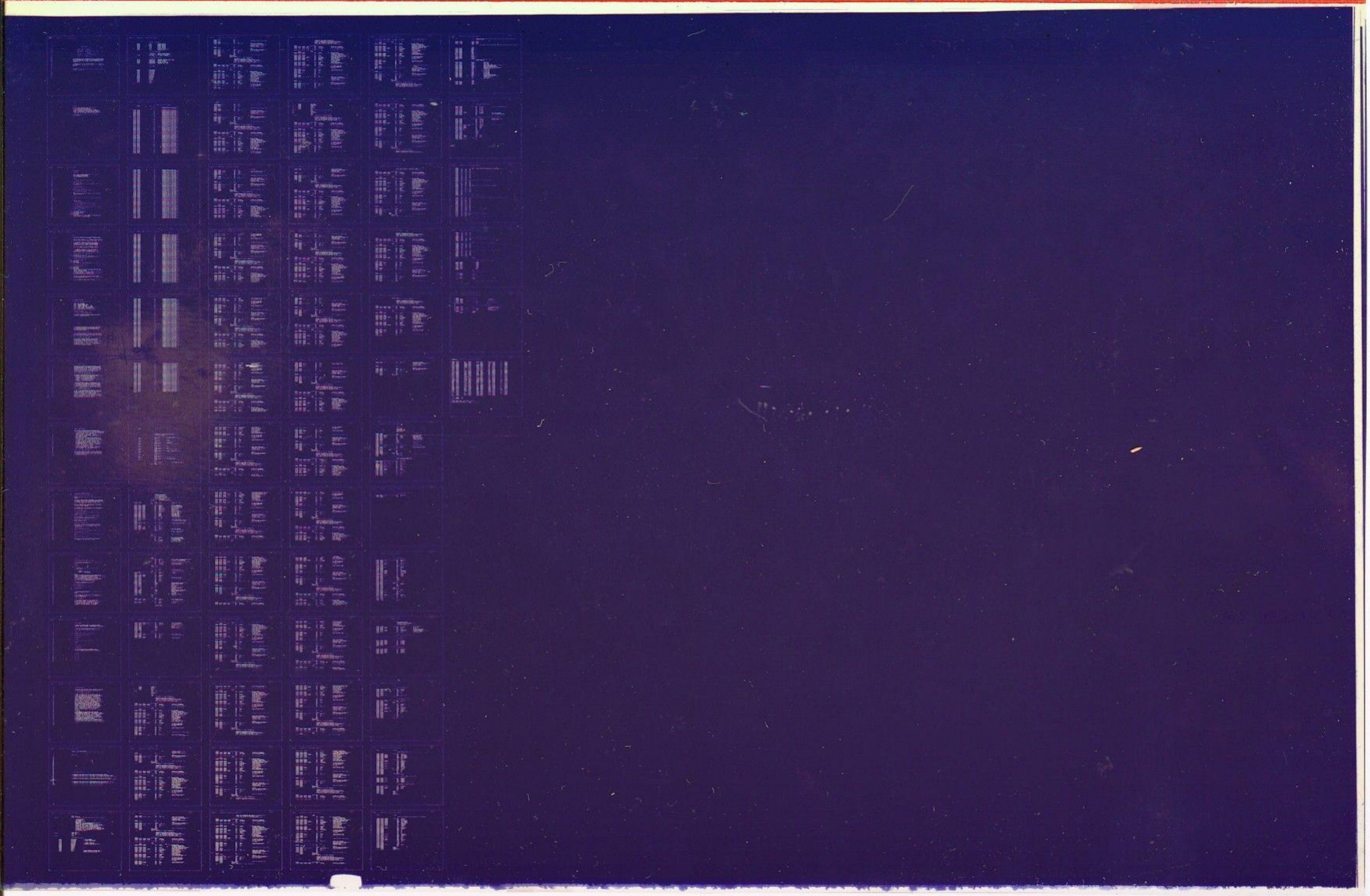
DH11

DH11 SPD SEL LOG TST
CZDHDD0

AH-FG23D-MC
1 OF 1 OCT 1985

digital
MADE IN USA



```
.REM !
```

IDENTIFICATION

PRODUCT CODE: AC-8456D-MC

PRODUCT NAME: CZDHDDO DH11 SPD SEL LOG TST

DATE: 12-JUNE-1985

MAINTAINER: NAC SOFTWARE ENGINEERING

AUTHOR: MICHAEL DAVIS

```
1
2                                   1.      ABSTRACT
3
4                                           THE DH11 SPEED SELECTION LOGIC TEST VERIFIES
5                                           THAT THE SPEED SELECTION FUNCTIONS
6                                           OF THE LINE PARAMETER REGISTER OPERATE PROPERLY
7                                           FOR EACH TRANSMITTER AND RECEIVER LINE.  TRANSMITTER
8                                           TIMING IS CHECKED FIRST, AND THEN RECEIVER TIMING IS
9                                           TESTED.  THE PROGRAM USES A RELATIVE TIMING COMPARISION
10                                          TO DETERMINE IF LINE SPEED SELECTION IS CORRECT.
11
12                                  NOTE:   THE EXTERNAL CLOCK FUNCTIONS (SPEED CODES 16 AND 17)
13                                          ARE NOT TESTED.
14
```

```
                        2.      REQUIREMENTS

                        2.1     EQUIPMENT

                                PDP-11 FAMILY STANDARD COMPUTER WITH 4KW OF MEMORY
                                ASR-33 TELETYPE OR EQUIVALENT
                                DH11 ASYNCHRONOUS MULTIPLEXER
                                DM11 MAINTENANCE CARD INSTALLED

                        2.2     STORAGE

                                THE PROGRAM LOADS INTO 4KW OF MEMORY

                        3.      LOADING PROCEDURE

                                THE STANDART PROCEDURE FOR LOADING ABSOLUTE BINARY TAPES
                                IS TO BE USED

                        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

                                SW00=1

                        4.1.3   TO START PROGRAM AT SELECTED TEST AFTER PROGRAM RESTART

                                SW01=1

                        4.2     STARTING ADDRESS

                                THE STARTING ADDRESS FOR ALL TESTS IS 000200

                                THE RESTART ADDRESS FOR ALL TESTS I 0002000

                                THE STARTING ADDRESS TO ENTER A SELECTED TEST IS 000200

                        4.3     PROGRAM AND/OR OPERATOR ACTION

                        4.3.1   INITIAL PROGRAM START

                        4.3.1.1 LOAD PROGRAM INTO MEMORY
                        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 "DH11 SPEED SELECTION LOGIC TEST"
                                AND WILL THEN TYPE "VECTOR ADDRESS- AND WAIT FOR AN
                                INPUT FROM THE TELETYPE KEYBOARD.
```

```
                4.3 (CONT'D)

                4.3.1.6 TYPE IN THE ADDRESS OF THE RECEIVER INTERRUPT VECTOR
                        FOR THE DH11 TO BE TESTED FOLLOWED BY <CARRIAGE RETURN>

                NOTE:   WORDS IN ANGLE BRACKETS, I.E. <CARRIAGE RETURN> MEAN THAT
                        THE TELETYPE KEY WITH THE NAMED FUNCTION SHOULD BE STRUCK

                        IF AN INCORRECT ADDRESS IS ENTERED, THE PROGRAM
                        WILL TYPE "?" AND WILL REPEAT THE SECOND MESSAGE OF 4.3.1.5
                4.3.1.7 THE PROGRAM WILL TYPE "CONTROL REGISTER ADDRESS-"
                        AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD

                4.3.1.8 TYPE IN THE ADDRESS OF THE SYSTEM CONTROL REGISTER OF THE
                        DH11 TO BE TESTED FOLLOWED BY <CARRIAGE RETURN>

                        IF AN INCORRECT ADDRESS IS TYPED, THE PROGRAM WILL TYPE
                        "?" AND WILL THEN REPEAT THE MESSAGE OF 4.3.1.7
                4.3.1.9 THE PROGRAM WILL TYPE "R" TO INDICATE THAT IT IS
                        ABOUT TO START TESTING, AND THEN TESTING WILL BEGIN

                4.3.2   PROGRAM RESTART WITH ALL SWITCHES DOWN

                4.3.2.1 PERFORM 4.3.1.2 TO 4.3.1.5
                4.3.2.2 THE PROGRAM WILL TYPE "DH11 SPEED SELECTION LOGIC TEST"
                        AND WILL THEN CONTINUE AS DESCRIBED IN 4.3.1.9

                4.3.3   PROGRAM RESTART WITH SW00=1

                4.3.3.1 LOAD ADDRESS 000200
                4.3.3.2 SET SW01=1
                4.3.3.3 PRESS START
                4.3.3.4 THE PROGRAM WILL PERFORM AS DESCRIBED IN 4.3.1.5 TO 4.3.1.9

                4.3.4   PROGRAM RESTART WITH SW01=1

                4.3.4.1 LOAD ADDRESS 000200
                4.3.4.2 SET SW01=1
                4.3.4.3 PRESS START
                4.3.4.4 THE PROGRAM WILL TYPE "DH11 SPEED SELECTION LOGIC TEST"
                        AND WILL THEN TYPE "TEST PC-" AND WILL 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 <CARRIAGE RETURN>
                4.3.4.6 THE PROGRAM WILL TYPE R TO INDICATE THAT IT HAS STARTED
                        AND WILL START TESTING AT THE SELECTED TEST.

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

                NOTE:   IF IT IS DESIRED TO LOOP ON THE TEST THAT IS SELECTED
                        SET SW14=1 BEFORE ENTERING THE TEST ADDRESS
```

```
1
2                                   5.      OPERATING PROCEDURE
3
4                                   5.1     OPERATIONAL SWITCH SETTINGS
5
6                                           SW15=1, HALT ON ERROR
7                                           SW14=1, LOOP ON CURRENT TEST
8                                           SW13=1, SUPPRESS ERROR TYPEOUT
9                                           SW11=1, INHIBIT ITERATIONS
10                                          SW10=1, ESCAPE TO NEXT TEST ON ERROR
11                                          SW09=1, FREEZE VARIABLE PARAMETER IN CURRENT TEST
12                                          SW01=1, START PROGRAM AT SELECTED TEST
13                                          SW00=1, CHANGE PARAMETERS AT PROGRAM RESTART
14
15                                  5.2     SUBROUTINE ABSTRACTS
16
17                                  5.2.1   TRAPCATCHER (LOCATIONS 000000-000776)
18
19                                          THIS ROUTINE IS USED TO INTERCEPT UNEXPECTED INTERRUPTS
20                                          AND TRAPS.  THE AREA FROM 000000-000776 IS LOADED
21                                          WITH THE FOLLOWING SEQUENCE
22
23                                                  2
24                                                  0
25                                                  4
26                                                  0
27                                                  ...
28                                                  772
29                                                  0
30                                                  776
31                                                  0
32
33                                          IF AN UNEXPECTED INTERRUPT OR TRAP OCCURS, THE PROGRAM
34                                          WILL HALT WITH THE PC 2 GREATER THAN THE ADDRESS TO
35                                          WHICH THE PROGRAM TRAPPED.  THE PROCESSOR STACK MAY
36                                          BE EXAMINED TO DETERMINE WHERE THE PROGRAM WAS WHEN THE
37                                          TRAP OR INTERRUPT OCCURED.
38
39                                  5.2.2   START   (PROGRAM INITIALIZATION)
40
41                                          THIS ROUTINE INITIALIZES ALL PROGRAM FLAGS AND COUNTERS,
42                                          TYPES THE PROGRAM TITLE MESSAGE, AND INPUTS THE VECTOR
43                                          AND CONTROL REGISTER ADDRESSES OF THE DH11 TO BE TESTED.
44
45                                  5.2.3   BEGIN   (PROGRAM START AND RESTART)
46
47                                          THIS ROUTINE IS ENTERED IMMEDIATLY AFTER "START"
48                                          AND EACH TIME A PROGRAM PASS HAS BEEN COMPLETED.
49                                          THE ROUTINE SETS UP THE PROCESSOR STACK AND STATUS WORD
50                                          AND THEN TRANSFERS CONTROL TO THE TEST AT WHICH
51                                          TESTING WILL BEGIN.  IF SW01=0 WHEN THIS ROUTINE IS ENTERD
52                                          TESTING WILL START AT T1 (TEST 1).  IF SW01=1 WHEN
53                                          THIS ROUTINE IS ENTERED, TESTING WILL START AT THE PC
54                                          ENTERED FROM THE TELETYPE KEYBOARD.
```

```
                5.2.4   EOP     (END OF PASS)

                        THIS ROUTINE IS ENTERED ONCE PER PASS AFTER ALL TESTS
                        HAVE BEEN COMPLETED.  THIS ROUTINE TYPES THE MAINDEC
                        IDENTIFICATION CODE OF THE PROGRAM, CLEARS ERROR FLAGS
                        AND UPDATES THE PASS COUNT.  IF THE PROGRAM WAS LOADED
                        UNDER ACT11 OR DDP, THE ROUTINE CHECKS FOR RETURN TO
                        THE ACT11 OR DDP MONITOR.  IF THE PROGRAM IS NOT UNDER
                        MONITOR CONTROL, THE ROUTINE TRANSFERS TO BEGIN.

                5.2.5   SCOPER  (SCOPE LOOP AND ITERATION HANDLER)

                        THIS ROUTINE IS ENTERED EACH TIME A TEST IS COMPLETED.
                        THE ROUTINE CHECKS FOR THE FOLLOWING UPON ENTRY
                        A) IF SW10=1, THE ROUTINE WILL TRANSFER TO THE NEXT TEST
                           IN SEQUENCE, AFTER CLEARING ERROR FLAGS.
                        B) IF SW11=1, THE ROUTINE WILL TRANSFER TO THE NEXT TEST
                           SEQUENCE, AFTER CLEARING ERROR FLAGS.
                        C) IF SW14=1, THE ROUTINE WILL LOOP ON THE CURRENT TEST
                           REGARDLESS OF THE ITERATION COUNT.

                        IF NONE OF THE ABOVE IS TRUE, THE ROUTINE WILL ADD 1
                        TO THE COUNT OF TEST ITERATIONS, AND COMPARE THIS VALUE
                        TO THE NUMBER OF ITERATIONS THAT SHOULD BE PERFORMED.
                        IF THESE NUMBERS ARE EQUAL, THE ROUTINE WILL TRANSFER
                        TO THE NEXT TEST IN SEQUENCE.  THE THE NUMBERS ARE NOT
                        EQUAL, THE TEST CURRENTLY IN PROGRESS WILL BE REPEATED.

                5.2.6   SCOP1R  (FREEZE ON CURRENT DATA)

                        THE CALL TO THIS ROUTINE FOLLOWS IMMEDIATLY AFTER THE
                        THE CALL TO THE ERROR HANDLER IN THOSE TESTS THAT HAVE
                        VARIABLE PARAMETERS.  THIS ROUTINE IS ALWAYS ENTERED IN
                        THOSE TESTS, WHETHER OR NOT AN ERROR OCCURS.
                        IF SW09=1, THE ROUTINE WILL TRANSFER CONTROL BACK TO THE
                        TEST AT A POINT WHICH WILL ALLOW REPEATING THE FUNCTION
                        UNDER TEST CONTINUOUSLY WITH THE SAME DATA.  IF THIS
                        OPTION IS SELECTED, THE ROUTINE "SCOPER" IS NEVER ENTERED
                        AND ITERATION COUNTS WILL NOT BE UPDATED.
```

```
                        5.2.7   ERRORS  (ERROR HANDLER)

                                THIS ROUTINE IS ENTERED UPON ERROR DETECTION ONLY.
                                WITH ALL CONSOLE SWITCHES DOWN, THE ROUTINE PROCEDES AS FOLLOWS:
                                A) THE PC OF THE INSTRUCTION THAT CALLED THE ERROR HANDLER
                                   IS ACCESSED THRU THE STACK, AND THEN THE EMT INSTRUCTION
                                   ITSELF IS FETCHED.  THE 8 LSB OF THE EMT
                                   INSTRUCTION ARE THE ERROR CODE.  THIS CODE IS
                                   USED TO ACCESS A TABLE OF ERROR MESSAGES AND ERROR
                                   DATA STORAGE LOCATIONS.
                                B) IF THE TEST THAT FAILED DID NOT FAIL PREVIOUSLY
                                   DURING THIS PASS,  A COMPLETE ERROR REPORT IS MADE
                                   IF THE TEST THAT FAILED FAILED MOR THAT ONCE DURING
                                   THE CURRENT PASS, ONLY THE DATA RELATING TO THE FAILUER
                                   IS TYPED.  IF SW13=1, NO ERROR TYPEOUT IS MADE.
                                C) THE ROUTINE NOW CHECKS FOR HALT ON ERROR.  IF SW15=1
                                   THE PROGRAM WILL HALT WITH THE PC OF THE CALL TO
                                   THE ERROR ROUTINE IN R0.  IF SW15=0, THE PROGRAM WILL
                                   NOT HALT, BUT WILL CHECK FOR ESCAPE TO NEXT TEST.
                                D) IF SW10=0, THE ROUTINE WILL RETURN
                                   TO THE TEST IN PROGRESS.  IF SW10=1, THE ROUTINE WILL
                                   ABORT THE CURRENT TEST, AND TRANSFER TO THE NEXT
                                   TEST IN SEQUENCE, THRU THE ROUTINE "SCOPER".

                        5.2.8   TRPSRV  (TRAP DECODE AND DISPATCH)

                                THIS ROUTINE DECODES THE 8 LSB OF THE TRAP INSTRUCTION
                                THAT CAUSED TH PROGRAM INTERRUPT, AND TRANSFERS CONTROL
                                TO THE ROUTINE THRU THE TABLE "TRPTAB" USING THE 8 LSB
                                OF THE TRAP INSTRUCTION AS  AN OFFSET TO THE POINTER TO
                                THE ROUTINE TO BE ENTERED.
```

5.3 PROGRAM AND OR OPERATOR ACTION

5.3.1 PROGRAM START WITH ALL SWITCHES DOWN

5.3.1.1 REFER TO SECTIONS 4.3.1 AND 4.3.2 FOR INITIAL PROGRAM BEHAVIOR.

5.3.1.2 AFTER "R" HAS BEEN TYPED BY THE PROGRAM, TEST EXECUTION WILL BEGIN. EACH TEST WILL BE REPEATED A SELECTED NUMBER OF ITERATIONS (SEE LISTING FOR EXACT NUMBER FOR EACH TEST) AND THEN THE PROGRAM WILL PROCEED TO THE NEXT TEST.

