3391#
DUPRT    016366    1270   2317   2367   2464   2503   2570   2572   3391*  3392*  3393*  3394*  3395*  3396
                   3400#
DURIS    017050    2317*  2327   2328*  2336   2337*  2367*  2389   2390*  2421   2422*  2570*  2596   2597*
                   2613   2614*  2643   2644*  2876*  3502#
DURIV    017046    1214   1217   1218   2316*  2327*  2336*  2366*  2377*  2389*  2421*  2569*  2596*  2613*
                   2643*  2874*  2881   3501#
DUTIS    017054    2464*  2481   2482*  2485   2486*  2503*  2532   2533*  2572*  2598   2599*  2615   2616*
                   2645   2646*  2880*  3504#
DUTIV    017052    2463*  2481*  2485*  2502*  2514*  2532*  2571*  2598*  2615*  2645*  2878*  3503#
EIGHT  = 006000     757#  1367   1422   1477   1527   1571   1646   1692   1777   1888   1941   2000   2048
                   2133   2212   2361   2451   2555   2777
EM0      017056    3477   3506#
EM1      017072    3480   3508#
EM2      017134    3483   3514#
EM3      017176    3486   3520#
EOPHLT   000250    1050#  2842
ERRCNT   001126    1078#  1177*  3185*
ERRFLG   001220    1124#  1176*  2890*  2934*  3142*  3157   3163*  3231*
ERRMSG   014252    3154*  3165#
ERRST    017400    3146   3541#
ERTAB0   014406    3162   3194#
EVENB    016700    1540   1581   3457#
EVEPAR= 001400      760#  1527   1571   1692   1835   1888   1941
EVPAR  = 000400     749#
EXITER   014342    3180   3185#
FIVE   = 000000     754#  1610   2692
FRMERR= 020000      745#  1746   1749   1752
HALTS    014322    3138   3179#
HOXEN  = 000010     771#
HILIM    013544    2990*  3016   3032#
HOLD     001132    1083#  2785   2791   2800   2806
HPARCS   017034    3412*  3495#
HRXCSR   017024    3406*  3491#
HRXDBU   017030    3411*  3493#
HTXCSR   017040    3416*  3497#
HTXDBU   017044    3420*  3499#
ICOUNT   001116    1074#  2932   2936*
INBUF    016150    1281   1285   2968   2996   3110   3114   3380#
INIFLG   001216    1122#  1182   1185*
INSTER= 104404     1145#  1289   3011   3119
INSTR  = 104403    1142#  1198   1209   1219   1230   1255   1265   1279   1292   1296   1300   1304   1323
                   1336
INSTR2   013364    2975   2984#
ISYM00= 000000      753#  1360   1367   1415   1422   1520   1527   1685   1692   1827   1835
JMRBY    001177    1105#  1307   2313   2350   2766
KEEPAD   001202    1110#  1206*  1208   1252   1254   2868
KEEPIV   001210    1113#  1217*  2869
LASTAD   001204    1111#  1235   1246   1260
LESS1    016370    2318   2368   2382   2465   2504   2573   3396*  3397*  3398*  3401#
LEVEL0= 000000      723#
LEVEL1= 000040      722#
LEVEL2= 000100      721#
```

H06


```
LEVEL3= 000140          720#
LEVEL4= 000200          719#    3401
LEVEL5= 000240          718#    3400
LEVEL6= 000300          717#
LEVEL7= 000340          716#    2321    2341    2372    2394    2432    2475    2489    2512    2520    2535    2593    2610
                        2641    2671
LIGHTS= 177570          683#    2892*
LIMITS  013476          3006    3016#
LOBITS  013550          2992*   3020    3034#   3035
LOCK    001114          1073#   2047*   2688*   2947
LOGICA  013002          2896#
LOKFLG  001221          1125#   1326    1327
LOLIM   013542          2989*   3018    3031#
LPCNT   001120          1075#   1173*   2931*   2932    2935*
LSTERR  001130          1079#   1178*   2889*   3139    3141*   3232*
MCOW    015106          2851    3277#
MCRLF   015263          3064    3297#
MDATA   016254          3084    3095    3384#
MEPASS  015353          2844    3307#
MEXT  = 010000          776#    2774
MEXTJ   015752          1305    3354#
MINT  = 004000          775#    1364    1419    1474    1524    1568    1607    1643    1689    1774    1832    1885    1938
                        1997    2052    2130    2209    2358    2448    2693
MLASTD  015013          1231    3266#
MLEVEL  015470          1266    3321#
MLOCK   015425          1324    3315#
MMULT   014734          1220    3258#
MPFAIL  015266          3230    3298#
MGM     015257          2982    3296#
MR      015406          1346    3312#
MRANGE  015176          1256    3287#
MREGAD  014653          1199    3249#
MRESET= 000400          766#    1359    1361    1414    1416    1469    1471    1519    1521    1563    1565    1602    1604
                        1638    1640    1684    1686    1768    1770    1826    1828    1879    1881    1932    1934    1992
                        1994    2049    2051    2124    2126    2204    2206    2315    2353    2355    2442    2444    2501
                        2546    2548    2689    2749    2768    2770
MSYNC   015515          1280    3325#
MTDATA= 040000          763#    1364    1419    1474    1524    1568    1607    1643    1689    1997    2209    2358    2408
                        2412    3427
MTITLE  014556          1184    3237#
MTSTPC  015412          1337    3313#
MULTD   001176          1104#   1222    1223    1315    2847
MVECTO  014631          1210    3245#
MWIRE4  015676          1301    3346#
MWIRE5  015631          1297    3339#
MWIRE6  015563          1293    3332#
NEXT    001112          1072#   1358*   1413*   1468*   1518*   1562*   1601*   1637*   1683*   1767*   1825*   1978*   1931*
                        1989*   2046*   2122*   2199*   2312*   2349*   2440*   2500*   2545*   2687*   2765*   2918    2937
                        3190
NOPAR = 000000          758#    1367    1422    1477    1610    1646    1777    2000    2048    2133    2212    2361    2451
                        2555    2692
ODDPAR= 001000          759#    2777
ODD8    016614          3440#
ONCE    001350          1183    1186#
OPTCLR  001175          1103#   1303
OUTCRY  012742          2845    2884#
```

```
OUTMUL  001700     1228    1253    1264#
OVRRUN= 040000      744#   1746    2176
PARAM = 104405     1148#   1200    1211    1232    1257    1267    1338
PARAM1  013416     2995#   3012
PARCSR  017032     1360*   1367*   1415*   1422*   1470*   1477*   1520*   1527*   1564*   1571*   1603*   1610*   1639*
                   1646*   1685*   1692*   1769*   1777*   1827*   1835*   1880*   1888*   1933*   1941*   1993*   2000*
                   2050*   2053*   2125*   2133*   2205*   2212*   2354*   2361*   2443*   2451*   2547*   2555*   2694*
                   2708*   2769*   2777*   3409*   3494#
PAREN = 001000      748#
PARER = 010000      746#   1721    1746    1752
PARERR  013472     2998    3000    3002    3011#   3017    3019    3021
PASCNT  001124     1077#   1175*   2891*   2892
PC    =%000007      678#   1056*   1207*   1273*   1379*   1391*   1395*   1434*   1446*   1450*   1486*   1498*   1502*
                   1540*   1545*   1581*   1583*   1622*   1658*   1705*   1724*   1743*   2010*   2016*   2031*   2225*
                   2237*   2249*   2258*   2277*   2288*   2842*   2864*   2870*   2873*   2882*   2896*   3399*   3421*
                   3438*   3442*   3454*   3459*   3471*
PFTAB   014550     3228    3234#
POPR0 = 012600      692#   3184
POP1SP= 005726      690#
POP2SP= 022626      694#
PS    = 177776      684#   1170*   1311*   2318*   2321*   2341*   2368*   2372*   2382*   2394*   2432*   2465*   2475*
                   2489*   2504*   2512*   2520*   2535*   2573*   2593*   2610*   2641*   2671*   2900*   3222*
PUSHR0= 010046      691#   3181
PUSH1S= 005746      689#
PUSH2S= 024646      693#
RECACT= 004000      730#   2011    2026    2034    2232    2238    2242    2251    2259    2280    2704    2726
REPLAY  012646     2864    2870    2872#
RESREG  014320     3171    3178#
RESTAR  014452     3208    3214#
RESTRT  013012     2865    2894    2900#
RESOS = 104407     1154#   3178
RETURN  001110     1071#   1180*   1345*   1347    2901*   2918*   2919    2937*   2938    3190*   3192    3233
RING  = 040000      727#
RINTEN= 000100      735#   2386    2398    2402    2426    2581    2595    2618    2642
ROTADO  001214     1117#   1227*   1239*   1241    1243*   1248    1251*   2859*   2862    2866*
RPOKE   016526     1379    1391    1395    1434    1446    1450    1486    1498    1502    1545    1583    1622    1658
                   1705    1724    1743    2010    2016    2031    2225    2237    2249    2258    2277    2288    3427#
                   3437
RTS   = 000004      739#
RUNIT   012552     2850    2856#   2863
RXCSR   017022     1368*   1380    1423*   1435    1478*   1487    1528*   1572*   1611*   1618*   1623    1647*   1654*
                   1659    1695*   1702*   1706    1725    2001*   2011    2026    2034    2135*   2146    2173    2214*
                   2232    2238    2241*   2242    2250*   2251    2254    2259    2262    2280    2283    2289    2322
                   2325*   2331*   2332*   2339*   2362*   2373    2379*   2386*   2395    2398*   2402*   2426*   2556*
                   2581*   2595*   2611    2618*   2642*   2652    2695*   2704    2726    2750    2781*   2817    2886
                   3404*   3490#
RXDBUF  017026     1375    1383    1398    1430    1438    1453    1482    1490    1505    1535    1546    1576    1584
                   1693    1712    1728    2136    2152    2177    2222    2266    2293    2380    2423    2425    2574
                   2612    2690    2731    2778    2820    3408*   3492#
RXDONE= 000200      734#
RXERR = 100000      743#   1721    1746    1749    1752    2176
R0    =%000000      671#   1189*   1192*   1194*   1376*   1384    1396*   1399    1431*   1439    1451*   1454    1483*
                   1491    1503*   1506    1537*   1547    1578*   1585    1711*   1716    1721*   1729    1746*   1749*
                   1752*   1785*   1788*   1794    1802*   1805    1843*   1846*   1852    1860*   1863    1896*   1899*
                   1905    1913*   1916    1949*   1952*   1958    1966*   1969    2082*   2083*   2096    2151*   2153
                   2176*   2178    2265*   2267    2292*   2294    2333*   2334*   2383*   2384*   2418*   2419*   2424*
```

```
                         2427    2469*   2470*   2506*   2507*   2529*   2530*   2575*   2649    2653*   2656    2732*   2733*
                         2734    2796*   2821    2836*   2838*   2852*   2927*   3048    3053*   3074*   3078*   3085    3182*
                         3200    3220*
R1       =%000001         672#   1190*   1192    1193*   1195*   1196    1383*   1384    1398*   1399    1438*   1439    1453*
                         1454    1490*   1491    1505*   1506    1546*   1547    1584*   1585    1712*   1716    1728*   1729
                         1792*   1793*   1794    1803*   1804*   1805    1850*   1851*   1852    1861*   1862*   1863    1903*
                         1904*   1905    1914*   1915*   1916    1956*   1957*   1958    1967*   1968*   1969    2085*   2093*