5.3.1.3 WHEN ALL ITERATIONS HAVE BEEN COMPLETED, THE PROGRAM WILL TYPE "CZDHD-D" AND THEN RESTART TESTING AT TEST 1 (LOCATION T1 IN THE PROGRAM).

5.3.1.4 IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE, AND THEN CONTINUE THE TEST IN PROGRESS.

5.3.2 PROGRAM START WITH SW00=1

THE PROGRAM WILL PERFORM AS DESCRIBED IN 4.3.1 AND 5.3.1

5.3.3 PROGRAM START WITH SW01=1

5.3.3.1 REFER TO SECTION 4.3.4 FOR INITIAL PROGRAM BEHAVIOR

5.3.3.2 TEST EXECUTION WILL START AT THE ADDRESS SPECIFIED AND WILL CONTINUE AS DESCRIBED IN 5.3.1.2

5.3.3.3 AFTER "CZDHD-D" HAS BEEN TYPED, THE PROGRAM WILL RESUME TESTING AT TEST 1

5.3.4 PROGRAM OPERATION WITH SW15=1

SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR, THE PROGRAM WILL HALT AFTER THE ERROR TYPEOUT, AND THE PC+2 OF THE CALL TO THE ERROR ROUTINE WILL BE DISPLAYED IN R0.

5.3.5 PROGRAM OPERATION WITH SW13=1

SAME AS 5.3.1 EXCEPT THAT NO ERROR TYPEOUTS WILL OCCUR

5.3.6 PROGRAM OPERATION WITH SW11=1

SAME AS 5.3.1 EXCEPT THAT EACH TEST WILL BE REPEATED ONCE ONLY

5.3.7 PROGRAM OPERATION WITH SW10=1

SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR THE CURRENT TEST WILL BE ABORTED, AND THE PROGRAM WILL PROCEED TO THE NEXT TEST IN SEQUENCE.

```
                5.      (CONT'D)

                5.3.8   PROGRAM OPERATION WITH SW14=1, OR SW09=1

                        THESE FUNCTIONS ARE NORMALLY USED FOR TROUBLE SHOOTING.
                        SEE SECTION 6.3 FOR THEIR USE.

                6.      ERRORS

                6.1     ERROR HALTS

                        THE ERROR MESSAGE FORMAT FOR ALL ERROR TYPEOUTS
                        IS AS FOLLOWS

                        PC+2    MESSAGE
                                HEADER (IF APPLICABLE)
                                DATA    (IF APPLICABLE)

                        WHERE
                        PC+2 IS THE ADDRESS OF THE CALL TO THE ERROR HANDLER + 2
                        MESSAGE IS AN ASCII MESSAGE DESCRIBING (BRIEFLY) THE FAILURE
                        HEADER IS A DESCRIPTION OF THE DATA TO FOLLOW
                        DATA IS OCTAL INFORMATION RELATING TO THE CAUSE OF THE FAILURE
                        IF THE SAME ERROR OCCURS IN A GIVEN TEST ON THE SAME
                        PASS, AND IF DATA IS ASSOCIATED WITH THAT ERROR, ONLY
                        DATA IS TYPE ON SUCCEEDING ERROR TYPEOUTS

                        IF NO DATA IS ASSOCIATED WITH THE ERROR
                        THE COMPLETE ERROR MESSAGE IS TYPED.

                6.1.1   ERROR DESCRIPTIONS

                        SEE LISTING FOR DETAILS OF ERRORS

                6.2     ERROR RECOVERY

                6.2.1   SW15=0

                        IF THE PROGRAM IS RUN WITH SW15=0, NO OPERATOT 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

                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 200 TO RECOVER FROM THIS ERROR.
```

6.3 SCOPE LOOPING

6.3.1 TO SCOPE ON A SPECIFIC TEST, SET SW14=1 AND SW13=1
THIS WILL CAUSE THE PROGRAM TO CONTINUOUSLY LOOP ON THE
SAME TEST, AND WILL CAUSE ALL ERROR TYPEOUTS TO BE INHIBITED

6.3.2 TO SCOPE ON A SPECIFIC VALUE OF A PARAMETER WITHIN
A TEST, SET SW09=1 TO FREEZE THE DATA
(SEE LISTING FOR THOSE TESTS THAT INCORPORATE THIS FEATURE)

6. (CONT'D)

6.3.3 PROGRAM START TO SCOPE LOOP ON SELECTED TEST

PERFORM SECTION 4.3.4 WITH SW14=1

7. RESTRICTIONS

7.1 STARTING

THE DH11 TEST CARD MUST BE INSTALLED

7.2 RUNNING

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIME

THE TIME FOR ONE PASS OF THE PROGRAM (END OF
TYPEOUT OF CZDHD-D TO END OF TYPEOUT OF CZDHD-D)
IS GIVEN FOR VARIOUS PROCESSORS IN THE TABLE BELOW