                         2096    2152*   2153    2177*   2178    2266*   2267    2293*   2294    2380*   2425*   2427    2612*
                         2649    2654*   2655*   2656    2731*   2734    2820*   2821    2893*   2896    3047    3054*   3065*
                         3067    3068    3069    3070    3201    3219*
R2       =%000002         673#   1191*   1194    1195    2563*   2564*   2587*   2588*   2635*   2636*   2785*   2786*   2791*
                         2792*   2800*   2801*   2806*   2807*   3046    3055*   3146*   3147*   3172*   3173*   3174*   3175*
                         3202    3218*
R3       =%000003         674#   1375*   1430*   1482*   1535*   1576*   1693*   1778*   1836*   1889*   1942*   2076*   2136*
                         2222*   2423*   2574*   2652*   2690*   2778*   2969*   2979*   3045    3056*   3075*   3076*   3077*
                         3078    3084*   3085*   3090*   3093*   3203    3217*
R4       =%000004         675#   1536*   1537    1539    1551*   1552    1577*   1578    1580    1589*   1590    2048*   2052
                         2083    2100*   2611*   2654    2692*   2694    2700    2708    2709    2732    2753*   2754    2779*
                         2780    2796    2813*   2814    2825    2968*   2972*   2973*   2974    2996*   2997    2999    3001
                         3003*   3004    3005    3025*   3026*   3044    3057*   3072*   3075    3079*   3080*   3081*   3148*
                         3149*   3150*   3151*   3152*   3153*   3154    3155    3156    3204    3216*
R5       =%000005         676#   2620*   2659    2952*   2954    2958    2988*   2989    2990    2991    2992    2993    2994
                         2995*   3004*   3007*   3008*   3009*   3016    3018    3020    3026    3027*   3043    3058*   3073*
                         3082*   3109*   3112*   3116*   3144*   3145*   3147    3148    3150    3205    3215*
SAVPC     001170         1098#   3039*   3196
SAVR0     001152         1091#   3048*   3053    3173    3537
SAVR1     001154         1092#   3047*   3054    3174    3539
SAVR2     001156         1093#   3046*   3055
SAVR3     001160         1094#   3045*   3056    3172    3535
SAVR4     001162         1095#   3044*   3057
SAVR5     001164         1096#   3043*   3058
SAVSP     001166         1097#   3207*   3214
SAVOS  = 104406          1151#   3143
SCOPE  = 104400          1133#   1403    1458    1510    1554    1592    1628    1664    1754    1815    1868    1921    1974
                         2037    2103    2186    2307    2342    2433    2490    2536    2672    2756    2827
SCOP1  = 104401          1136#   2099    2740
SEND   = 000020           770#   1774    1832    1885    1938    2052    2130    2448    2552    2693    2774
SEREC     001174         1102#   1299
SETFLG= 104411           1160#   1221    1294    1298    1302    1305    1325
SEVEN  = 004000           756#   1835
SEXMIT    001173         1101#   1295
SHIFT     001134         1084#   1377*   1389*   1393*   1432*   1444*   1448*   1484*   1496*   1500*   1538*   1579*   1621*
                         1657*   1703*   1723*   1741*   1781*   1798*   1839*   1856*   1892*   1909*   1945*   1962*   2008*
                         2014*   2029*   2070*   2074*   2084*   2094*   2140*   2144*   2164*   2169*   2223*   2235*   2248*
                         2257*   2275*   2286*   2403*   2415*   2453*   2461*   2715*   2719*   2795*   2810*   2811    2815
                         3436*
SIX    = 002000           755#
SP     =%000006           677#   1171*   1312*   2326*   2376*   2399*   2480*   2513*   2521*   2617*   2626*   2629*   2910*
                         2913    2915    2916    2919*   2938*   2947*   2952    2953*   2964    2965*   2988    2994*   3039
                         3065    3066*   3109    3117*   3126*   3127*   3128*   3129*   3130*   3131*   3132*   3133    3139
                         3141    3144    3182    3191*   3200*   3201*   3202*   3203*   3204*   3205*   3206*   3207    3214*
                         3215    3216    3217    3218    3219    3220    3223*   3473*
SPACNT= 014017           3069*   3088    3091*   3101#
SRD    = 002000           731#
STACK  = 001100           685#   1171    1312    3191    3223
STD    = 000010           738#
```

```
STFLG   001217    1123#  1174*
STPSYN= 000400     733#  1618   1654   1702   2135   2556
SVOS    013560    3043#
SWR   = 177570     682#  1186   1321   1334   2840   2922   2929   2945   3137   3179   3186   3188
SW00  = 000001     665#  1186
SW01  = 000002     664#  1334
SW02  = 000004     663#
SW03  = 000010     662#
SW04  = 000020     661#
SW05  = 000040     660#
SW06  = 000100     659#  2840
SW08  = 000400     658#  3186
SW09  = 001000     657#  2945
SW10  = 002000     656#  3188
SW11  = 004000     655#  2929
SW12  = 010000     654#
SW13  = 020000     653#  3137
SW14  = 040000     652#  2922
SW15  = 100000     651#
SYNCNO  001172    1100#  1283*  1287*  1990   2201   2229   2272   2578   2724
SYNEXT= 020000     752#  1470   1477   1564   1571   2125   2133   2354   2361   2443   2451   2769   2777
SYNINT= 030000     751#  1603   1610   1639   1646   1769   1777   1880   1888   1933   1941   1993   2000
                  2048   2050   2205   2212   2547   2555   2692