PROCESSOR	TIME
PDP-11/05,10	
PDP-11/20	
PDP-11/40	
PDP-11/45	

```
                9.      PROGRAM DESCRIPTION

                        TRANSMITTER LINE SPEED SELECTION IS TESTED ON A LINE BY
                        LINE BASIS USING A RELATIVE TIMING TECHNIQUE TO DETERMINE
                        IF SPEED SELECTION FOR A SELECTED LINE IS CORRECT.

                        THE TEST PROCEEDS AS FOLLOWS:

                        A SPEED OF 50 BAUD IS SET FOR A SELECTED LINE, AND
                        A COUNT IS RECORDED FROM THE TIME THAT THE BAR BIT IS
                        SET FOR THAT LINE TO THE TIME THAT TRANSMITTER DONE IS SET.
                        THREE CHARACTERS ARE TRANSMITTED.  AT THE SAME
                        TIME, A TIMEOUT COUNTER IS STARTED.  IF THE TIMEOUT
                        COUNTER DECREMENTS TO 0 BEFORE TRANSMITTER DONE IS
                        RECEIVED, AN ERROR MESSAGE IS REPORTED.  IF THE TIMEOUT
                        DOES NOT OCCUR, THE TIME COUNT IS STORED, AND THE
                        NEXT LINE SPEED IS SELECTED.  TRANSMISSION IS RESTARTED
                        AND THE TIME COUNT, AND TIMEOUT ARE RESTARTED.
                        WHEN TRANSMITTER DONE IS RECEIVED, THE TIME COUNTS
                        FOR THE CURRENT SPEED AND THE PREVIOUS SPEED ARE
                        COMPARED.  IF THE TIME COUNT FOR THE CURRENT SPEED IS
                        GREATER THAN OR EQUAL TO THE COUNT FOR THE PREVIOUS
                        SPEED, A TIMING ERROR HAS OCCURED, SINCE A HIGHER
                        SELECTED BAUD RATE SHOULD MEAN THAT THE NUMBER
                        OF COUNTS RECORDED IS LESS THAN AT A LOWER
                        BAUD RATE. THIS PROCEDURE IS REPEATED FOR ALL SPEED
                        CODES 1-15.

                        THE NEXT GROUP OF TESTS VERIFIES THAT RECEIVER
                        SPEED SELECTION IS CORRECT, BY USING A RELATIVE TIMING
                        COMPARISION AS DESCRIBED ABOVE.  A CHARACTER IS TRANSMITTED
                        AS ABOVE AND THE TIME FROM THE START OF TRANSMISSION
                        TO THE TIME THAT CHARACTER AVAILABLE IS RECEIVED
                        IS RECORDED.  ALSO, THE TIMEOUT COUNT IS DECREMENTED.
                        IF THE TIMEOUT COUNTER DECREMENTS TO 0 BEFORE CHARACTER
                        AVAILABLE OCCURS, AN ERROR HAS OCCURED.
                        THE PROCEDURE IS REPEATED AT THE NEXT HIGHEST BAUD RATE
                        AND A COMPARISION IS MADE AS IN THE TRANSMITTER TESTS
                        IF THE TIME COUNT AT THE PRESENT BAUD RATE IS GREATER THAN
                        OR EQUAL TO THE PREVIOUS BAUD RATE, AN ERROR HAS OCCURED.

               10.      LISTING

                        !
```

```
                        ; DHMAC-A - DH11 MACRO LIBRARY
                        ; COPYRIGHT 1985, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754


                        .LIST   ME
                        .NLIST  MC,MD,CND
```



```
                        ;  CMS REPLACEMENT HISTORY

                        ; *9 SKONETSKI 26-APR-1985 16:23:08 "FIXED TYPO CAUSING ASSEMBLY ERRORS"
                        ; *8 SKONETSKI 22-APR-1985 16:48:03 "TYPO ERROR IN VECTOR CHANGE CODE SOURCE FIXED"
                        ; *7 SKONETSKI 22-APR-1985 16:26:04 "ADDED CODE TO SET VECTORS FOR PWR FAIL, ERRORS, AND EMT
TRAPS."
                        ; *6 SKONETSKI 22-APR-1985 14.22:35 "FIXED BRANCH ERROR IN END OF PASS ROUTINE"
                        ; *5 SKONETSKI 22-APR-1985 08:28:54 "FIXED BUG (AN OCTASC MACRO CALL WAS WRONG) AND ADDED A
CLEAN END OF PASS
MESSAGE."
                        ; *4 SKONETSKI 18-APR-1985 14:20:15 "ADDED SOFTWARE SWITCH REG SUPPORT, PUT UNTESTED"
                        ; *3 SKONETSKI 12-APR-1985 10:34:52 "FIXED PROBLEMS WITH SPURIOUS CR/LFS"
                        ; *2 SKONETSKI 11-APR-1985 16:00:24 "ADDED MACRO FROM SYSMAC.SML THAT SIZES FOR SOFTWARE SWI
TCH REGISTER"
                        ; *1 SKONETSKI 11-APR-1985 15:49:05 "LIBRARY FOR DH11 DIAGNOSTICS"
```

```
                                .LIST   ME
                                .NLIST  MC,MD,CND
000CC0                          .HEADER +/1976,1985/,+/DH11 SPEED SELECTION LOGIC TEST/,+/CZDHD-DO/

                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE DH11 SPEED SELECTION LOGIC TEST
                                        ;PROGRAM WILL TYPE "VECTOR ADDRESS-"
                                        ;TYPE IN THE ADDRESS OF THE RECEIVER INTERRUPT VECTOR
                                        ;FOR THE DH11 TO BE TESTED, FOLLOWED BY <CARRIAGE RETURN>
                                        ;PROGRAM WILL TYPE "CONTROL REGISTER ADDRESS-"
                                        ;TYPE IN THE ADDRESS OF THE SYSTEM CONTROL REGISTER
                                        ;FOR THE DH11 TO BE TESTED, FOLLOWED BY <CARRIAGE RETURN>          ; 3
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE " CZDHD-DO "
                                        ;AND THEN RESUM TESTING

                                .TITLE  CZDHD-DO
000000                          .ENABLE ABS
                                .NLIST  MC,MD,CND
                                .LIST   ME
000000                          .SYMBOLS

                                ;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                   ; 3
                001000          SW09=1000               ;=1,LOOP WITH CURRENT DATA
                000400          SW08=400
                000100          SW06=100
                000040          SW05=40
                000020          SW04=20
                000010          SW03=10
                000004          SW02=4
                000002          SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
                000001          SW00=1                  ;RESELECT VECTOR AND CONTROL REGISTER
                                                        ;ADDRESS AFTER PROGRAM RESTART
```

```
      0

                          ;REGISTER DEFINITIONS

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

                          ;LOCATION EQUIVALENCIES                                  ; 3

                          ;SWR=177570       ;CONSOLE SWITCH REGISTER               ; 4
                          ;LIGHTS=177570    ;PDP-11/45 DISPLAY REGISTER            ; 4
177776                    PS=177776         ;PROCESSOR STATUS WORD
015120                    STACK=ENDCOD+200      ;START OF PROCESSOR STACK          ; 3

                          ;INSTRUCTION DEFINITIONS

005746                    PUSH1SP=5746      ;DECREMENT PROCESSOR STACK 1 WORD
005726                    POP1SP=5726       ;INCREMENT PROCESSOR STACK 1 WORD
010046                    PUSHR0=10046      ;SAVE R0 ON STACK
012600                    POPR0=12600       ;RESTORE R0 FROM STACK
024646                    PUSH2SP=24646     ;DECREMENT STACK TWICE
022626                    POP2SP=22626      ;INCREMENT STACK TWICE
                          ;
                          .MACRO  HLT       $A
                                  EMT       $A
                          .ENDM   HLT
                          ;
                          ;
100000                    BIT15=100000
040000                    BIT14=40000                                              ; 3
020000                    BIT13=20000
010000                    BIT12=10000
004000                    BIT11=4000
002000                    BIT10=2000
001000                    BIT09=1000
000400                    BIT08=400
000200                    BIT07=200
000100                    BIT06=100
000040                    BIT05=40
000020                    BIT04=20
000010                    BIT03=10
000004                    BIT02=4
000002                    BIT01=2
000001                    BIT00=1
  1 000000                .CATCH
```

```
0
                                        ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
                000000          .=0
                000200          .REPT 200
                                        .+2             ;UNEXPECTED TRAP TO THIS LOCATION
                                        HALT            ;EXAMINE STACK TO FIND CAUSE
                                .ENDR
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                          HALT            ;EXAMINE STACK TO FIND CAUSE
000114  000116                          .+2             ;UNEXPECTED TRAP TO THIS LOCATION
000116  000000                          HALT            ;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
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
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
  1 001000              .SETVEC
```

```
0                                       ;STANDARD INTERRUPT VECTORS
          000200                .=200
  000200  000167  000600                JMP     START           ;GO TO START OF PROGRAM

1 000204                        .TRPDEF

                                        ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
                                        ;POINTERS TO SUBROUTINES CAN BE FOUND STARTING
                                        ;AT LOCATION "TRPTAB"

  000204                        TRPDEF  SCOPE,†/SCOPE LOOP AND ITERATION HANDLER/
          104400                        SCOPE=TRAP+Y            ;SCOPE LOOP AND ITERATION HANDLER
          000001                        Y=Y+1
  000204                        TRPDEF  TYPE,†/TELETYPE OUTPUT ROUTINE/
          104401                        TYPE=TRAP+Y             ;TELETYPE OUTPUT ROUTINE
          000002                        Y=Y+1
  000204                        TRPDEF  OCTASC,†/OCTAL TO ASCII CONVERSION/
          104402                        OCTASC=TRAP+Y           ;OCTAL TO ASCII CONVERSION
          000003                        Y=Y+1
  000204                        TRPDEF  INSTR,†/INPUT ASCII STRING/
          104403                        INSTR=TRAP+Y            ;INPUT ASCII STRING
          000004                        Y=Y+1
  000204                        TRPDEF  INSTER,†/STRING INPUT ERROR/
          104404                        INSTER=TRAP+Y           ;STRING INPUT ERROR
          000005                        Y=Y+1
  000204                        TRPDEF  PARAM,†/CONVERT STRING TO OCTAL, CHECK LIMITS/
          104405                        PARAM=TRAP+Y            ;CONVERT STRING TO OCTAL, CHECK LIMITS
          000006                        Y=Y+1
  000204                        TRPDEF  SAV05P,†/SAVE R0-R5, PC/
          104406                        SAV05P=TRAP+Y           ;SAVE R0-R5, PC
          000007                        Y=Y+1
  000204                        TRPDEF  RES05,†/RESTORE R0-R5/
          104407                        RES05=TRAP+Y            ;RESTORE R0-R5
          000010                        Y=Y+1
  000204                        TRPDEF  SCOPE1,†/CHECK FOR FREEZE ON CURRENT DATA/
          104410                        SCOPE1=TRAP+Y           ;CHECK FOR FREEZE ON CURRENT DATA
          000011                        Y=Y+1
2                               .MACRO  CODEM1
3                                       MOV     DHSSR,DHSLR             ;SET UP ADDRESS OF SILO
4                                       INC     DHSLR                   ;STATUS REGISTER HIGH BYTE
5                               .ENDM   CODEM1
6 000204                        .START  DHRVEC,3,4,DHSCR,0,177776,7,10,,,1
```

```
      0         001000                  .=1000

                                        ;PROGRAM INITIALIZATION
                                        ;LOCK OUT INTERRUPTS
                                        ;SET UP PROCESSOR STACK
                                        ;SET UP POWER FAIL VECTOR
                                        ;CLEAR PROGRAM FLAGS AND COUNTS
                                        ;TYPE TITLE MESSAGE
                                .IIF NB <>,     ;DETERMINE MEMORY SIZE
                                .IIF NB <>,     ;SET UP TRACE TRAP RETURN

     001000  177570             SWR:    .WORD   177570          ; SWITCH DHSCR ADDRESS                  ; 4
     001002  177570             LIGHTS: .WORD   177570          ; LIGHTS                                ; 4
                                                                                                        ; 4
     001004  012767  000340  176764  START:  MOV     #340,PS         ;LOCK OUT INTERRUPTS
     001012  012706  015120          MOV     #STACK,SP       ;SET UP PROCESSOR STACK
     001016  012702  000024          MOV     #24,R2          ; POINT TO VECTOR AREA                  ; 7
     001022  012722  014070          MOV     #PFAIL,(R2)+    ;SET UP POWER FAIL TRAP                 ; 7
     001026  012722  000340          MOV     #340, (R2)+     ;SERVICE AT LEVEL 7                     ; 7
     001032  012722  012716          MOV     #ERRORS, (R2)+  ;ERROR HANDLER                          ; 7
     001036  012722  000340          MOV     #340, (R2)+     ;SERVICE AT LEVEL 7                     ; 7
     001042  012722  013130          MOV     #TRPSRV, (R2)+  ;GENERAL HANDLER DISPATCH SERVICE       ; 7
     001046  012712  000340          MOV     #340, (R2)      ;SERVICE AT LEVEL 7                     ; 8
     001052  005067  012774          CLR     STFLG           ;CLEAR TEST START FLAG
     001056  005067  012730          CLR     PASCNT          ;CLEAR PASS COUNT
     001062  005067  012726          CLR     ERRCNT          ;CLEAR ERROR COUNT
     001066  005067  012716          CLR     ERRFLG          ;CLEAR ERROR FLAG
     001072  005067  012712          CLR     ERRFLG          ;CLEAR LAST ERROR PC
     001076  016746  176702          MOV     4, -(SP)        ; PUSH TRAP VECTOR                      ; 4
     001102  016746  176700          MOV     6, -(SP)                                                ; 4
     001106  012767  001122  176670  MOV     #1$, 4          ; SET UP TRAP VECTOR                    ; 4
     001114  005777  177660          TST     @SWR            ; TEST SWITCH REGISTER ADDRESS          ; 4
     001120  000405                  BR      2$              ; IF SUCCESSFUL, LEAVE IT ALONE         ; 4
     001122                  1$:                                                                     ; 4
     001122  012767  000176  177650  MOV     #176, SWR       ; POINT TO SOFT SWITCH DHSCR            ; 4
     001130  005067  177646          CLR     LIGHTS          ; 0 MEANS WE ARE NOT GOING TO USE LIGHTS ; 4
     001134                  2$:                                                                     ; 5
     001134  005726                  TST     (SP)+           ; CLEAN UP STACK                        ; 4
     001136  005726                  TST     (SP)+                                                   ; 4
     001140  012667  176642          MOV     (SP)+, 6                                                ; 4
     001144  012667  176634          MOV     (SP)+, 4                                                ; 4
     001150  104401  014240          TYPE    ,MTITLE         ;TYPE TITLE MESSAGE
     001154  005767  012670          TST     INIFLG          ;CHECK INITIALIZATION FLAG
                                .IF NB  <DHRVEC>
     001160  001001                  BNE     VEC1            ;IF NOT 0, CHECK SWITCHES
                                                             ;FOR REINITIALIZATION
                                .IFF
                                        BNE     BEGIN           ;IF NOT 0, START TEST
                                .ENDC
                                .IF NB  <>
                                SIZE:   CLR     R0
                                        MOV     #2$,@#4         ;SET UP TIME OUT RETURN
                                1$:     TST     (R0)+           ;WILL TRAP WHEN NO MEMORY               ; 9
                                        BR      1$              ;LOCATION RESPONDED, CONTINUE
                                2$:     MOV     R0,HCORE        ;R0 CONTAINS ADDRESS OF
                                        SUB     #2,HCORE        ;NON EXISTANT MEMORY                    ; 9
                                        MOV     #6,@#4          ;RESTORE TRAPCATCHER
```

```
                                .ENDC
                                .IF NB  <>
                        TRACER: MOV     #1$,@#10                ;SET UP ILLEGAL INSTRUCTION TRAP RETURN
                                SXT     R0                      ;DO 11/40, 11/45 INSTRUCTION
                                MOV     #RTT,TRTRET             ;11/40,45 RTT RETURN FROM TRACE TRAP
                                BR      2$
                        1$:     MOV     #RTI,TRTRET             ;1105,10,20 RTI RETURN FROM TRACE TRAP
                                MOV     #12,@#10                ;RESTROE TRAPCATCHER
                                MOV     #TRTRET,@#16            ;SET UP TRACE TRAP VECTOR
                                .ENDC
                                .IF NB  <DHRVEC>                                                        ; 3
                                .IF B   <>
001162  000404                  BR      VEC2
                                .IFF
                                TST     INIFLG                  ;IF INITIALIZE FLAG=0
                                BEQ     VEC2                    ;GET VECTOR AND CSR ADDRESS
                                .ENDC
001164  032777  000001  177606  VEC1:   BIT     #SW00,@SWR              ;IF SW00=1, GET NEW VECTOR      ; 4
001172  001445                  BEQ     BEGIN                   ;AND CSR
001174                          VEC2:                                                                   ; 4
001174  012701  000300          MOV     #300,R1                                                         ; 4
001200  012702  000302          MOV     #302,R2
001204  012703  000004          MOV     #4,R3
001210  010211                  1$:     MOV     R2,(R1)                 ;RESTORE TRAPCATCHER
001212  005012                  CLR     (R2)                    ;IN FLOATING VECTOR AREA
001214  060301                  ADD     R3,R1
001216  060302                  ADD     R3,R2
001220  020127  001000          CMP     R1,#1000
001224  001371                  BNE     1$
001226  104403                  INSTR                           ;INPUT ADDRESS OF DEVICE VECTOR
001230  014306                  MVECTOR                         ;MESSAGE "VECTOR ADDRESS-"
001232  104405                  PARAM                           ;CONVERT STRING TO OCTAL
001234  000300                  300                             ;LOW LIMIT
001236  000770                  770                             ;HIGH  LIMIT                            ; 3
001240  014000                  DHRVEC                          ;LOCATIONS TO BE FILLED
001242     003          .BYTE   3                               ;NUMBER OF LOCATIONS
001243     004          .BYTE   4                               ;LSB MASK
001244  104403                  INSTR                           ;INPUT ADDRESS OF DEVICE CSR
001246  014330                  MREGAD                          ;MESSAGE "CONTROL REGISTER ADDRESS-"
001250  104405                  PARAM                           ;CONVERT STRING TO OCTAL
001252  000000                  0                               ;LOW LIMIT
001254  177776                  177776                          ;HIGH LIMIT
001256  013756                  DHSCR                           ;LOCATIONS TO BE FILLED
001260     007          .BYTE   7                               ;NUMBER OF LOCATIONS
001261     010          .BYTE   10                              ;LSB MASK
                                .ENDC
                                .IF NB  <1>
001262                          CODEM1
001262  016767  012506  012506          MOV     DHSSR,DHSLR             ;SET UP ADDRESS OF SILO
001270  005267  012502                  INC     DHSLR                   ;STATUS REGISTER HIGH BYTE
                                .ENDC
001274  005767  012550                  TST     INIFLG                  ;IF INITIALIZATION FLAG
001300  001002                          BNE     BEGIN                   ;IS CLEARED
001302  005167  012542                  COM     INIFLG                  ;SET IT

                                        ;PROGRAM START                                                  ; 3
                                        ;CHECK FOR PROGRAM START AT SELECTED ADDRESS
```

```
001306  012767  000340  176462  BEGIN:  MOV     #340,PS         ;LOCK OUT INTERRUPTS
001314  012706  015120                  MOV     #STACK,SP       ;SET UP PROCESSOR STACK
001320  032777  000002  177452          BIT     #SW01,@SWR      ;IF SW01=1                      ; 4
001326  001410                          BEQ     1$              ;GET PC FOR PROGRAM START
001330  104403                          INSTR                   ;GET PC
001332  014517                          MTSTPC                  ;MESSAGE "TEST PC"
001334  104405                          PARAM                   ;CONVERT STRING TO OCTAL
001336  000000                          0
001340  017500                          17500
001342  014016                          RETRN
001344     001                  .BYTE   1
001345     001                  .BYTE   1
001346  000410                          BR      2$
001350  012767  001400  012440  1$:     MOV     #T1,RETRN       ;NORMAL START, TEST 1
001356  005767  012470                  TST     STFLG           ;IF LOOPING, BYPASS TYPEOUT
001362  001004                          BNE     3$
001364  005167  012462                  COM     STFLG
001370  104401  014513          2$:     TYPE    ,MR             ;TYPE "R" TO INDICATE START
001374  000177  012416          3$:     JMP     @RETRN          ;START TESTING                  ; 3
```

```
                        XLINE=LINE
                        LINE=0
                        XBIT=BITX
                        BITX=1
                        .REPT   20
                        SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/
                        .NLIST
                        LINE=LINE+1
                        BITX=BITX+BITX
                        .LIST
                        .ENDR
001400                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                ;TRANSMITTER LINE SPEED SELECTION TEST
                                ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 0
                                ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

001400                  TS \XN,10,5$,1$
001400  012767  000340  176370 T1:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
001406  012767  000010  012410         MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
001414  012767  001620  012376         MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
001422  012767  001456  012372         MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000002          XN=XN+1
001430  012705  000000         MOV     #0,R5           ;LINE 0 WILL BE TESTED
001434  012700  002000         MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                ;OF INITIAL (LOWEST) SPEED
001440  012701  000015         MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
001444  012704  000001         MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
001450  012767  177777  012402 MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
001456  012777  004000  012272 1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
001464  010577  012266         MOV     R5,@DHSCR               ;SELECT LINE 0 FOR TESTING
001470  005077  012270         CLR     @DHBA                   ;CLEAR BUS ADDRESS
001474  012777  177775  012264 MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                ;3 CHARACTERS
001502  010077  012254         MOV     R0,@DHLPR               ;SELECT LINE SPEED
001506  005067  012350         CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
001512  005067  012346         CLR     TEMP1                   ;SET UP NO CLOCK TIMER
001516  012767  000010  012342 MOV     #10,TEMP2
001524  012777  000001  012236 MOV     #1,@DHBAR               ;SET BAR BIT FOR LINE 0
                                                                ;TO START TRANSMISSION
001532  005777  012220         2$:     TST     @DHSCR          ;WAIT FOR TRANSMITTER
                                                                ;TO FINISH
001536  100412                 BMI     3$
001540  005267  012316         INC     TIME2                   ;UPDATE TRANSMITTER TIMER
001544  005267  012314         INC     TEMP1                   ;UPDATE NO CLOCK TIMER
001550  001370                 BNE     2$
001552  005367  012310         DEC     TEMP2
001556  001365                 BNE     2$
001560                         HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
001560  104001                 EMT     1
001562  000405                 BR      4$
001564  026767  012272  012266 3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
001572  103401                 BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
```

```
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
001574                          HLT     2                               ;TRANSMITTER TIMING ERROR FOR
001574  104002                  EMT     2
                                                                        ;LINE 0
001576  104410          4$:     SCOPE1                                  ;CHECK FOR FREEZE ON CURRENT DATA
001600  016767  012256  012252        MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
001606  005204                  INC     R4                              ;SELECT NEXT SPEED
001610  062700  002000          ADD     #2000,R0
001614  005301                  DEC     R1
001616  001317                  BNE     1$
001620  104400          5$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
        000001          LINE=LINE+1
        000002          BITX=BITX+BITX