SYNSCH= 000020     737#  1368   1423   1478   1528   1572   1611   1647   1695   2001   2135   2214   2241
                  2250   2362   2556   2695   2750   2781
SYSTST= 014000     777#  2552
TEMP    016212    3074   3224*  3225*  3382#
TEMP1   001140    1086#  1378*  1390*  1394*  1433*  1445*  1449*  1485*  1497*  1501*  1539*  1542*  1543*
                  1580*  1620*  1656*  1704*  1722*  1742*  1745*  1748*  1751*  1780*  1786*  1838*  1844*
                  1891*  1897*  1944*  1950*  2009*  2015*  2030*  2224*  2236*  2276*  2287*  2404*  2410*
                  3429*  3440   3450*  3452*  3457   3467*  3469*
TEMP2   001142    1087#  3428*  3430*  3431*  3432*  3433   3440*  3444*  3457*  3461*
TEMP3   001144    1088#  3441*  3445*  3448*  3458*  3462*  3465*
TEMP4   001146    1089#
TEMP5   001150    1090#  2562*  2566*  2567   2586*  2590*  2591   2634*  2638*  2639
TKCSR   001100    1064#  2925   2970
TKDBR   001102    1065#  2927   2972   2978
TLAST = 012176    1340   2830#
TPCSR   001104    1066#  2956   2976
TPDBR   001106    1067#  2958*  2978*
TRPOK   014102    3129#
TRPREG  016764    3473#
TSTNO   001122    1076#  1179*  1357*  1412*  1467*  1517*  1561*  1600*  1636*  1682*  1766*  1824*  1877*
                  1930*  1988*  2045*  2121*  2198*  2311*  2348*  2439*  2499*  2544*  2686*  2764*
TST1    002224    1339   1345   1357#  2901   2902
TST10   004732    1767   1824#
TST11   005154    1825   1877#
TST12   005376    1878   1930#
TST13   005620    1931   1988#
TST14   006122    1989   2045#
TST15   006420    2046   2121#
TST16   006722    2122   2198#
TST17   007452    2199   2311#
TST18   007652    2312   2348#
TST19   010316    2349   2439#
TST2    002466    1358   1412#
```

```
TST20    010612        2440    2499#
TST21    011006        2500    2544#
TST22    011650        2545    2686#
TST23    012176        2687    2764#   2830
TST24 =  ****** U      2765
TST3     002730        1413    1467#
TST4     003156        1468    1517#
TST5     003356        1518    1561#
TST6     003526        1562    1600#
TST7     003712        1601    1636#
TST8     004076        1637    1682#
TST9     004500        1683    1766#
TTST     013112        1318*   1319*   1329*   1330*   1332*   1333*   2923#
TXCSR    017036        1359*   1361*   1364*   1370*   1371*   1373*   1374*   1414*   1416*   1419*   1425*   1426*   1428*
                       1429*   1469*   1471*   1474*   1480*   1481*   1519*   1521*   1524*   1530*   1531*   1533*   1534*
                       1563*   1565*   1568*   1574*   1575*   1602*   1604*   1607*   1613*   1614*   1616*   1617*   1638*
                       1640*   1643*   1649*   1650*   1652*   1653*   1684*   1686*   1689*   1697*   1698*   1700*   1701*
                       1768*   1770*   1774*   1778    1783*   1784*   1790*   1791*   1792    1801*   1803    1810    1826*
                       1828*   1832*   1836    1841*   1842*   1848*   1849*   1850    1859*   1861    1879*   1881*   1885*
                       1889    1894*   1895*   1901*   1902*   1903    1912*   1914    1932*   1934*   1938*   1942    1947*
                       1948*   1954*   1955*   1956    1965*   1967    1992*   1994*   1997*   2003*   2004*   2006*   2007*
                       2020*   2021*   2023*   2024*   2049*   2051*   2052*   2054    2059*   2060*   2061    2065*   2066*
                       2067    2072*   2073*   2076    2087*   2088*   2090    2124*   2126*   2130*   2138*   2139*   2142*
                       2143*   2167*   2168*   2204*   2206*   2209*   2216*   2217*   2219*   2220*   2315*   2353*   2355*
                       2358*   2364*   2365*   2408*   2412*   2413*   2414*   2442*   2444*   2448*   2455*   2456*   2459*
                       2460*   2466*   2469*   2476    2479*   2488*   2501*   2505*   2522    2534*   2546*   2548*   2552*
                       2582*   2594*   2619*   2633*   2689*   2693*   2698*   2699*   2702*   2703*   2711    2717*   2718*
                       2744*   2745    2749*   2768*   2770*   2774*   2783*   2789*   2798*   2804*   3414*   3427*   3433*
                       3434*   3435*   3496#
TXDBUF   017042        1779*   1837*   1890*   1943*   2057*   2134*   2159*   2452*   2585*   2632*   2700*   2709*   2780*
                       2814*   3418*   3498#
TXDONE=  000200         767#
TXINTE=  000100         768#   2505    2534    2582    2594    2619    2633
TYPDAT   014266        3160    3167    3170#
TYPE  =  104402        1139#   1184    1346    2843    2846    2851    2966    2981    3063    3094    3164    3168    3229
TYPMSG   014236        3158    3161#
USER  =  000000         774#
VOID  =  000001         741#
WRDCNT   014014        3067*   3096*   3099#
ZERO     012560        2857#
$E    =  000031         634#   1358    1359#   1413    1414#   1468    1469#   1518    1519#   1562    1563#   1601    1602#
                       1637    1638#   1683    1684#   1767    1768#   1825    1826#   1878    1879#   1931    1932#   1989
                       1990#   2046    2048#   2122    2123#   2199    2200#   2312    2313#   2349    2350#   2440    2441#
                       2500    2501#   2545    2546#   2687    2689#   2765    2766#
$N    =  000027         634#   1357    1359#   1412    1414#   1467    1469#   1517    1519#   1561    1563#   1600    1602#