001622                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                        ;TRANSMITTER LINE SPEED SELECTION TEST
                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 1
                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

001622                  TS \XN,10,5$,1$
001622  012767  000340  176146  T2:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
001630  012767  000010  012166          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
001636  012767  002042  012154          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
001644  012767  001700  012150          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000003          XN=XN+1
001652  012705  000001          MOV     #1,R5           ;LINE 1 WILL BE TESTED
001656  012700  002000          MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                ;OF INITIAL (LOWEST) SPEED
001662  012701  000015          MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
001666  012704  000001          MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
001672  012767  177777  012160  MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
001700  012777  004000  012050  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
001706  010577  012044          MOV     R5,@DHSCR               ;SELECT LINE 1 FOR TESTING
001712  005077  012046          CLR     @DHBA                   ;CLEAR BUS ADDRESS
001716  012777  177775  012042  MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                ;3 CHARACTERS
001724  010077  012032          MOV     R0,@DHLPR               ;SELECT LINE SPEED
001730  005067  012126          CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
001734  005067  012124          CLR     TEMP1                   ;SET UP NO CLOCK TIMER
001740  012767  000010  012120  MOV     #10,TEMP2
001746  012777  000002  012014  MOV     #2,@DHBAR               ;SET BAR BIT FOR LINE 1
                                                                ;TO START TRANSMISSION
001754  005777  011776  2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                ;TO FINISH
001760  100412                  BMI     3$
001762  005267  012074          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
001766  005267  012072          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
001772  001370                  BNE     2$
001774  005367  012066          DEC     TEMP2
002000  001365                  BNE     2$
002002                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
002002  104001                  EMT     1
```

```
002004  000405                          BR      4$
002006  026767  012050  012044  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
002014  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
002016                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
002016  104002                          EMT     2
                                                                        ;LINE 1
002020  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
002022  016767  012034  012030          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
002030  005204                          INC     R4                      ;SELECT NEXT SPEED
002032  062700  002000                  ADD     #2000,R0
002036  005301                          DEC     R1
002040  001317                          BNE     1$
002042  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000002                  LINE=LINE+1
        000004                  BITX=BITX+BITX
002044                          SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 2
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

002044                          TS \XN,10,5$,1$
002044  012767  000340  175724  T3:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002052  012767  000010  011744          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002060  012767  002264  011732          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
002066  012767  002122  011726          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000004          XN=XN+1
002074  012705  000002                  MOV     #2,R5           ;LINE 2 WILL BE TESTED
002100  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
002104  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
002110  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
002114  012767  177777  011736          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
002122  012777  004000  011626  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
002130  010577  011622                  MOV     R5,@DHSCR               ;SELECT LINE 2 FOR TESTING
002134  005077  011624                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
002140  012777  177775  011620          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
002146  010077  011610                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
002152  005067  011704                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
002156  005067  011702                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
002162  012767  000010  011676          MOV     #10,TEMP2
002170  012777  000004  011572          MOV     #4,@DHBAR               ;SET BAR BIT FOR LINE 2
                                                                        ;TO START TRANSMISSION
002176  005777  011554          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
002202  100412                          BMI     3$
002204  005267  011652                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
002210  005267  011650                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
002214  001370                          BNE     2$
002216  005367  011644                  DEC     TEMP2
```

```
002222  001365                          BNE     2$
002224                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
002224  104001                          EMT     1
002226  000405                          BR      4$
002230  026767  011626  011622  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
002236  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
002240                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
002240  104002                          EMT     2
                                                                        ;LINE 2
002242  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
002244  016767  011612  011606          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
002252  005204                          INC     R4                      ;SELECT NEXT SPEED
002254  062700  002000                  ADD     #2000,R0
002260  005301                          DEC     R1
002262  001317                          BNE     1$
002264  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000003                  LINE=LINE+1
        000010                  BITX=BITX+BITX
002266                          SPEED   \LINE,\BITX,3,^//,2000,^/TRANSMITTER/

                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 3
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

002266                          TS \XN,10,5$,1$
002266  012767  000340  175502  T4:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002274  012767  000010  011522          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002302  012767  002506  011510          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
002310  012767  002344  011504          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000005          XN=XN+1
002316  012705  000003                  MOV     #3,R5           ;LINE 3 WILL BE TESTED
002322  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
002326  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
002332  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
002336  012767  177777  011514          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
002344  012777  004000  011404  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
002352  010577  011400                  MOV     R5,@DHSCR               ;SELECT LINE 3 FOR TESTING
002356  005077  011402                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
002362  012777  177775  011376          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
002370  010077  011366                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
002374  005067  011462                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
002400  005067  011460                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
002404  012767  000010  011454          MOV     #10,TEMP2
002412  012777  000010  011350          MOV     #10,@DHBAR              ;SET BAR BIT FOR LINE 3
                                                                        ;TO START TRANSMISSION
002420  005777  011332          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
002424  100412                          BMI     3$
002426  005267  011430                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
```

```
002432  005267  011426                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
002436  001370                          BNE     2$
002440  005367  011422                  DEC     TEMP2
002444  001365                          BNE     2$
002446                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
002446  104001                          EMT     1
002450  000405                          BR      4$
002452  026767  011404  011400  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
002460  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
002462                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
002462  104002                          EMT     2
                                                                        ,LINE 3
002464  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
002466  016767  011370  011364          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
002474  005204                          INC     R4                      ;SELECT NEXT SPEED
002476  062700  002000                  ADD     #2000,R0
002502  005301                          DEC     R1
002504  001317                          BNE     1$
002506  104400                  5$:     SCOPE                           :CHECK FOR ITERATIONS, LOOP
        000004                  LINE=LINE+1
        000020                  BITX=BITX+BITX
0C2510                          SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 4
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

002510                          TS \XN,10,5$,1$
002510  012767  000340  175260  T5:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002516  012767  000010  011300          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002524  012767  002730  011266          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
002532  012767  002566  011262          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000006          XN=XN+1
002540  012705  000004                  MOV     #4,R5           ;LINE 4 WILL BE TESTED
002544  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SE'ECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
002550  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
002554  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
002560  012767  177777  011272          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
002566  012777  004000  011162  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
002574  010577  011156                  MOV     R5,@DHSCR               ;SELECT LINE 4 FOR TESTING
002600  005077  011160                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
002604  012777  177775  011154          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
002612  010077  011144                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
002616  005067  011240                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
002622  005067  011236                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
002626  012767  000010  011232          MOV     #10,TEMP2
002634  012777  000020  011126          MOV     #20,@DHBAR              ;SET BAR BIT FOR LINE 4
                                                                        ;TO START TRANSMISSION
002642  005777  011110          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
```

```
                                                                        ;TO FINISH
002646  100412                          BMI     3$
002650  005267  011206                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
002654  005267  011204                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
002660  001370                          BNE     2$
002662  005367  011200                  DEC     TEMP2
002666  001365                          BNE     2$
002670                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
002670  104001                          EMT     1
002672  000405                          BR      4$
002674  026767  011162  011156  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
002702  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
002704                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
002704  104002                          EMT     2
                                                                        ;LINE 4
002706  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
002710  016767  011146  011142          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
002716  005204                          INC     R4                      ;SELECT NEXT SPEED
002720  062700  002000                  ADD     #2000,R0
002724  005301                          DEC     R1
002726  001317                          BNE     1$
002730  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000005                  LINE=LINE+1
        000040                  BITX=BITX+BITX
002732                          SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 5
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

002732                          TS \XN,10,5$,1$
002732  012767  000340  175036  T6:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002740  012767  000010  011056          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002746  012767  003152  011044          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
002754  012767  003010  011040          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000007          XN=XN+1
002762  012705  000005                  MOV     #5,R5           ;LINE 5 WILL BE TESTED
002766  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
002772  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
002776  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
003002  012767  177777  011050          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
003010  012777  004000  010740  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
003016  010577  010734                  MOV     R5,@DHSCR               ;SELECT LINE 5 FOR TESTING
003022  005077  010736                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
003026  012777  177775  010732          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
003034  010077  010722                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
003040  005067  011016                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
003044  005067  011014                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
003050  012767  000010  011010          MOV     #10,TEMP2
```

```
003056  012777  000040  010704          MOV     #40,@DHBAR              ;SET BAR BIT FOR LINE 5
                                                                        ;TO START TRANSMISSION
003064  005777  010666          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
003070  100412                          BMI     3$
003072  005267  010764                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
003076  005267  010762                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
003102  001370                          BNE     2$
003104  005367  010756                  DEC     TEMP2
003110  001365                          BNE     2$
003112                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
003112  104001                          EMT     1
003114  000405                          BR      4$
003116  026767  010740  010734  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
003124  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
003126                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
003126  104002                          EMT     2
                                                                        ;LINE 5
003130  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
003132  016767  010724  010720          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
003140  005204                          INC     R4                      ;SELECT NEXT SPEED
003142  062700  002000                  ADD     #2000,R0
003146  005301                          DEC     R1
003150  001317                          BNE     1$
003152  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000006          LINE=LINE+1
        000100          BITX=BITX+BITX
003154                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                ;TRANSMITTER LINE SPEED SELECTION TEST
                                ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 6
                                ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

003154                  TS \XN,10,5$,1$
003154  012767  000340  174614  T7:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003162  012767  000010  010634          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003170  012767  003374  010622          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
003176  012767  003232  010616          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000010          XN=XN+1
003204  012705  000006                  MOV     #6,R5           ;LINE 6 WILL BE TESTED
003210  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
003214  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
003220  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
003224  012767  177777  010626          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
003232  012777  004000  010516  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
003240  010577  010512                  MOV     R5,@DHSCR               ;SELECT LINE 6 FOR TESTING
003244  005077  010514                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
003250  012777  177775  010510          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
003256  010077  010500                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
```

```
003262  005067  010574                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
003266  005067  010572                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
003272  012767  000010  010566          MOV     #10,TEMP2
003300  012777  000100  010462          MOV     #100,@DHBAR             ;SET BAR BIT FOR LINE 6
                                                                        ;TO START TRANSMISSION
003306  005777  010444          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
003312  100412                          BMI     3$
003314  005267  010542                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
003320  005267  010540                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
003324  001370                          BNE     2$
003326  005367  010534                  DEC     TEMP2
003332  001365                          BNE     2$
003334                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
003334  104001                          EMT     1
003336  000405                          BR      4$
003340  026767  010516  010512  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
003346  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
003350                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
003350  104002                          EMT     2
                                                                        ;LINE 6
0C3352  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
003354  016767  010502  010476          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
003362  005204                          INC     R4                      ;SELECT NEXT SPEED
003364  062700  002000                  ADD     #2000,R0
003370  005301                          DEC     R1
003372  001317                          BNE     1$
003374  10440C                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000007                  LINE=LINE+1
        000200                  BITX=BITX+BITX
003376                          SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 7
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

003376                          TS \XN,10,5$,1$
003376  012767  000340  174372  T10:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003404  012767  000010  010412          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003412  012767  003616  010400          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
003420  012767  003454  010374          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000011          XN=XN+1
003426  012705  000007                  MOV     #7,R5           ;LINE 7 WILL BE TESTED
003432  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
003436  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
003442  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
003446  012767  177777  010404          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
003454  012777  004000  010274  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
003462  010577  010270                  MOV     R5,@DHSCR               ;SELECT LINE 7 FOR TESTING
003466  005077  010272                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
```

```
003472  012777  177775  010266          MOV     #-3,@DHBC           ;SET UP TO TRANSMIT
                                                                    ;3 CHARACTERS
003500  010077  010256                  MOV     R0,@DHLPR           ;SELECT LINE SPEED
003504  005067  010352                  CLR     TIME2               ;CLEAR TRANSMITTER TIME TIMER
003510  005067  010350                  CLR     TEMP1               ;CLEAR NO CLOCK TIMER
003514  012767  000010  010344          MOV     #10,TEMP2
003522  012777  000200  010240          MOV     #200,@DHBAR         ;SET BAR BIT FOR LINE 7
                                                                    ;TO START TRANSMISSION
003530  005777  010222          2$:     TST     @DHSCR              ;WAIT FOR TRANSMITTER
                                                                    ;TO FINISH
003534  100412                          BMI     3$
003536  005267  010320                  INC     TIME2               ;UPDATE TRANSMITTER TIMER
003542  005267  010316                  INC     TEMP1               ;UPDATE NO CLOCK TIMER
003546  001370                          BNE     2$
003550  005367  010312                  DEC     TEMP2
003554  001365                          BNE     2$
003556                                  HLT     1                   ;TRANSMITTER DID NOT FINISH, ERROR
003556  104001                          EMT     1
003560  000405                          BR      4$
003562  026767  010274  010270  3$:     CMP     TIME2,TIME1         ;VERIFY THAT TRANSMITTER
003570  103401                          BLO     4$                  ;WAS FASTER AT THIS SELECTED SPEED
                                                                    ;(NUMBER OF COUNTS IN TIME2
                                                                    ;LESS THAN TIME1)
003572                                  HLT     2                   ;TRANSMITTER TIMING ERROR FOR
003572  104002                          EMT     2
                                                                    ;LINE 7
003574  104410          4$:             SCOPE1                      ;CHECK FOR FREEZE ON CURRENT DATA
003576  016767  010260  010254          MOV     TIME2,TIME1         ;SET UP FOR NEXT COMPARISION
003604  005204                          INC     R4                  ;SELECT NEXT SPEED
003606  062700  002000                  ADD     #2000,R0
003612  005301                          DEC     R1
003614  001317                          BNE     1$
003616  104400          5$:             SCOPE                       ;CHECK FOR ITERATIONS, LOOP
        000010          LINE=LINE+1
        000400          BITX=BITX+BITX
003620                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                        ;TRANSMITTER LINE SPEED SELECTION TEST
                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 10
                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

003620                  TS \XN,10,5$,1$
003620  012767  000340  174150  T11:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003626  012767  000010  010170          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003634  012767  004040  010156          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
003642  012767  003676  010152          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000012          XN=XN+1
003650  012705  000010                  MOV     #10,R5          ;LINE 10 WILL BE TESTED
003654  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
003660  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
003664  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
003670  012767  177777  010162          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
```

```
003676  012777  004000  010052  1$:     MOV     @#BIT11,@DHSCR          ;CLEAR INTERFACE
003704  010577  010046                  MOV     R5,@DHSCR               ;SELECT LINE 10 FOR TESTING
003710  005077  010050                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
003714  012777  177775  010044          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
003722  010077  010034                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
003726  005067  010130                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
003732  005067  010126                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
003736  012767  000010  010122          MOV     #10,TEMP2
003744  012777  000400  010016          MOV     #400,@DHBAR             ;SET BAR BIT FOR LINE 10
                                                                        ;TO START TRANSMISSION
003752  005777  010000          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
003756  100412                          BMI     3$
003760  005267  010076                  INC     TIME2                   ;UPDATE TRANSMITTER TIMER
003764  005267  010074                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
003770  001370                          BNE     2$
003772  005367  010070                  DEC     TEMP2
003776  001365                          BNE     2$
004000                                  HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
004000  104001                          EMT     1
004002  000405                          BR      4$
004004  026767  010052  010046  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
004012  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
004014                                  HLT     2                       ;TRANSMITTER TIMING ERROR FOR
004014  104002                          EMT     2
                                                                        ;LINE 10
004016  104410          4$:     SCOPE1                                  ;CHECK FOR FREEZE ON CURRENT DATA
004020  016767  010036  010032          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
004026  005204                          INC     R4                      ;SELECT NEXT SPEED
004030  062700  002000                  ADD     #2000,R0
004034  005301                          DEC     R1
004036  001317                          BNE     1$
004040  104400          5$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
        000011  LINE=LINE+1
        001000  BITX=BITX+BITX
004042          SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                        ;TRANSMITTER LINE SPEED SELECTION TEST
                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 11
                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

004042          TS \XN,10,5$,1$
004042  012767  000340  173726  T12:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004050  012767  000010  007746          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004056  012767  004262  007734          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
004064  012767  004120  007730          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000013  XN=XN+1
004072  012705  000011                  MOV     #11,R5          ;LINE 11 WILL BE TESTED
004076  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
```

```
004102  012701  000015                  MOV     #15,R1          ;15 DIFFERENT SPEEDS WILL BE TESTED
004106  012704  000001                  MOV     #1,R4           ;BINARY CODE FOR INITIAL SPEED
004112  012767  177777  007740          MOV     #-1,TIME1       ;INITIALIZE COMPARISION VALUE
004120  012777  004000  007630  1$:     MOV     #BIT11,@DHSCR   ;CLEAR INTERFACE
004126  010577  007624                  MOV     R5,@DHSCR       ;SELECT LINE 11 FOR TESTING
004132  005077  007626                  CLR     @DHBA           ;CLEAR BUS ADDRESS
004136  012777  177775  007622          MOV     #-3,@DHBC       ;SET UP TO TRANSMIT
                                                                ;3 CHARACTERS
004144  010077  007612                  MOV     R0,@DHLPR       ;SELECT LINE SPEED
004150  005067  007706                  CLR     TIME2           ;CLEAR TRANSMITTER TIME TIMER
004154  005067  007704                  CLR     TEMP1           ;SET UP NO CLOCK TIMER
004160  012767  000010  007700          MOV     #10,TEMP2
004166  012777  001000  007574          MOV     #1000,@DHBAR    ;SET BAR BIT FOR LINE 11
                                                                ;TO START TRANSMISSION
004174  005777  007556          2$:     TST     @DHSCR          ;WAIT FOR TRANSMITTER
                                                                ;TO FINISH
004200  100412                          BMI     3$
004202  005267  007654                  INC     TIME2           ;UPDATE TRANSMITTER TIMER
004206  005267  007652                  INC     TEMP1           ;UPDATE NO CLOCK TIMER
004212  001370                          BNE     2$
004214  005367  007646                  DEC     TEMP2
004220  001365                          BNE     2$
004222                                  HLT     1               ;TRANSMITTER DID NOT FINISH, ERROR
004222  104001                          EMT     1
004224  000405                          BR      4$
004226  026767  007630  007624  3$:     CMP     TIME2,TIME1     ;VERIFY THAT TRANSMITTER
004234  103401                          BLO     4$              ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
004236                                  HLT     2               ;TRANSMITTER TIMING ERROR FOR
004236  104002                          EMT     2
                                                                ;LINE 11
004240  104410                  4$:     SCOPE1                  ;CHECK FOR FREEZE ON CURRENT DATA
004242  016767  007614  007610          MOV     TIME2,TIME1     ;SET UP FOR  _XT COMPARISION
004250  005204                          INC     R4              ;SELECT NEXT SPEED
004252  062700  002000                  ADD     #2000,R0
004256  005301                          DEC     R1
004260  001317                          BNE     1$
004262  104400                  5$:     SCOPE                   ;CHECK FOR ITERATIONS, LOOP
        000012          LINE=LINE+1
        002000          BITX=BITX/BITX
004264                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                ;TRANSMITTER LINE SPEED SELECTION TEST
                                ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 12
                                ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

004264                  TS \XN,10,5$,1$
004264  012767  000340  173504  T13:    MOV     #340,PS         ;DISABLE ALL INTERRUPTS
004272  012767  000010  007524          MOV     #10,ICOUNT      ;SET UP FOR 10 ITERATIONS
004300  012767  004504  007512          MOV     #5$,ESCAPE      ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
004306  012767  004342  007506          MOV     #1$,FREEZ1      ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000014          XN=XN+1
```

```
004314  012705  000012                  MOV     #12,R5          ;LINE 12 WILL BE TESTED
004320  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
004324  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
004330  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
004334  012767  177777  007516          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
004342  012777  004000  007406  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
004350  010577  007402                  MOV     R5,@DHSCR               ;SELECT LINE 12 FOR TESTING
004354  005077  007404                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
004360  012777  177775  007400          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
004366  010077  007370                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
004372  005067  007464                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
004376  005067  007462                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
004402  012767  000010  007456          MOV     #10,TEMP2
004410  012777  002000  007352          MOV     #2000,@DHBAR            ;SET BAR BIT FOR LINE 12
                                                                        ;TO START TRANSMISSION
004416  005777  007334          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
004422  100412                  BMI     3$
004424  005267  007432          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
004430  005267  007430          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
004434  001370                  BNE     2$
004436  005367  007424          DEC     TEMP2
004442  001365                  BNE     2$
004444                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
004444  104001                  EMT     1
004446  000405                  BR      4$
004450  026767  007406  007402  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
004456  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
004460                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
004460  104002                  EMT     2
                                                                ;LINE 12
004462  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
004464  016767  007372  007366          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
004472  005204                  INC     R4                      ;SELECT NEXT SPEED
004474  062700  002000                  ADD     #2000,R0
004500  005301                  DEC     R1
004502  001317                  BNE     1$
004504  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000013          LINE=LINE+1
        004000          BITX=BITX+BITX
004506                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                ;TRANSMITTER LINE SPEED SELECTION TEST
                                ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 13
                                ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

004506                  TS \XN,10,5$,1$
004506  012767  000340  173262  T14:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004514  012767  000010  007302          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004522  012767  004726  007270          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
```

```
004530  012767  004564  007264          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000015                  XN=XN+1
004536  012705  000013          MOV     #13,R5          ;LINE 13 WILL BE TESTED
004542  012700  002000          MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                ;OF INITIAL (LOWEST) SPEED
004546  012701  000015          MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
004552  012704  000001          MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
004556  012767  177777  007274  MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
004564  012777  004000  007164  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
004572  010577  007160          MOV     R5,@DHSCR               ;SELECT LINE 13 FOR TESTING
004576  005077  007162          CLR     @DHBA                   ;CLEAR BUS ADDRESS
004602  012777  177775  007156  MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                ;3 CHARACTERS
004610  010077  007146          MOV     R0,@DHLPR               ;SELECT LINE SPEED
004614  005067  007242          CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
004620  005067  007240          CLR     TEMP1                   ;SET UP NO CLOCK TIMER
004624  012767  000010  007234  MOV     #10,TEMP2
004632  012777  004000  007130  MOV     #4000,@DHBAR            ;SET BAR BIT FOR LINE 13
                                                                ;TO START TRANSMISSION
004640  005777  007112          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                ;TO FINISH
004644  100412                  BMI     3$
004646  005267  007210          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
004652  005267  007206          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
004656  001370                  BNE     2$
004660  005367  007202          DEC     TEMP2
004664  001365                  BNE     2$
004666                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
004666  104001                  EMT     1
004670  000405                  BR      4$
004672  026767  007164  007160  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
004700  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
004702                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
004702  104002                  EMT     2
                                                                ;LINE 13
004704  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
004706  016767  007150  007144  MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
004714  005204                  INC     R4                      ;SELECT NEXT SPEED
004716  062700  002000          ADD     #2000,R0
004722  005301                  DEC     R1
004724  001317                  BNE     1$
004726  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000014          LINE=LINE+1
        010000          BITX=BITX+BITX
004730                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                                ;TRANSMITTER LINE SPEED SELECTION TEST
                                ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 14
                                ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

004730                  TS \XN,10,5$,1$
004730  012767  000340  173040  T15:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
```

```
004736  012767  000010  007060          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004744  012767  005150  007046          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
004752  012767  005006  007042          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000016                  XN=XN+1
004760  012705  000014                  MOV     #14,R5          ;LINE 14 WILL BE TESTED
004764  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
004770  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
004774  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
005000  012767  177777  007052          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
005006  012777  004000  006742  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
005014  010577  006736                  MOV     R5,@DHSCR               ;SELECT LINE 14 FOR TESTING
005020  005077  006740                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
005024  012777  177775  006734          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
005032  010077  006724                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
005036  005067  007020                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
005042  005067  007016                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
005046  012767  000010  007012          MOV     #10,TEMP2
005054  012777  010000  006706          MOV     #10000,@DHBAR           ;SET BAR BIT FOR LINE 14
                                                                        ;TO START TRANSMISSION
005062  005777  006670          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
005066  100412                  BMI     3$
005070  005267  006766          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
005074  005267  006764          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
005100  001370                  BNE     2$
005102  005367  006760          DEC     TEMP2
005106  001365                  BNE     2$
005110                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
005110  104001                  EMT     1
005112  000405                  BR      4$
005114  026767  006742  006736  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
005122  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
005124                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
005124  104002                  EMT     2
                                                                ;LINE 14
005126  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
005130  016767  006726  006722          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
005136  005204                  INC     R4                      ;SELECT NEXT SPEED
005140  062700  002000          ADD     #2000,R0
005144  005301                  DEC     R1
005146  001317                  BNE     1$
005150  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000015          LINE=LINE+1
        020000          BITX=BITX+BITX
005152                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                        ;TRANSMITTER LINE SPEED SELECTION TEST
                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 15
                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED
```

```
005152                          TS \XN,10,5$,1$
005152  012767  000340  172616  T16:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
005160  012767  000010  006636          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
005166  012767  005372  006624          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
005174  012767  005230  006620          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000017          XN=XN+1
005202  012705  000015                  MOV     #15,R5          ;LINE 15 WILL BE TESTED
005206  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
005212  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
005216  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
005222  012767  177777  006630          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
005230  012777  004000  006520  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
005236  010577  006514                  MOV     R5,@DHSCR               ;SELECT LINE 15 FOR TESTING
005242  005077  006516                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
005246  012777  177775  006512          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
005254  010077  006502                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
005260  005067  006576                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
005264  005067  006574                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
005270  012767  000010  006570          MOV     #10,TEMP2
005276  012777  020000  006464          MOV     #20000,@DHBAR           ;SET BAR BIT FOR LINE 15
                                                                        ;TO START TRANSMISSION
005304  005777  006446          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
005310  100412                  BMI     3$
005312  005267  006544          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
005316  005267  006542          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
005322  001370                  BNE     2$
005324  005367  006536          DEC     TEMP2
005330  001365                  BNE     2$
005332                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
005332  104001                  EMT     1
005334  000405                  BR      4$
005336  026767  006520  006514  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
005344  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
005346                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
005346  104002                  EMT     2
                                                                ;LINE 15
005350  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
005352  016767  006504  006500          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
005360  005204                  INC     R4                      ;SELECT NEXT SPEED
005362  062700  002000          ADD     #2000,R0
005366  005301                  DEC     R1
005370  001317                  BNE     1$
005372  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000016          LINE=LINE+1
        040000          BITX=BITX+BITX
005374                  SPEED   \LINE,\BITX,3,+//,2000,+/TRANSMITTER/

                        ;TRANSMITTER LINE SPEED SELECTION TEST
                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 16
```

```
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

005374                          TS \XN,10,5$,1$
005374  012767  000340  172374  T17:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
005402  012767  000010  006414          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
005410  012767  005614  006402          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
005416  012767  005452  006376          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000020                  XN=XN+1
005424  012705  000016                  MOV     #16,R5          ;LINE 16 WILL BE TESTED
005430  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
005434  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
005440  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
005444  012767  177777  006406          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
005452  012777  004000  006276  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
005460  010577  006272                  MOV     R5,@DHSCR               ;SELECT LINE 16 FOR TESTING
005464  005077  006274                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
005470  012777  177775  006270          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
005476  010077  006260                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
005502  005067  006354                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
005506  005067  006352                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
005512  012767  000010  006346          MOV     #10,TEMP2
005520  012777  040000  006242          MOV     #40000,@DHBAR           ;SET BAR BIT FOR LINE 16
                                                                        ;TO START TRANSMISSION
005526  005777  006224          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
005532  100412                  BMI     3$
005534  005267  006322          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
005540  005267  006320          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
005544  001370                  BNE     2$
005546  005367  006314          DEC     TEMP2
005552  001365                  BNE     2$
005554                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
005554  104001                  EMT     1
005556  000405                  BR      4$
005560  026767  006276  006272  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
005566  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
005570                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
005570  104002                  EMT     2
                                                                ;LINE 16
005572  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
005574  016767  006262  006256          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
005602  005204                  INC     R4                      ;SELECT NEXT SPEED
005604  062700  002000                  ADD     #2000,R0
005610  005301                  DEC     R1
005612  001317                  BNE     1$
005614  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000017                  LINE=LINE+1
        100000                  BITX=BITX+BITX
005616                          SPEED   \LINE,\BITX,3,^//,2000,^/TRANSMITTER/
```

```
                                        ;TRANSMITTER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 3 CHARACTERS AT A SELECTED SPEED ON LINE 17
                                        ;VERIFY THAT TRANSMITTER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

005616                          TS \XN,10,5$,1$
005616  012767  000340  172152  T20:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
005624  012767  000010  006172          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
005632  012767  006036  006160          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
005640  012767  005674  006154          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000021                  XN=XN+1
005646  012705  000017                  MOV     #17,R5          ;LINE 17 WILL BE TESTED
005652  012700  002000                  MOV     #2000,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
005656  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
005662  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
005666  012767  177777  006164          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
005674  012777  004000  006054  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
005702  010577  006050                  MOV     R5,@DHSCR               ;SELECT LINE 17 FOR TESTING
005706  005077  006052                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
005712  012777  177775  006046          MOV     #-3,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;3 CHARACTERS
005720  010077  006036                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
005724  005067  006132                  CLR     TIME2                   ;CLEAR TRANSMITTER TIME TIMER
005730  005067  006130                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
005734  012767  000010  006124          MOV     #10,TEMP2
005742  012777  100000  006020          MOV     #100000,@DHBAR          ;SET BAR BIT FOR LINE 17
                                                                        ;TO START TRANSMISSION
005750  005777  006002          2$:     TST     @DHSCR                  ;WAIT FOR TRANSMITTER
                                                                        ;TO FINISH
005754  100412                  BMI     3$
005756  005267  006100          INC     TIME2                   ;UPDATE TRANSMITTER TIMER
005762  005267  006076          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