                       1636    1638#   1682    1684#   1766    1768#   1824    1826#   1877    1879#   1930    1932#   1988
                       1990#   2045    2048#   2121    2123#   2198    2200#   2311    2313#   2348    2350#   2439    2441#
                       2499    2501#   2544    2546#   2686    2689#   2764    2766#   2830#
$Y    =  000012        1126#   1132    1135#   1138#   1141#   1144#   1147#   1150#   1153#   1156#   1159#   1162#
.     =  017440         779#    780     782     784     786     788     790     792     794     796     798     800     802
                        804     806     808     810     812     814     816     818     820     822     824     826     828
                        830     832     834     836     838     840     842     844     846     848     850     852     854
                        856     858     860     862     864     866     868     870     872     874     876     878     880
                        882     884     886     888     890     892     894     896     898     900     902     904     906
                        908     910     912     914     916     918     920     922     924     926     928     930     932
                        934     936     938     940     942     944     946     948     950     952     954     956     959
```

```
                          960     962     964     966     968     970     972     974     976     978     980     982     984
                          986     988     990     992     994     996     998    1000    1002    1004    1006    1008    1010
                         1012    1014    1016    1018    1020    1022    1024    1026    1028    1030    1032    1034    1039#
                         1046#   1049#   1060#   1381    1385    1400    1436    1440    1455    1488    1492    1507    1548
                         1586    1624    1660    1707    1717    1726    1730    1795    1806    1811    1853    1864    1906
                         1917    1959    1970    2012    2027    2035    2055    2062    2068    2097    2147    2154    2174
                         2179    2202    2233    2239    2243    2252    2255    2260    2263    2268    2281    2284    2290
                         2295    2323    2335    2351    2374    2385    2396    2420    2428    2471    2477    2508    2523
                         2531    2565    2568    2589    2637    2650    2657    2661    2666    2705    2712    2727    2735
                         2746    2751    2787    2793    2802    2808    2818    2822    3210    3226    3376#   3381#   3383#
                         3385#   3540#   3541#
.BEGIN   002032          1188    1311#
.CONVR   013644          1158    3063#
.EOP     012456          2765    2836#
.ERRTA   016772          3153    3477#
.HLT     014122          1042    3137#
.INSTE   013356          1146    2981#
.INSTR   013264          1143    2964#
.INST1   013276          2966#   2983
.MSG     013300          2964*   2967#
.PARAM   013366          1149    2988#
.PFAIL   014414          1040    1172    3200#   3221
.RESOS   013612          1155    3053#
.SAVOS   013552          1152    3039#
.SCOPE   013032          1134    2906#
.SCOP1   013214          1137    2945#
.SETFL   014022          1161    3109#   3120
.START   001246          1047    1170#   1180    2855
.TRPSR   014070          1044    3126#
.TRPTA   001222          1131#   3131
.TYPE    013232          1140    2952#
```

```
COMMEN      1#
ENDCOM      1#
ESCAPE      1#
GETPRI      1#
GETSWR      1#
HLT       695#   1382   1386   1401   1437   1441   1456   1489   1493   1508   1549   1597   1625   1661   1708
          1718   1727   1731   1796   1807   1812   1854   1865   1907   1918   1960   1971   2013   2028   2036
          2056   2063   2069   2098   2148   2155   2175   2180   2234   2240   2244   2253   2256   2261   2264
          2269   2282   2285   2291   2296   2324   2340   2375   2387   2397   2429   2430   2472   2478   2509
          2524   2525   2600   2647   2651   2658   2662   2667   2706   2713   2728   2736   2747   2752   2819
          2823
MULT        1#
NEWTST      1#
POP         1#
PRGEND    634#   2829
PRGFRT    634#    635
PUSH        1#
PUSSYF    634#   2783   2797
REPORT      1#
RSETUP    634#   1359   1414   1469   1519   1563   1602   1638   1684   1992   2204   2353
SETPRI      1#
SETUP       1#
SKIP        1#
SLASH       1#
STARS       1#
SWRSU       1#
TSETUP    634#   1768   1826   1879   1932   2124   2442   2546   2768
TYPBIN      1#
TYPDEC      1#
TYPNAM      1#
TYPNUM      1#
TYPOCS      1#
TYPOCT      1#
TYPTXT      1#
$BEGIN    634#   1308
$BINAR    634#   2757
$BUFFE    634#   3377
$CABLE    634#
$CATCH    634#    778
$CLRVE    634#   1186
$CONVR    634#   3060
$DNA      634#   1755
          634#   2829
$GL  L    634#   1219   1292   1296   1300   1304   1323
$GETPA    634#   1198   1209   1229   1255   1264   1336
$GETSY    634#   1274
$HEADE    634#    635
$HLT      634#   3134
$INSTR    634#   2961
$ISOB     634#   1511   1555
$MATCH    634#   2673
$MRR      634#
$MRRW     634#
$MRW      634#
$MSG      634#   3237
$PARAM    634#   2985
```

B07


```
$PFAIL     634#   3197
$POKE      634#   1783   1799   1941   1847   1894   1900   1947   1953   2059   2065   2071   2086   2138   2141
          2166    2455   2458   2698   2702   2716