005766  001370                  BNE     2$
005770  005367  006072          DEC     TEMP2
005774  001365                  BNE     2$
005776                          HLT     1                       ;TRANSMITTER DID NOT FINISH, ERROR
005776  104001                  EMT     1
006000  000405                  BR      4$
006002  026767  006054  006050  3$:     CMP     TIME2,TIME1             ;VERIFY THAT TRANSMITTER
006010  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
006012                          HLT     2                       ;TRANSMITTER TIMING ERROR FOR
006012  104002                  EMT     2
                                                                ;LINE 17
006014  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
006016  016767  006040  00603+          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
006024  005204                  INC     R4                      ;SELECT NEXT SPEED
006026  062700  002000                  ADD     #2000,R0
006032  005?01                  DEC     R1
006034  001317                  BNE     1$
006036  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                000020                  LINE=LINE+1
                000000                  BITX=BITX+BITX
     16         000020                  XLINE=LINE
     17         000000                  LINE=0
     18         000000                  XBIT=BITX
     19         000001                  BITX=1
     21         000020                  .REPT   20
     22                                 SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/
     23                                 .NLIST
     24                                 LINE=LINE+1
     25                                 BITX=BITX+BITX
     26                                 .LIST
     27                                 .ENDR
        006040                          SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 0
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

        006040                          TS \XN,10,5$,1$
        006040  012767  000340  171730  T21:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
        006046  012767  000010  005750          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
        006054  012767  006260  005736          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TC NEXT TEST
                                        .IF NB  <1$>
        006062  012767  006116  005732          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                        .ENDC
                000022                  XN=XN+1
        006070  012705  000000                  MOV     #0,R5           ;LINE 0 WILL BE TESTED
        006074  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
        006100  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
        006104  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
        006110  012767  177777  005742          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
        006116  012777  004000  005632  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
        006124  010577  005626                  MOV     R5,@DHSCR               ;SELECT LINE 0 FOR TESTING
        006130  005077  005630                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
        006134  012777  177777  005624          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                                ;1 CHARACTERS
        006142  010077  005614                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
        006146  005067  005710                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
        006152  005067  005706                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
        006156  012767  000010  005702          MOV     #10,TEMP2
        006164  012777  000001  005576          MOV     #1,@DHBAR               ;SET BAR BIT FOR LINE 0
                                                                                ;TO START TRANSMISSION
        006172  105777  005560          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                                ;TO FINISH
        006176  100412                          BMI     3$
        006200  005267  005656                  INC     TIME2                   ;UPDATE RECEIVER TIMER
        006204  005267  005654                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
        006210  001370                          BNE     2$
        006212  005367  005650                  DEC     TEMP2
        006216  001365                          BNE     2$
        006220                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
        006220  104001                          EMT     1
        006222  000405                          BR      4$
```

```
006224  026767  005632  005626  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
006232  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
006234                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
006234  104002                          EMT     2
                                                                        ;LINE 0
006236  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
006240  016767  005616  005612          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
006246  005204                          INC     R4                      ;SELECT NEXT SPEED
006250  062700  002100                  ADD     #2100,R0
006254  005301                          DEC     R1
006256  001317                          BNE     1$
006260  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000001                  LINE=LINE+1
        000002                  BITX=BITX+BITX
006262                          SPEED   \LINE,\BITX,1,+/B/,2100,+/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 1
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

006262                          TS \XN,10,5$,1$
006262  012767  000340  171506  T22:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
006270  012767  000010  005526          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
006276  012767  006502  005514          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
006304  012767  006340  005510          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000023          XN=XN+1
006312  012705  000001                  MOV     #1,R5           ;LINE 1 WILL BE TESTED
006316  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
006322  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
006326  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
006332  012767  177777  005520          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
006340  012777  004000  005410  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
006346  010577  005404                  MOV     R5,@DHSCR               ;SELECT LINE 1 FOR TESTING
006352  005077  005406                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
006356  012777  177777  005402          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
006364  010077  005372                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
006370  005067  005466                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
006374  005067  005464                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
006400  012767  000010  005460          MOV     #10,TEMP2
006406  012777  000002  005354          MOV     #2,@DHBAR               ;SET BAR BIT FOR LINE 1
                                                                        ;TO START TRANSMISSION
006414  105777  005336          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
006420  100412                          BMI     3$
006422  005267  005434                  INC     TIME2                   ;UPDATE RECEIVER TIMER
006426  005267  005432                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
006432  001370                          BNE     2$
006434  005367  005426                  DEC     TEMP2
006440  001365                          BNE     2$
```

```
006442                                          HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
006442  104001                                  EMT     1
006444  000405                                  BR      4$
006446  026767  005410  005404  3$:             CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
006454  103401                                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                                ;LESS THAN TIME1)
006456                                          HLT     2                       ;RECEIVER TIMING ERROR FOR
006456  104002                                  EMT     2
                                                                                ;LINE 1
006460  104410                          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
006462  016767  005374  005370                  MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
006470  005204                                  INC     R4                      ;SELECT NEXT SPEED
006472  062700  002100                          ADD     #2100,R4
006476  005301                                  DEC     R1
006500  001317                                  BNE     1$
006502  104400                          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000002                          LINE=LINE+1
        000004                          BITX=BITX+BITX
006504                                  SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                                ;RECEIVER LINE SPEED SELECTION TEST
                                                ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 2
                                                ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

006504                                  TS \XN,10,5$,1$
006504  012767  000340  171264          T23:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
006512  012767  000010  005304                  MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
006520  012767  006724  005272                  MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                        .IF NB  <1$>
006526  012767  006562  005266                  MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                        .ENDC
        000024                          XN=XN+1

006534  012705  000002                  MOV     #2,R5           ;LINE 2 WILL BE TESTED
006540  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
006544  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
006550  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
006554  012767  177777  005276          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
006562  012777  004000  005166  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
006570  010577  005162                  MOV     R5,@DHSCR               ;SELECT LINE 2 FOR TESTING
006574  005077  005164                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
006600  012777  177777  005160          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
006606  010077  005150                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
006612  005067  005244                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
006616  005067  005242                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
006622  012767  000010  005236          MOV     #10,TEMP2
006630  012777  000004  005132          MOV     #4,@DHBAR               ;SET BAR BIT FOR LINE 2
                                                                        ;TO START TRANSMISSION
006636  105777  005114          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
006642  100412                          BMI     3$
006644  005267  005212                  INC     TIME2                   ;UPDATE RECEIVER TIMER
006650  005267  005210                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
```

```
006654  001370                          BNE     2$
006656  005367  005204                  DEC     TEMP2
006662  001365                          BNE     2$
006664                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
006664  104001                          EMT     1
006666  000405                          BR      4$
006670  026767  005166  005162  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
006676  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
006700                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
006700  104002                          EMT     2
                                                                        ;LINE 2
006702  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
006704  016767  005152  005146          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
006712  005204                          INC     R4                      ;SELECT NEXT SPEED
006714  062700  002100                  ADD     #2100,R0
006720  005301                          DEC     R1
006722  001317                          BNE     1$
006724  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000003                  LINE=LINE+1
        000010                  BITX=BITX+BITX
006726                          SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 3
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

006726                          TS \XN,10,5$,1$
006726  012767  000340  171042  T24:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
006734  012767  000010  005062          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
006742  012767  007146  005050          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
006750  012767  007004  005044          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000025          XN=XN+1
006756  012705  000003                  MOV     #3,R5           ;LINE 3 WILL BE TESTED
006762  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
006766  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
006772  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
006776  012767  177777  005054          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
007004  012777  004000  004744  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
007012  010577  004740                  MOV     R5,@DHSCR               ;SELECT LINE 3 FOR TESTING
007016  005077  004742                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
007022  012777  177777  004736          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
007030  010077  004726                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
007034  005067  005022                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
007040  005067  005020                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
007044  012767  000010  005014          MOV     #10,TEMP2
007052  012777  000010  004710          MOV     #10,@DHBAR              ;SET BAR BIT FOR LINE 3
                                                                        ;TO START TRANSMISSION
007060  105777  004672          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
```

```
007064  100412                          BMI     3$
007066  005267  004770                  INC     TIME2                   ;UPDATE RECEIVER TIMER
007072  005267  004766                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
007076  001370                          BNE     2$
007100  005367  004762                  DEC     TEMP2
007104  001365                          BNE     2$
007106                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
007106  104001                          EMT     1
007110  000405                          BR      4$
007112  026767  004744  004740  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
007120  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
007122                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
007122  104002                          EMT     2
                                                                        ;LINE 3
007124  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
007126  016767  004730  004724          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
007134  005204                          INC     R4                      ;SELECT NEXT SPEED
007136  062700  002100                  ADD     #2100,R0
007142  005301                          DEC     R1
007144  001317                          BNE     1$
007146  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000004                  LINE=LINE+1
        000020                  BITX=BITX+BITX
007150                          SPEED   \LINE,\BITX,1,+/B/,2100,+/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 4
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

007150                          TS \XN,10,5$,1$
007150  012767  000340  170620  T25:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007156  012767  000010  004640          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
007164  012767  007370  004626          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
007172  012767  007226  004622          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000026                  XN=XN+1
007200  012705  000004                  MOV     #4,R5           ;LINE 4 WILL BE TESTED
007204  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
007210  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
007214  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
007220  012767  177777  004632          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
007226  012777  004000  004522  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
007234  010577  004516                  MOV     R5,@DHSCR               ;SELECT LINE 4 FOR TESTING
007240  005077  004520                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
007244  012777  177777  004514          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
007252  010077  004504                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
007256  005067  004600                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
007262  005067  004576                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
007266  012767  000010  004572          MOV     #10,TEMP2
007274  012777  000020  004466          MOV     #20,@DHBAR              ;SET BAR BIT FOR LINE 4
```

```
007302  105777  004450          2$:     TSTB    @DHSCR                  ;TO START TRANSMISSION
                                                                        ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
007306  100412                          BMI     3$
007310  005267  004546                  INC     TIME2                   ;UPDATE RECEIVER TIMER
007314  005267  004544                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
007320  001370                          BNE     2$
007322  005367  004540                  DEC     TEMP2
007326  001365                          BNE     2$
007330                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
007330  104001                          EMT     1
007332  000405                          BR      4$
007334  026767  004522  004516  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
007342  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
007344                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
007344  104002                          EMT     2
                                                                        ;LINE 4
007346  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
007350  016767  004506  004502          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
007356  005204                          INC     R4                      ;SELECT NEXT SPEED
007360  062700  002100                  ADD     #2100,R0
007364  005301                          DEC     R1
007366  001317                          BNE     1$
007370  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000005                  LINE=LINE+1
        000040                  BITX=BITX+BITX
007372                          SPEED   \LINE,\BITX,1,+/B/,2100,+/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 5
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

007372                          TS \XN,10,5$,1$
007372  012767  000340  170376  T26:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007400  012767  000010  004416          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
007406  012767  007612  004404          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
007414  012767  007450  004400          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000027          XN=XN+1
007422  012705  000005                  MOV     #5,R5           ;LINE 5 WILL BE TESTED
007426  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
007432  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
007436  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
007442  012767  177777  004410          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
007450  012777  004000  004300  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
007456  010577  004274                  MOV     R5,@DHSCR               ;SELECT LINE 5 FOR TESTING
007462  005077  004276                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
007466  012777  177777  004272          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
007474  010077  004262                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
007500  005067  004356                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
```

```
007504  005067  004354                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
007510  012767  000010  004350          MOV     #10,TEMP2
007516  012777  000040  004244          MOV     #40,@DHBAR              ;SET BAR BIT FOR LINE 5
                                                                        ;TO START TRANSMISSION
007524  105777  004226          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
007530  100412                          BMI     3$
007532  005267  004324                  INC     TIME2                   ;UPDATE RECEIVER TIMER
007536  005267  004322                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
007542  001370                          BNE     2$
007544  005367  004316                  DEC     TEMP2
007550  001365                          BNE     2$
007552                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
007552  104001                          EMT     1
007554  000405                          BR      4$
007556  026767  004300  004274  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
007564  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
007566                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
007566  104002                          EMT     2
                                                                        ;LINE 5
007570  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
007572  016767  004264  004260          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
007600  005204                          INC     R4                      ;SELECT NEXT SPEED
007602  062700  002100                  ADD     #2100,R0
007606  005301                          DEC     R1
007610  001317                          BNE     1$
007612  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000006                  LINE=LINE+1
        000100                  BITX=BITX+BITX
007614                          SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                ;RECEIVER LINE SPEED SELECTION TEST
                                ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 6
                                ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

007614                          TS \XN,10,5$,1$
007614  012767  000340  170154  T27:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007622  012767  000010  004174          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
007630  012767  010034  004162          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
007636  012767  007672  004156          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000030          XN=XN+1
007644  012705  000006                  MOV     #6,R5           ;LINE 6 WILL BE TESTED
007650  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
007654  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
007660  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
007664  012767  177777  004166          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
007672  012777  004000  004056  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
007700  010577  004052                  MOV     R5,@DHSCR               ;SELECT LINE 6 FOR TESTING
007704  005077  004054                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
007710  012777  177777  004050          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
```

```
007716  010077  004040                    MOV     R0,@DHLPR               ;1 CHARACTERS
007722  005067  004134                    CLR     TIME2                   ;SELECT LINE SPEED
007726  005067  004132                    CLR     TEMP1                   ;CLEAR RECEIVER TIME TIMER
007732  012767  000010  004126            MOV     #10,TEMP2               ;SET UP NO CLOCK TIMER
007740  012777  000100  004022            MOV     #100,@DHBAR             ;SET BAR BIT FOR LINE 6
                                                                          ;TO START TRANSMISSION
007746  105777  004004          2$:       TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                          ;TO FINISH
007752  100412                            BMI     3$
007754  005267  004102                    INC     TIME2                   ;UPDATE RECEIVER TIMER
007760  005267  004100                    INC     TEMP1                   ;UPDATE NO CLOCK TIMER
007764  001370                            BNE     2$
007766  005367  004074                    DEC     TEMP2
007772  001365                            BNE     2$
007774                                    HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
007774  104001                            EMT     1
007776  000405                            BR      4$
010000  026767  004056  004052  3$:       CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
010006  103401                            BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                          ;(NUMBER OF COUNTS IN TIME2
                                                                          ;LESS THAN TIME1)
010010                                    HLT     2                       ;RECEIVER TIMING ERROR FOR
010010  104002                            EMT     2
                                                                          ;LINE 6
010012  104410                  4$:       SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
010014  016767  004042  004036            MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
010022  005204                            INC     R4                      ;SELECT NEXT SPEED
010024  062700  002100                    ADD     #2100,R0
010030  005301                            DEC     R1
010032  001317                            BNE     1$
010034  104400                  5$:       SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000007          LINE=LINE+1
        000200          BITX=BITX+BITX
010036                  SPEED   \LINE,\BITX,1.+/8/,2100.+/RECEIVER/

                                ;RECEIVER LINE SPEED SELECTION TEST
                                ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 7
                                ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

010036                  TS \XN,10,5$,1$
010036  012767  000340  167732  T30:      MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
010044  012767  000010  003752            MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
010052  012767  010256  003740            MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
010060  012767  010114  003734            MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000031          XN=XN+1
010066  012705  000007                    MOV     #7,R5           ;LINE 7 WILL BE TESTED
010072  012700  002100                    MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                          ;OF INITIAL (LOWEST) SPEED
010076  012701  000015                    MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
010102  012704  000001                    MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
010106  012767  177777  003744            MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
010114  012777  004000  003634  1$:       MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
```

```
010122  010577  003630                  MOV     R5,@DHSCR       ;SELECT LINE 7 FOR TESTING
010126  005077  003632                  CLR     @DHBA           ;CLEAR BUS ADDRESS
010132  012777  177777  003626          MOV     #-1,@DHBC       ;SET UP TO TRANSMIT
                                                                ;1 CHARACTERS
010140  010077  003616                  MOV     R0,@DHLPR       ;SELECT LINE SPEED
010144  005067  003712                  CLR     TIME2           ;CLEAR RECEIVER TIME TIMER
010150  005067  003710                  CLR     TEMP1           ;SET UP NO CLOCK TIMER
010154  012767  000010  003704          MOV     #10,TEMP2
010162  012777  000200  003600          MOV     #200,@DHBAR     ;SET BAR BIT FOR LINE 7
                                                                ;TO START TRANSMISSION
010170  105777  003562          2$:     TSTB    @DHSCR          ;WAIT FOR RECEIVER
                                                                ;TO FINISH
010174  100412                          BMI     3$
010176  005267  003660                  INC     TIME2           ;UPDATE RECEIVER TIMER
010202  005267  003656                  INC     TEMP1           ;UPDATE NO CLOCK TIMER
010206  001370                          BNE     2$
010210  005367  003652                  DEC     TEMP2
010214  001365                          BNE     2$
010216                                  HLT     1               ;RECEIVER DID NOT FINISH, ERROR
010216  104001                          EMT     1
010220  000405                          BR      4$
010222  026767  003634  003630  3$:     CMP     TIME2,TIME1     ;VERIFY THAT RECEIVER
010230  103401                          BLO     4$              ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
010232                                  HLT     2               ;RECEIVER TIMING ERROR FOR
010232  104002                          EMT     2
                                                                ;LINE 7
010234  104410                  4$:     SCOPE1                  ;CHECK FOR FREEZE ON CURRENT DATA
010236  016767  003620  003614          MOV     TIME2,TIME1     ;SET UP FOR NEXT COMPARISION
010244  005204                          INC     R4              ;SELECT NEXT SPEED
010246  062700  002100                  ADD     #2100,R0
010252  005301                          DEC     R1
010254  001317                          BNE     1$
010256  104400                  5$:     SCOPE                   ;CHECK FOR ITERATIONS, LOOP
        000010                  LINE=LINE+1
        000400                  BITX=BITX+BITX
010260                          SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 10
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

010260                          TS \XN,10,5$,1$
010260  012767  000340  167510  T31:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
010266  012767  000010  003530          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
010274  012767  010500  003516          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                        .IF NB  <1$>
010302  012767  010336  003512          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                        .ENDC
        000032                  XN=XN+1
010310  012705  000010                  MOV     #10,R5          ;LINE 10 WILL BE TESTED
010314  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                ;OF INITIAL (LOWEST) SPEED
010320  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
```

```
010324  012704  000001              MOV     #1,R4           ;BINARY CODE FOR INITIAL SPEED
010330  012767  177777  003522      MOV     #-1,TIME1       ;INITIALIZE COMPARISION VALUE
010336  012777  004000  003412  1$: MOV     #BIT11,@DHSCR   ;CLEAR INTERFACE
010344  010577  003406              MOV     R5,@DHSCR       ;SELECT LINE 10 FOR TESTING
010350  005077  003410              CLR     @DHBA           ;CLEAR BUS ADDRESS
010354  012777  177777  003404      MOV     #-1,@DHBC       ;SET UP TO TRANSMIT
                                                            ;1 CHARACTERS
010362  010077  003374              MOV     R0,@DHLPR       ;SELECT LINE SPEED
010366  005067  003470              CLR     TIME2           ;CLEAR RECEIVER TIME TIMER
010372  005067  003466              CLR     TEMP1           ;SET UP NO CLOCK TIMER
010376  012767  000010  003462      MOV     #10,TEMP2
010404  012777  000400  003356      MOV     #400,@DHBAR     ;SET BAR BIT FOR LINE 10
                                                            ;TO START TRANSMISSION
010412  105777  003340          2$: TSTB    @DHSCR          ;WAIT FOR RECEIVER
                                                            ;TO FINISH
010416  100412                      BMI     3$
010420  005267  003436              INC     TIME2           ;UPDATE RECEIVER TIMER
010424  005267  003434              INC     TEMP1           ;UPDATE NO CLOCK TIMER
010430  001370                      BNE     2$
010432  005367  003430              DEC     TEMP2
010436  001365                      BNE     2$
010440                              HLT     1               ;RECEIVER DID NOT FINISH, ERROR
010440  104001                      EMT     1
010442  000405                      BR      4$
010444  026767  003412  003406  3$: CMP     TIME2,TIME1     ;VERIFY THAT RECEIVER
010452  103401                      BLO     4$              ;WAS FASTER AT THIS SELECTED SPEED
                                                            ;(NUMBER OF COUNTS IN TIME2
                                                            ;LESS THAN TIME1)
010454                              HLT     2               ;RECEIVER TIMING ERROR FOR
010454  104002                      EMT     2
                                                            ;LINE 10
010456  104410                  4$: SCOPE1                  ;CHECK FOR FREEZE ON CURRENT DATA
010460  016767  003376  003372      MOV     TIME2,TIME1     ;SET UP FOR NEXT COMPARISION
010466  005204                      INC     R4              ;SELECT NEXT SPEED
010470  062700  002100              ADD     #2100,R0
010474  005301                      DEC     R1
010476  001317                      BNE     1$
010500  104400                  5$: SCOPE                   ;CHECK FOR ITERATIONS, LOOP
        000011              LINE=LINE+1
        001000              BITX=BITX+BITX
010502                      SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                    ;RECEIVER LINE SPEED SELECTION TEST
                                    ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 11
                                    ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                    ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                    ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

010502                      TS \XN,10,5$,1$
010502  012767  000340  167266  T32:  MOV     #340,PS         ;DISABLE ALL INTERRUPTS
010510  012767  000010  003306      MOV     #10,ICOUNT      ;SET UP FOR 10 ITERATIONS
010516  012767  010722  003274      MOV     #5$,ESCAPE      ;SET UP TO ESCAPE TO NEXT TEST
                            .IF NB  <1$>
010524  012767  010560  003270      MOV     #1$,FREEZ1      ;SET UP TO LOOP WITH DATA       ; 3
                            .ENDC
        000033              XN=XN+1
010532  012705  000011              MOV     #11,R5          ;LINE 11 WILL BE TESTED
```

```
010536  012700  002100              MOV     #2100,R0        ;CONSTANT FOR SELECTION
                                                            ,OF INITIAL (LOWEST) SPEED
010542  012701  000015              MOV     #15,R1          ;15 DIFFERENT SPEEDS WILL BE TESTED
010546  012704  000001              MOV     #1,R4           ;BINARY CODE FOR INITIAL SPEED
010552  012767  177777  003300      MOV     #-1,TIME1       ;INITIALIZE COMPARISION VALUE
010560  012777  004000  003170  1$: MOV     #BIT11,@DHSCR   ;CLEAR INTERFACE
010566  010577  003164              MOV     R5,@DHSCR       ;SELECT LINE 11 FOR TESTING
010572  005077  003166              CLR     @DHBA           ;CLEAR BUS ADDRESS
010576  012777  177777  003162      MOV     #-1,@DHBC       ;SET UP TO TRANSMIT
                                                            ;1 CHARACTERS
010604  010077  003152              MOV     R0,@DHLPR       ;SELECT LINE SPEED
010610  005067  003246              CLR     TIME2           ;CLEAR RECEIVER TIME TIMER
010614  005067  003244              CLR     TEMP1           ;SET UP NO CLOCK TIMER
010620  012767  000010  003240      MOV     #10,TEMP2
010626  012777  001000  003134      MOV     #1000,@DHBAR    ;SET BAR BIT FOR LINE 11
                                                            ;TO START TRANSMISSION
010634  105777  003116          2$: TSTB    @DHSCR          ;WAIT FOR RECEIVER
                                                            ;TO FINISH
010640  100412                      BMI     3$
010642  005267  003214              INC     TIME2           ;UPDATE RECEIVER TIMER
010646  005267  003212              INC     TEMP1           ;UPDATE NO CLOCK TIMER
010652  001370                      BNE     2$
010654  005367  003206              DEC     TEMP2
010660  001365                      BNE     2$
010662                              HLT     1               ;RECEIVER DID NOT FINISH, ERROR
010662  104001                      EMT     1
010664  000405                      BR      4$
010666  026767  003170  003164  3$: CMP     TIME2,TIME1     ;VERIFY THAT RECEIVER
010674  103401                      BLO     4$              ;WAS FASTER AT THIS SELECTED SPEED
                                                            ;(NUMBER OF COUNTS IN TIME2
                                                            ;LESS THAN TIME1)
010676                              HLT     2               ;RECEIVER TIMING ERROR FOR
010676  104002                      EMT     2
                                                            ;LINE 11
010700  104410                  4$: SCOPE1                  ;CHECK FOR FREEZE ON CURRENT DATA
010702  016767  003154  003150      MOV     TIME2,TIME1     ;SET UP FOR NEXT COMPARISION
010710  005204                      INC     R4              ;SELECT NEXT SPEED
010712  062700  002100              ADD     #2100,R0
010716  005301                      DEC     R1
010720  001317                      BNE     1$
010722  104400                  5$: SCOPE                   ;CHECK FOR ITERATIONS, LOOP
        000012          LINE=LINE+1
        002000          BITX=BITX+BITX
010724                  SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                ;RECEIVER LINE SPEED SELECTION TEST
                                ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 12
                                ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

010724                  TS \XM^10,5$,1$
010724  012767  000340  167044 T33: MOV     #340,PS         ;DISABLE ALL INTERRUPTS
010732  012767  000010  003064      MOV     #10,ICOUNT      ;SET UP FOR 10 ITERATIONS
010740  012767  011144  003052      MOV     #5$,ESCAPE      ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
010746  012767  011002  003046      MOV     #1$,FREEZ1      ;SET UP TO LOOP WITH DATA       ; 3
```

```
                                .ENDC
        000034                  XN=XN+1
 010754 012705 000012           MOV     #12,R5          ;LINE 12 WILL BE TESTED
 010760 012700 002100           MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                ;OF INITIAL (LOWEST) SPEED
 010764 012701 000015           MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
 010770 012704 000001           MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
 010774 012767 177777 003056    MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
 011002 012777 004000 002746 1$: MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
 011010 010577 002742           MOV     R5,@DHSCR               ;SELECT LINE 12 FOR TESTING
 011014 005077 002744           CLR     @DHBA                   ;CLEAR BUS ADDRESS
 011020 012777 177777 002740    MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                ;1 CHARACTERS
 011026 010077 002730           MOV     R0,@DHLPR               ;SELECT LINE SPEED
 011032 005067 003024           CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
 011036 005067 003022           CLR     TEMP1                   ;SET UP NO CLOCK TIMER
 011042 012767 000010 003016    MOV     #10,TEMP2
 011050 012777 002000 002712    MOV     #2000,@DHBAR            ;SET BAR BIT FOR LINE 12
                                                                ;TO START TRANSMISSION
 011056 105777 002674        2$: TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                ;TO FINISH
 011062 100412                  BMI     3$
 011064 005267 002772           INC     TIME2                   ;UPDATE RECEIVER TIMER
 011070 005267 002770           INC     TEMP1                   ;UPDATE NO CLOCK TIMER
 011074 001370                  BNE     2$
 011076 005367 002764           DEC     TEMP2
 011102 001365                  BNE     2$
 011104                         HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
 011104 104001                  EMT     1
 011106 000405                  BR      4$
 011110 026767 002746 002742 3$: CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
 011116 103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
 011120                         HLT     2                       ;RECEIVER TIMING ERROR FOR
 011120 104002                  EMT     2
                                                                ;LINE 12
 011122 104410               4$: SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
 011124 016767 002732 002726    MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
 011132 005204                  INC     R4                      ;SELECT NEXT SPEED
 011134 062700 002100           ADD     #2100,R0
 011140 005301                  DEC     R1
 011142 001317                  BNE     1$
 011144 104400               5$: SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000013                  LINE=LINE+1
        004000                  BITX=BITX+BITX
 011146                         SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                ;RECEIVER LINE SPEED SELECTION TEST
                                ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 13
                                ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

 011146                         TS \XN,10,5$,1$
 011146 012767 000340 166622 T34: MOV    #340,PS                 ;DISABLE ALL INTERRUPTS
 011154 012767 000010 002642    MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
```

```
011162  012767  011366  002630          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
011170  012767  011224  002624          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000035                  XN=XN+1
011176  012705  000013                  MOV     #13,R5          ;LINE 13 WILL BE TESTED
011202  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
011206  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
011212  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
011216  012767  177777  002634          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
011224  012777  004000  002524  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
011232  010577  002520                  MOV     R5,@DHSCR               ;SELECT LINE 13 FOR TESTING
011236  005077  002522                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
011242  012777  177777  002516          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
011250  010077  002506                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
011254  005067  002602                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
011260  005067  002600                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
011264  012767  000010  002574          MOV     #10,TEMP2
011272  012777  004000  002470          MOV     #4000,@DHBAR            ;SET BAR BIT FOR LINE 13
                                                                        ;TO START TRANSMISSION
011300  105777  002452          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
011304  100412                  BMI     3$
011306  005267  002550          INC     TIME2                   ;UPDATE RECEIVER TIMER
011312  005267  002546          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
011316  001370                  BNE     2$
011320  005367  002542          DEC     TEMP2
011324  001365                  BNE     2$
011326                          HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
011326  104001                  EMT     1
011330  000405                  BR      4$
011332  026767  002524  002520  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
011340  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
011342                          HLT     2                       ;RECEIVER TIMING ERROR FOR
011342  104002                  EMT     2
                                                                ;LINE 13
011344  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
011346  016767  002510  002504          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
011354  005204                  INC     R4                      ;SELECT NEXT SPEED
011356  062700  002100                  ADD     #2100,R0
011362  005301                  DEC     R1
011364  001317                  BNE     1$
011366  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000014          LINE=LINE+1
        010000          BITX=BITX+BITX
011370                  SPEED   \LINE,\BITX,1,^/8/,2100,^/RECEIVER/

                        ;RECEIVER LINE SPEED SELECTION TEST
                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 14
                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED
```

```
011370                                  TS \XN,10,5$,1$
011370  012767  000340  166400  T35:    MOV     #340,PS             ;DISABLE ALL INTERRUPTS
011376  012767  000010  002420          MOV     #10,ICOUNT          ;SET UP FOR 10 ITERATIONS
011404  012767  011610  002406          MOV     #5$,ESCAPE          ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
011412  012767  011446  002402          MOV     #1$,FREEZ1          ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000036                  XN=XN+1
011420  012705  000014                  MOV     #14,R5          ;LINE 14 WILL BE TESTED
011424  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
011430  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
011434  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
011440  012767  177777  002412          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
011446  012777  004000  002302  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
011454  010577  002276                  MOV     R5,@DHSCR               ;SELECT LINE 14 FOR TESTING
011460  005077  002300                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
011464  012777  177777  002274          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
011472  010077  002264                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
011476  005067  002360                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
011502  005067  002356                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
011506  012767  000010  002352          MOV     #10,TEMP2
011514  012777  010000  002246          MOV     #10000,@DHBAR           ;SET BAR BIT FOR LINE 14
                                                                        ;TO START TRANSMISSION
011522  105777  002230          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
011526  100412                          BMI     3$
011530  005267  002326                  INC     TIME2                   ;UPDATE RECEIVER TIMER
011534  005267  002324                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
011540  001370                          BNE     2$
011542  005367  002320                  DEC     TEMP2
011546  001365                          BNE     2$
011550                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
011550  104001                          EMT     1
011552  000405                          BR      4$
011554  026767  002302  002276  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
011562  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
011564                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
011564  104002                          EMT     2
                                                                        ;LINE 14
011566  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
011570  016767  002266  002262          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
011576  005204                          INC     R4                      ;SELECT NEXT SPEED
011600  062700  002100                  ADD     #2100,R0
011604  005301                          DEC     R1
011606  001317                          BNE     1$
011610  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000015                  LINE=LINE+1
        020000                  BITX=BITX+BITX
011612                          SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/

                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 15
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
```

```
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

011612                          TS \XN,10,5$,1$
011612  012767  000340  166156  T36:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
011620  012767  000010  002176          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
011626  012767  012032  002164          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
011634  012767  011670  002160          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000037                  XN=XN+1
011642  012705  000015                  MOV     #15,R5          ;LINE 15 WILL BE TESTED
011646  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
011652  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
011656  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
011662  012767  177777  002170          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
011670  012777  004000  002060  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
011676  010577  002054                  MOV     R5,@DHSCR               ;SELECT LINE 15 FOR TESTING
011702  005077  002056                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
011706  012777  177777  002052          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
011714  010077  002042                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
011720  005067  002136                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
011724  005067  002134                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
011730  012767  000010  002130          MOV     #10,TEMP2