$POKER     634#   1372   1427   1479   1532   1573   1615   1651   1699   2005   2019   2022   2218   2363
$RCNET     634#   1665
$RECAC     634#
$REG       634#   3036
$RESET     634#   1359   1361   1414   1416   1469   1471   1519   1521   1563   1565   1602   1604   1638   1640
          1684    1686   1768   1770   1826   1828   1879   1881   1932   1934   1992   1994   2124   2126   2204
          2206    2315   2353   2355   2442   2444   2501   2546   2548   2689   2749   2768   2770
$RXACT     634#
$SCOPE     634#   2903
$SCOP1     634#   2942
$SETFL     634#   3103
$SETVE     634#   1036
$START     634#   1162
$STRIP     634#   1593   1629
$SYMBO     634#    648
$SYNCR     634#   1368   1423   1528   1611   1647   1695   2001   2214
$TRAPS     634#   1126
$TRPAR     634#   1816   1869   1922
$TRPDE     634#   1132   1135   1138   1141   1144   1147   1150   1153   1156   1159
$TRPSR     634#   3121
$TSTNO     634#   1357   1412   1467   1517   1561   1600   1636   1682   1766   1824   1877   1930   1988   2045
          2121    2198   2311   2348   2439   2499   2544   2686   2764
$TYPE      634#   2949
$UNIBU     634#
$VARIA     634#   1059
$WORDF     634#
$WORDO     634#   1349   1404   1459
$WORDP     634#
$$ESCA       1#
$$NEWT       1#
$$SKIP       1#
.EQUAT       1#
.HEADE       1#
.KT11        1#
.SETUP       1#
.SWRHI       1#
.$ACT1       1#
.$APTB       1#
.$APTH       1#
.$APTY       1#
.$ASTA       1#
.$CATC       1#
.$CMTA       1#
.$DB2D       1#
.$DB20       1#
.$DIV        1#
.$EOP        1#
.$ERRO       1#
.$ERRT       1#
.$MULT       1#
.$POWE       1#
.$RAND       1#
.$RDDE       1#
```

C07


```
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#
.$SIZE      1#
.$SUPR      1#
.$TRAP      1#
.$TYPB      1#
.$TYPD      1#
.$TYPE      1#
.$TYPO      1#
.$40CA      1#
[illegible] 1#
```

```
ADC     3445  3462
ADD     1194  1195  1245  2856  2857  2875  2877  2879  2953  2965  3027  3066  3077  3117  3131
        3150  3153  3473
ASL     3007  3008  3009  3129  3149  3151  3391  3392  3393  3394  3395
ASR     3079  3080  3081  3431
BCC     1787  1845  1898  1951  2411  3466
BCS     1244  2860  3449
BEQ     1317  1322  1328  1335  1385  1400  1440  1455  1492  1507  1548  1586  1717  1730  1735
        1737  1739  1795  1806  1853  1864  1906  1917  1959  1970  2012  2027  2091  2097  2154
        2179  2227  2243  2252  2260  2268  2295  2428  2650  2657  2661  2666  2705  2722  2735
        2751  2822  2841  2848  2863  2894  2923  2946  2955  2975  2998  3006  3089  3140  3158
        3167  3171  3189
BGE     2568  2592  2640
BGT     3002
BHI     1247  3017
BIC     1370  1373  1425  1428  1480  1530  1533  1574  1613  1616  1649  1652  1697  1700  1784
        1791  1793  1804  1842  1849  1851  1862  1895  1902  1904  1915  1948  1955  1957  1968
        2003  2006  2020  2023  2060  2066  2073  2088  2139  2143  2168  2216  2219  2241  2325
        2339  2364  2386  2398  2408  2413  2426  2456  2460  2479  2488  2534  2594  2595  2618
        2619  2633  2642  2655  2699  2703  2718  2733  2789  2804  3076  3130  3152  3398  3427
        3432  3434  3452  3469
BICB    2973  3003
BIS     1241  1248  1359  1361  1368  1371  1374  1414  1416  1423  1426  1429  1469  1471  1478
        1481  1519  1521  1528  1531  1534  1543  1563  1565  1572  1575  1602  1604  1611  1614
        1617  1618  1638  1640  1647  1650  1653  1654  1684  1686  1695  1698  1701  1702  1768
        1770  1783  1788  1790  1801  1826  1828  1841  1846  1848  1859  1879  1881  1894  1899
        1901  1912  1932  1934  1947  1952  1954  1965  1992  1994  2001  2004  2007  2021  2024
        2049  2051  2059  2065  2072  2087  2124  2126  2135  2138  2142  2167  2204  2206  2214
        2217  2220  2250  2315  2331  2332  2353  2355  2362  2365  2379  2402  2412  2414  2442
        2444  2455  2459  2466  2468  2501  2505  2546  2548  2556  2581  2582  2689  2695  2698
        2702  2717  2744  2749  2768  2770  2781  2783  2798  3433  3435  3450  3467
BISB    2083  3004
BIT     1186  1321  13[illegible]  2011  2026  2034  2090  2232  2238  2242  2251  2259  2290  2704  2726
        2750  2840  2862  2922  2929  2945  3137  3185  3188
BITB    3020
BLO     3019
BLOS    2933
BLT     1197  3000
BMI     1225  1381  1436  1488  1726  2055  2068  2174  2230  2273  2290  2323  2351  2374  2396
        2477  2523  2579  2625  2725  2746  2818
BNE     1183  1187  1282  1286  1314  1553  1591  1627  1663  1710  1799  1857  1910  1963  2033
        2035  2075  2095  2101  2145  2150  2170  2172  2233  2239  2279  2281  2335  2385  2416
        2420  2462  2471  2508  2531  2565  2589  2637  2720  2727  2755  2787  2793  2802  2808
        2812  2816  2826  2839  2850  2930  2980  3021  3029  3083  3087  3092  3097  3111  3115
        3138  3160  3187  3226  3437  3447  3464
BPL     1624  1660  1707  1811  1991  2062  2147  2202  2255  2263  2284  2314  2712  2767  2926
        2957  2971  2977  3180
BR      1253  1284  1288  1290  1320  1331  1344  2319  2369  2473  2510  2607  2648  2707  2871
        2908  2914  2917  2924  2928  2959  2983  3010  3012  3113  3120  3210  3451  3468
CLC     1242  1541  2089  2409  2858
CLR     1173  1175  1177  1178  1193  1226  1227  1240  1536  1577  1785  1843  1896  1949  2082
        2085  2333  2383  2418  2469  2506  2529  2562  2563  2586  2587  2634  2635  2779  2989
        2890  2935  2995  3224  3231  3232  3428  3441  3458
CLRB    1174  1176  1283  2934  3093  3112  3142
CMP     1196  1246  1384  1399  1439  1454  1491  1506  1547  1585  1716  1729  1736  1738  1794
        1805  1852  1863  1905  1916  1958  1969  2096  2153  2178  2267  2294  2427  2567  2591
```

E07


```
        2639  2649  2656  2659  2665  2734  2811  2821  2915  2932  3016  3018  3139
CMPB    1281  1285  2754  2974  2997  2999  3001  3005  3110  3114
COMB    1185
DEC     1626  1662  1709  1734  1798  1856  1909  1962  2032  2074  2094  2144  2149  2169  2171
        2226  2274  2278  2415  2461  2580  2624  2719  2721  2786  2792  2801  2807  2810  2838
        2979  3082  3096  3436  3446  3463
DECB    3028  3086  3091
EMT      695
HALT     781   783   785   787   789   791   793   795   797   799   801   803   805   807   809
         811   813   815   817   819   821   823   825   827   829   831   833   835   837   839
         841   843   845   847   849   851   853   855   857   859   861   863   865   867   869
         871   873   875   877   879   881   883   885   887   889   891   893   895   897   899
         901   903   905   907   909   911   913   915   917   919   921   923   925   927   929
         931   933   935   937   939   941   943   945   947   949   951   953   955   957   959
         961   963   965   967   969   971   973   975   977   979   981   983   985   987   989
         991   993   995   997   999  1001  1003  1005  1007  1009  1011  1013  1015  1017  1019
        1021  1023  1025  1027  1029  1031  1033  1035  1050  2853  3183  3209
INC     1551  1589  2334  2384  2419  2470  2507  2530  2564  2566  2588  2590  2636  2638  2750
        2813  2891  2931  3185  3225  3405  3407  3410  3413  3415  3417  3419
INCB    2100
JMP     1047  1188  1228  1263  1347  1747  1750  1753  2203  2352  2583  2855  2865  2902  3133
        3192  3233
JSR     1207  1273  1379  1391  1395  1434  1446  1450  1486  1498  1502  1540  1545  1581  1583
        1622  1658  1705  1724  1743  2010  2016  2031  2225  2237  2249  2258  2277  2288  2842
        2864  2870  2873  2896
MOV     1170  1171  1172  1179  1180  1189  1190  1191  1192  1206  1208  1217  1218  1239  1251
        1252  1254  1311  1312  1318  1319  1329  1330  1332  1333  1345  1357  1358  1360  1364
        1367  1375  1376  1377  1378  1383  1389  1390  1393  1394  1396  1398  1412  1413  1415
        1419  1422  1430  1431  1432  1433  1438  1444  1445  1448  1449  1451  1453  1467  1468
        1470  1474  1477  1482  1483  1484  1485  1490  1496  1497  1500  1501  1503  1505  1517
        1518  1520  1524  1527  1535  1537  1538  1539  1546  1561  1562  1564  1568  1571  1576
        1578  1579  1580  1584  1600  1601  1603  1607  1610  1619  1620  1621  1636  1637  1639
        1643  1646  1655  1656  1657  1682  1683  1685  1689  1692  1693  1694  1703  1704  1711
        1712  1720  1721  1722  1723  1728  1741  1742  1745  1746  1748  1749  1751  1752  1766
        1767  1769  1774  1777  1778  1780  1781  1792  1802  1803  1824  1825  1827  1832  1835
        1836  1838  1839  1850  1860  1861  1877  1878  1880  1885  1888  1889  1891  1892  1903
        1913  1914  1930  1931  1933  1938  1941  1942  1944  1945  1956  1966  1967  1988  1989
        1993  1997  2000  2006  2009  2014  2015  2025  2029  2030  2045  2046  2047  2048  2050
        2052  2053  2070  2076  2084  2121  2122  2125  2130  2133  2136  2137  2140  2151  2152
        2158  2164  2176  2177  2198  2199  2205  2209  2212  2221  2222  2223  2224  2235  2236
        2248  2257  2265  2266  2270  2275  2276  2286  2287  2292  2293  2311  2312  2316  2317
        2318  2321  2326  2327  2328  2336  2337  2341  2348  2349  2354  2358  2361  2366  2367
        2368  2372  2376  2377  2380  2382  2389  2390  2394  2399  2403  2404  2421  2422  2423
        2424  2425  2432  2439  2440  2443  2448  2451  2453  2463  2464  2465  2475  2480  2481
        2482  2485  2486  2489  2499  2500  2502  2503  2504  2512  2513  2514  2520  2521  2532
        2533  2535  2544  2545  2547  2552  2555  2569  2570  2571  2572  2573  2574  2575  2576
        2593  2596  2597  2598  2599  2610  2611  2612  2613  2614  2615  2616  2617  2620  2626
        2629  2641  2643  2644  2645  2646  2652  2653  2654  2671  2686  2687  2688  2690  2692
        2693  2694  2708  2710  2715  2731  2732  2764  2765  2769  2774  2777  2778  2785  2791
        2795  2796  2800  2806  2820  2836  2852  2866  2868  2869  2872  2874  2876  2878  2880
        2881  2892  2893  2900  2901  2910  2911  2913  2916  2918  2919  2927  2936  2937  2938
        2947  2952  2964  2968  2969  2988  2989  2990  2991  2994  2996  3025  3026  3039  3043
        3044  3045  3046  3047  3048  3053  3054  3055  3056  3057  3058  3065  3067  3070  3072
        3074  3075  3084  3109  3126  3128  3132  3141  3144  3146  3147  3148  3154  3155  3156
        3172  3173  3174  3175  3182  3190  3191  3200  3201  3202  3203  3204  3205  3206  3207
        3208  3214  3215  3216  3217  3218  3219  3220  3221  3222  3223  3396  3404  3406  3409
```

F07


```
          3409  3411  3412  3414  3416  3418  3420  3440  3442  3457  3459
MOVB      1297  1779  1837  1890  1943  2057  2134  2159  2452  2585  2632  2700  2709  2780  2814
          2958  2972  2978  2992  2993  3068  3069  3073  3078  3085  3090  3116  3163
NOP       1291  1329  1330  2897  2898  2899
RESET     2837  2895
RETURN    1341  3236
ROL       1243  1542  2859
ROR       1786  1844  1897  1950  2410  3429  3430  3444  3448  3461  3465
RORB      2093
RTI       2329  2391  2400  2483  2518  2528  2621  2628  2630  2920  2939  2948  2960  2984  3030
          3049  3059  3098  3118  3193  3475
RTS       1056  2882  3399  3421  3438  3454  3471
SEC       2092
SUB       3127  3145  3397
TRAP      1133  1136  1139  1142  1145  1148  1151  1154  1157  1160
TST       1313  1810  2322  2395  2476  2711  2745  2815  2849  2912  3159  3166  3170  3179
TSTB      1182  1223  1315  1327  1380  1435  1487  1552  1590  1623  1659  1706  1725  1990  2054
          2061  2067  2146  2173  2201  2229  2254  2262  2272  2283  2289  2313  2350  2373  2522
          2578  2724  2766  2817  2825  2847  2925  2954  2956  2970  2976  3088  3157
.ASCII    3354  3361
.ASCIZ    3237  3245  3249  3258  3266  3275  3277  3287  3296  3297  3298  3307  3312  3313  3315
          3321  3325  3332  3339  3346  3370  3506  3508  3514  3520  3528
.BLKW     3541
.BYTE     1100  1101  1102  1103  1104  1105  1122  1123  1124  1125  1204  1205  1215  1216  1236
          1237  1261  1262  1271  1272  1342  1343  2885  3195  3235  3534  3536  3538
.ENABL       1   630   632   634
.END      3542
.ENDC     1359  1375  1414  1430  1469  1482  1519  1535  1544  1563  1576  1582  1602  1618  1638
          1654  1679  1684  1734  1768  1826  1861  1879  1914  1932  1967  1990  2047  2123  2200
          2313  2350  2441  2501  2546  2688  2766  2768  2795  2810
 EQUIV     695
.EVEN     1106  3376  3531
.IF       1358  1368  1413  1423  1468  1478  1518  1528  1541  1562  1572  1582  1601  1611  1637
          1647  1679  1683  1734  1767  1825  1860  1879  1913  1931  1966  1989  2046  2122  2199
          2312  2349  2440  2500  2545  2687  2765  2766  2783  2797
.IFF      1358  1359  1368  1413  1414  1423  1468  1469  1482  1518  1519  1528  1562  1563  1576
          1601  1602  1611  1637  1638  1647  1683  1684  1767  1768  1825  1826  1861  1878  1879
          1913  1931  1932  1966  1989  1990  2046  2047  2122  2123  2199  2200  2312  2313  2349
          2350  2440  2441  2500  2501  2545  2546  2687  2688  2765  2795  2810
.IFT      1368  1423  1478  1528  1572  1611  1647  1679  1734  1860  1913  1966  2783  2797
.IIF      1358  1359  1413  1414  1468  1469  1518  1519  1562  1563  1601  1602  1637  1638  1683
          1684  1767  1768  1825  1826  1878  1879  1931  1932  1989  1990  2046  2047  2122  2123
          2199  2200  2312  2313  2349  2350  2440  2441  2500  2501  2545  2546  2687  2688  2765
          2766  2824
.IRP      1132  1135  1138  1141  1144  1147  1150  1153  1156  1159  1357  1358  1412  1413  1467
          1468  1517  1518  1561  1562  1600  1601  1636  1637  1682  1683  1766  1767  1824  1825
          1877  1878  1930  1931  1988  1989  2045  2046  2121  2122  2198  2199  2311  2312  2348
          2349  2439  2440  2499  2500  2544  2545  2686  2687  2764  2765  2830
.LIST        1     4   629   634   648  1126  1135  1138  1141  1144  1147  1150  1153  1156  1159
          1162  1357  1358  1359  1412  1413  1414  1467  1468  1469  1517  1518  1519  1561  1562
          1563  1600  1601  1602  1636  1637  1638  1682  1683  1664  1766  1767  1768  1824  1825
          1826  1877  1878  1879  1930  1931  1932  1988  1989  1990  2045  2046  2047  2048  2121
          2122  2123  2198  2199  2200  2311  2312  2313  2348  2349  2350  2439  2440  2441  2499
          2500  2501  2544  2545  2546  2686  2687  2688  2689  2764  2765  2766  2830
.MACRO       1   634
.NLIST       1     4   634   648  1126  1135  1138  1141  1144  1147  1150  1153  1156  1159  1162
```

G07


```
          1357   1358   1359   1412   1413   1414   1467   1468   1469   1517   1518   1519   1561   1562   1563
          1600   1601   1602   1636   1637   1638   1682   1693   1684   1766   1767   1768   1824   1825   1826
          1877   1878   1879   1930   1931   1932   1988   1389   1990   2045   2046   2047   2048   2121   2122
          2123   2198   2199   2200   2311   2312   2313   2348   2349   2350   2439   2440   2441   2499   2500
          2501   2544   2545   2546   2686   2687   2688   2689   2764   2765   2766   2830
.PAGE        4     48    633    667    778   1036   1099
.RADIX    1357   1358   1359   1412   1413   1414   1467   1468   1469   1517   1518   1519   1561   1562   1563
          1600   1601   1602   1636   1637   1638   1682   1683   1684   1766   1767   1768   1824   1825   1826
          1877   1878   1879   1930   1931   1932   1988   1989   1990   2045   2046   2047   2048   2121   2122
          2123   2199   2199   2200   2311   2312   2313   2348   2349   2350   2439   2440   2441   2499   2500
          2501   2544   2545   2546   2686   2687   2688   2689   2764   2765   2766   2830
.REM         5     49
.REPT      780
.TITLE     646


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *,DDDUAA.SEQ/SOL/CRF/PAGNUM/NL:TOC/DS:ERFZ=SYSMAC.CO,DDDUAA.P11
 RUN-TIME: 40 51 5 SECONDS
 RUN-TIME RATIO: 381/97=3.8
 CORE USED:  44K  (87 PAGES)
```