011736  012777  020000  002024          MOV     #20000,@DHBAR           ;SET BAR BIT FOR LINE 15
                                                                        ;TO START TRANSMISSION
011744  105777  002006          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
011750  100412                  BMI     3$
011752  005267  002104          INC     TIME2                   ;UPDATE RECEIVER TIMER
011756  005267  002102          INC     TEMP1                   ;UPDATE NO CLOCK TIMER
011762  001370                  BNE     2$
011764  005367  002076          DEC     TEMP2
011770  001365                  BNE     2$
011772                          HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
011772  104001                  EMT     1
011774  000405                  BR      4$
011776  026767  002060  002054  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
012004  103401                  BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                ;(NUMBER OF COUNTS IN TIME2
                                                                ;LESS THAN TIME1)
012006                          HLT     2                       ;RECEIVER TIMING ERROR FOR
012006  104002                  EMT     2
                                                                ;LINE 15
012010  104410          4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
012012  016767  002044  002040          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
012020  005204                  INC     R4                      ;SELECT NEXT SPEED
012022  062700  002100                  ADD     #2100,R0
012026  005301                  DEC     R1
012030  001317                  BNE     1$
012032  104400          5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000016          LINE=LINE+1
        040000          BITX=BITX+BITX
012034                  SPEED   \LINE,\BITX,1,^/B/,2100,^/RECEIVER/
```

```
                                        ;RECEIVER LINE SPEED SELECTION TEST
                                        ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 16
                                        ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                        ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                        ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

012034                          TS \XN,10,5$,1$
012034  012767  000340  165734  T37:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
012042  012767  000010  001754          MOV     #10.ICOUNT              ;SET UP FOR 10 ITERATIONS
012050  012767  012254  001742          MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                .IF NB  <1$>
012056  012767  012112  001736          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                .ENDC
        000040                  XN=XN+1
012064  012705  000016                  MOV     #16,R5          ;LINE 16 WILL BE TESTED
012070  012700  002100                  MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                        ;OF INITIAL (LOWEST) SPEED
012074  012701  000015                  MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
012100  012704  000001                  MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
012104  012767  177777  001746          MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
012112  012777  004000  001636  1$:     MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
012120  010577  001632                  MOV     R5,@DHSCR               ;SELECT LINE 16 FOR TESTING
012124  005077  001634                  CLR     @DHBA                   ;CLEAR BUS ADDRESS
012130  012777  177777  001630          MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                        ;1 CHARACTERS
012136  010077  001620                  MOV     R0,@DHLPR               ;SELECT LINE SPEED
012142  005067  001714                  CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
012146  005067  001712                  CLR     TEMP1                   ;SET UP NO CLOCK TIMER
012152  012767  000010  001706          MOV     #10,TEMP2
012160  012777  040000  001602          MOV     #40000,@DHBAR           ;SET BAR BIT FOR LINE 16
                                                                        ;TO START TRANSMISSION
012166  105777  001564          2$:     TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                        ;TO FINISH
012172  100412                          BMI     3$
012174  005267  001662                  INC     TIME2                   ;UPDATE RECEIVER TIMER
012200  005267  001660                  INC     TEMP1                   ;UPDATE NO CLOCK TIMER
012204  001370                          BNE     2$
012206  005367  001654                  DEC     TEMP2
012212  001365                          BNE     2$
012214                                  HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
012214  104001                          EMT     1
012216  000405                          BR      4$
012220  026767  001636  001632  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
012226  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
012230                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
012230  104002                          EMT     2
                                                                        ;LINE 16
012232  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
012234  016767  001622  001616          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
012242  005204                          INC     R4                      ;SELECT NEXT SPEED
012244  062700  002100                  ADD     #2100,R0
012250  005301                          DEC     R1
012252  001317                          BNE     1$
012254  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000017                  LINE=LINE+1
```

```
          100000                    BITX=BITX+BITX
012256                              SPEED   \LINE,\BITX,1,+/B/,2100,+/RECEIVER/

                                    ;RECEIVER LINE SPEED SELECTION TEST
                                    ;TRANSMIT 1 CHARACTERS AT A SELECTED SPEED ON LINE 17
                                    ;VERIFY THAT RECEIVER DONE OCCURS AT THE SELECTED SPEED
                                    ;VERIFY THAT THE AMOUN OF TIME TAKEN IS LESS
                                    ;AT THIS SPEED THAN AT THE PREVIOUSLY SELECTED SPEED

012256                              TS \XN,10,5$,1$
012256  012767  000340  165512      T40:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
012264  012767  000010  001532              MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
012272  012767  012476  001520              MOV     #5$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
                                    .IF NB  <1$>
012300  012767  012334  001514              MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA       ; 3
                                    .ENDC
          000041                    XN=XN+1
012306  012705  000017              1$:     MOV     #17,R5          ;LINE 17 WILL BE TESTED
012312  012700  002100                      MOV     #2100,R0                ;CONSTANT FOR SELECTION
                                                                            ;OF INITIAL (LOWEST) SPEED
012316  012701  000015                      MOV     #15,R1                  ;15 DIFFERENT SPEEDS WILL BE TESTED
012322  012704  000001                      MOV     #1,R4                   ;BINARY CODE FOR INITIAL SPEED
012326  012767  177777  001524              MOV     #-1,TIME1               ;INITIALIZE COMPARISION VALUE
012334  012777  004000  001414              MOV     #BIT11,@DHSCR           ;CLEAR INTERFACE
012342  010577  001410                      MOV     R5,@DHSCR               ;SELECT LINE 17 FOR TESTING
012346  005077  001412                      CLR     @DHBA                   ;CLEAR BUS ADDRESS
012352  012777  177777  001406              MOV     #-1,@DHBC               ;SET UP TO TRANSMIT
                                                                            ;1 CHARACTERS
012360  010077  001376                      MOV     R0,@DHLPR               ;SELECT LINE SPEED
012364  005067  001472                      CLR     TIME2                   ;CLEAR RECEIVER TIME TIMER
012370  005067  001470                      CLR     TEMP1                   ;SET UP NO CLOCK TIMER
012374  012767  000010  001464              MOV     #10,TEMP2
012402  012777  100000  001360              MOV     #100000,@DHBAR          ;SET BAR BIT FOR LINE 17
                                                                            ;TO START TRANSMISSION
012410  105777  001342          2$:         TSTB    @DHSCR                  ;WAIT FOR RECEIVER
                                                                            ;TO FINISH
012414  100412                              BMI     3$
012416  005267  001440                      INC     TIME2                   ;UPDATE RECEIVER TIMER
012422  005267  001436                      INC     TEMP1                   ;UPDATE NO CLOCK TIMER
012426  001370                              BNE     2$
012430  005367  001432                      DEC     TEMP2
012434  001365                              BNE     2$
012436                                      HLT     1                       ;RECEIVER DID NOT FINISH, ERROR
012436  104001                              EMT     1
012440  000405                              BR      4$
```

```
012442  026767  001414  001410  3$:     CMP     TIME2,TIME1             ;VERIFY THAT RECEIVER
012450  103401                          BLO     4$                      ;WAS FASTER AT THIS SELECTED SPEED
                                                                        ;(NUMBER OF COUNTS IN TIME2
                                                                        ;LESS THAN TIME1)
012452                                  HLT     2                       ;RECEIVER TIMING ERROR FOR
012452  104002                          EMT     2

                                                                        ;LINE 17
012454  104410                  4$:     SCOPE1                          ;CHECK FOR FREEZE ON CURRENT DATA
012456  016767  001400  001374          MOV     TIME2,TIME1             ;SET UP FOR NEXT COMPARISION
012464  005204                          INC     R4                      ;SELECT NEXT SPEED
012466  062700  002100                  ADD     #2100,R0
012472  005301                          DEC     R1
012474  001317                          BNE     1$
012476  104400                  5$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
        000020                  LINE=LINE+1
        000000                  BITX=BITX+BITX
```

```
      1 012500                          .EOP    †/BEGIN/

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

 012500  104401                   EOP:   TYPE                    ;TYPE NAME OF TEST
 012502  014460                          MEPASS
 012504  005067  001344                  CLR     LAST           ;CLEAR LAST ERROR PC
 012510  005067  001274                  CLR     ERRFLG         ;CLEAR ERROR FLAG
 012514  005267  001272                  INC     PASCNT         ;UPDATE PASS COUNT
 012520  005767  166256                  TST     LIGHTS         ; ARE WE USING LIGHTS?          ; 4
 012524  001005                          BNE     2$             ; BRANCH IF WE ARE              ; 6
 012526  104401                          TYPE                   ; TYPE PASCOUNT MESSAGE         ; 5
 012530  014473                          PASTXT                                                 ; 5
 012532  104402                          OCTASC                 ; PRINT PASSCOUNT               ; 4
 012534  012572                          PASARG                                                       ;4 ; 6
 012536  000403                          BR      3$             ; CONTINUE                      ; 4
 012540                           2$:                                                           ; 4
 012540  016767  001246  166234          MOV     PASCNT,LIGHTS  ;DISPLAY PASS COUNT
 012546                           3$:                                                           ; 4
 012546  013701  000042                  MOV     @#42,R1        ;CHECK FOR ACT-11 OR DDP
 012552  001405                          BEQ     RESTRT         ;IF NOT, CONTINUE TESTING
 012554  000005                          RESET
 012556  004711                  LOGICAL:        JSR     PC,(R1)
 012560  000240                          NOP
 012562  000240                          NOP
 012564  000240                          NOP
 012566  000167  166514          RESTRT: JMP     BEGIN
 012572  000001                  PASARG: .WORD   1              ; PARAMETERS TO PRINT PASSCOUNT  ; 5
 012574    006     002                   .BYTE   6,2                                            ; 5
 012576  014012                          .WORD   PASCN.                                         ; 5
      2 012600                          .SCOPE

                                                ;CHECK FOR LOOP ON CURRENT TEST                 ; 3
                                                ;CHECK FOR ITERATION SUPPRESSION

 012600  032777  002000  166172  SCOPER: BIT     @#SW10,@SWR                                    ; 4
 012606  001030                          BNE     4$
 012610  032777  040000  166162  1$:     BIT     @#SW14,@SWR                                    ; 4
 012616  001021                          BNE     3$
 012620  032777  004000  166152          BIT     @#SW11,@SWR                                    ; 4
 012626  001006                          BNE     2$
 012630  005267  001172                  INC     LPCNT
 012634  026767  001166  001162          CMP     LPCNT,ICOUNT
 012642  001007                          BNE     3$
 012644  005067  001156          2$:     CLR     LPCNT
 012650  005067  001134                  CLR     ERRFLG
 012654  011667  001136                  MOV     (SP),RETRN
 012660  000002                          RTI
 012662  016716  001130          3$:     MOV     RETRN,(SP)
 012666  000002                          RTI
 012670  005767  001114          4$:     TST     ERRFLG
 012674  001745                          BEQ     1$
 012676  000762                          BR      2$
```

```
      3 012700                          .SCOP1

                                                ;CHECK FOR FREEZE ON CURRENT DATA

        012700  032777  001000  166072  SCOP1R: BIT     @SW09,@SWR                    ; 4
        012706  001402                          BEQ     1$
        012710  016716  001106                  MOV     FREEZ1,(SP)
        012714  000002                  1$:     RTI
```

```
1 012716                                .ERROR

                                        ;ERROR HANDLER

 012716  032777  020000  166054 ERRORS: BIT     @#SW13,@#SWR                    ; 4
 012724  001055                         BNE     HALTS
 012726  021667  001122                 CMP     (SP),LAST
 012732  001404                         BEQ     1$
 012734  011667  001114                 MOV     (SP),LAST
 012740  005067  001044                 CLR     ERRFLG
 012744  104406                 1$:     SAV05P
 012746  011605                         MOV     (SP),R5
 012750  162705  000002                 SUB     #2,R5
 012754  011504                         MOV     (R5),R4
 012756  006304                         ASL     R4
 012760  006304                         ASL     R4
 012762  042704  177001                 BIC     #177001,R4
 012766  062704  014672                 ADD     #ERRTAB,R4
 012772  012467  000040                 MOV     (R4)+,ERRMSG
 012776  011467  000052                 MOV     (R4),DATABP
 013002  005767  001002                 TST     ERRFLG
 013006  001403                         BEQ     TYPMSG
 013010  005767  000040                 TST     DATABP
 013014  001011                         BNE     TYPDAT                          ; 3
 013016  104401                 TYPMSG: TYPE                                    ; 5
 013020  014370                         MCRLF                                   ; 5
 013022  104402                         OCTASC                                  ; 5
 013024  013122                         ERTABO
 013026  012767  000001  000754         MOV     #1,ERRFLG
 013034  104401                         TYPE
 013036  000000                 ERRMSG: 0
 013040  005767  000010         TYPDAT: TST     DATABP
 013044  001404                         BEQ     RESREG
 013046  104401                         TYPE                                    ; 5
 013050  014370                         MCRLF                                   ; 5
 013052  104402                         OCTASC
 013054  000000                 DATABP: 0
 013056  104407                 RESREG: RES05
 013060  005777  165714         HALTS:  TST     @#SWR                           ; 4
 013064  100005                         BPL     EXITER
 013066  010046                         PUSHR0
 013070  016600  000002                 MOV     2(SP),R0
 013074  000000                         HALT
 013076  012600                         POPR0
 013100  005267  000710         EXITER: INC     ERRCNT
 013104  032777  002000  165666         BIT     @#SW10,@#SWR                    ; 4
 013112  001402                         BEQ     1$
 013114  016716  000700                 MOV     ESCAPE,(SP)
 013120  000002                 1$:     RTI
 013122  000001                 ERTABO: 1
 013124     006     002                 .BYTE   6,2
 013126  014046                         SAVPC
```

```
                        .TRPSRV
                                ;TRAP DISPATCH SERVICE
                                ;ARGUMENT OF TRAP IS EXTRACTED
                                ;AND USED AS OFFSET TO OBTAIN POINTER                    ; 3
                                ;TO SELECTED SUBROUTINE

013130  011646          TRPSRV: MOV     (SP),-(SP)          ;GET PC OF RETURN
013132  162716  000002          SUB     #2,(SP)             ;=PC OF TRAP
013136  017616  000000          MOV     @(SP),(SP)          ;GET TRP
013142  006316          TRPOK:  ASL     (SP)                ;MULTIPLY TRAP ARG BY 2
013144  042716  177001          BIC     #177001,(SP)        ;CLEAR UNWANTED BITS
013150  062716  014612          ADD     #TRPTAB,(SP)        ;POINTER TO SUBROUTINE ADDRESS
013154  017616  000000          MOV     @(SP),(SP)          ;SUBROUTINE ADDRESS
013160  000136                  JMP     @(SP)+              ;GO TO SUBROUTINE
  2 013162                      .SAVREG

                                ;SAVE PC OF TEST THAT FAILED AND R0-R5

013162  016667  000004  000656 SV05P:  MOV     4(SP),SAVPC

                                ;SAVE R0-R5

013170  010567  000646          SV05:   MOV     R5,SAVR5
013174  010467  000640                  MOV     R4,SAVR4                                ; 3
013200  010367  000632                  MOV     R3,SAVR3
013204  010267  000624                  MOV     R2,SAVR2
013210  010167  000616                  MOV     R1,SAVR1
013214  010067  000610                  MOV     R0,SAVR0
013220  000002                          RTI
  3 013222                      .RESREG

                                ;RESTORE R0-R5

013222  016700  000602          RS05:   MOV     SAVR0,R0
013226  016701  000600                  MOV     SAVR1,R1
013232  016702  000576                  MOV     SAVR2,R2
013236  016703  000574                  MOV     SAVR3,R3
013242  016704  000572                  MOV     SAVR4,R4
013246  016705  000570                  MOV     SAVR5,R5
013252  000002                          RTI
```

```
      1 013254                          .TYPER

                                        ;TELETYPE OUTPUT ROUTINE

   013254  017605  000000        TYPER:  MOV     @(SP),R5                ; 3
   013260  062716  000002                ADD     #2,(SP)
   013264  105777  000462        1$:     TSTB    @TPCSR
   013270  100375                        BPL     1$
   013272  105715                        TSTB    (R5)
   013274  001001                        BNE     2$
   013276  000002                        RTI
   013300  112577  000450        2$:     MOVB    (R5)+,@TPDBR
   013304  000767                        BR      1$
      2 013306                          .INSTRG

                                        ;ASCII STRING INPUT ROUTINE

   013306  017667  000000  000006 INSTRG: MOV     @(SP),MSG
   013314  062716  000002                ADD     #2,(SP)
   013320  104401                INSTR1: TYPE
   013322  000000                MSG:    0
   013324  012704  014634                MOV     #INBUF,R4
   013330  012703  000007                MOV     #7,R3
   013334  105777  000406        1$:     TSTB    @TKCSR
   013340  100375                        BPL     1$
   013342  117714  000402                MOVB    @TKDBR,(R4)
   013346  142714  000200                BICB    #200,(R4)
   013352  122427  000015                CMPB    (R4)+,#15
   013356  001413                        BEQ     INSTR2
   013360  117777  000364  000366        MOVB    @TKDBR,@TPDBR
   013366  105777  000360        2$:     TSTB    @TPCSR
   013372  100375                        BPL     2$
   013374  005303                        DEC     R3
   013376  001356                        BNE     1$
   013400  104401                INSTRE: TYPE
   013402  014364                        MQM
   013404  000745                        BR      INSTR1
   013406  000002                INSTR2: RTI
```

```
1 013410                        .PARAMS

                                ;CONVERT ASCII STRING TO OCTAL                  . 3

013410  011605                 PARAMS: MOV     (SP),R5
013412  012567  000146                 MOV     (R5)+,LOLIM
013416  012567  000144                 MOV     (R5)+,HILIM
013422  012567  000142                 MOV     (R5)+,DEVADR
013426  112567  000140                 MOVB    (R5)+,LOBITS
013432  112567  000135                 MOVB    (R5)+,ADRCNT
013436  010516                        MOV     R5,(SP)
013440  005005                 PARAM1: CLR     R5
013442  012704  014634                 MOV     #INBUF,R4
013446  122714  000015                 CMPB    #15,(R4)
013452  001420                        BEQ     PARERR
013454  121427  000060         1$:     CMPB    (R4),#60
013460  002415                        BLT     PARERR
013462  121427  000067                 CMPB    (R4),#67
013466  003012                        BGT     PARERR
013470  142714  000060                 BICB    #60,(R4)
013474  152405                        BISB    (R4)+,R5
013476  122714  000015                 CMPB    #15,(R4)
013502  001406                        BEQ     LIMITS
013504  006305                        ASL     R5
013506  006305                        ASL     R5
013510  006305                        ASL     R5
013512  000760                        BR      1$
013514  104404                 PARERR: INSTER
013516  000750                        BR      PARAM1

                                ;TEST TO SEE IF NUMBER IS WITHIN LIMITS

013520  020567  000042         LIMITS: CMP     R5,HILIM
013524  101373                        BHI     PARERR
013526  020567  000032                 CMP     R5,LOLIM
013532  103770                        BLO     PARERR
013534  136705  000032                 BITB    LOBITS,R5                       ; 3
013540  001365                        BNE     PARERR

                                ;STORE NUMBER AT SPECIFIED ADDRESS

013542  016704  000022                 MOV     DEVADR,R4
013546  010524                 1$:     MOV     R5,(R4)+
013550  062705  000002                 ADD     #2,R5
013554  105367  000013                 DECB    ADRCNT
013560  001372                        BNE     1$
013562  000002                        RTI
013564  000000                 LOLIM:  0
013566  000000                 HILIM:  0
013570  000000                 DEVADR: 0
013572  000000                 LOBITS: 0
        013573                 ADRCNT=LOBITS+1
```

```
013574                          .OCTASC

                                ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER

013574  017601  000000  OCTASN: MOV     @(SP),R1                        ; 5
013600  062716  000002          ADD     #2,(SP)
013604  012167  000130          MOV     (R1)+,WRDCNT
013610  112167  000126  1$:     MOVB    (R1)+,CHRCNT
013614  112167  000123          MOVB    (R1)+,SPACNT                    ; 3
013620  013167  000120          MOV     @(R1)+,BINWRD
013624  016704  000114  2$:     MOV     BINWRD,R4
013630  116705  000106          MOVB    CHRCNT,R5
013634  012700  014646          MOV     #TEMP,R0
013640  010403          3$:     MOV     R4,R3
013642  042703  177770          BIC     #177770,R3
013646  062703  000260          ADD     #260,R3
013652  110320                  MOVB    R3,(R0)+
013654  006204                  ASR     R4
013656  006204                  ASR     R4
013660  006204                  ASR     R4
013662  005305                  DEC     R5
013664  001365                  BNE     3$
013666  012703  014660          MOV     #MDATA,R3
013672  114023          4$:     MOVB    -(R0),(R3)+
013674  105367  000042          DECB    CHRCNT
013700  001374                  BNE     4$
013702  105767  000035          TSTB    SPACNT
013706  001405                  BEQ     6$
013710  112723  000240  5$:     MOVB    #240,(R3)+
013714  105367  000023          DECB    SPACNT
013720  001373                  BNE     5$
013722  105013          6$:     CLRB    (R3)
013724  104401                  TYPE
013726  014660                  MDATA
013730  005367  000004          DEC     WRDCNT
013734  001325                  BNE     1$
013736  000002                  RTI
013740  000000          WRDCNT: 0
013742  000000          CHRCNT: 0
        013743          SPACNT=CHRCNT+1
013744  000000          BINWRD: 0
```

```
                                013746                  .POINT  ↑/DHSCR,DHNRC,DHLPR,DHBA,DHBC,DHBAR,DHBCR,DHSSR,DHSLR,DHRVEC,DHRLVL,DHTVEC,DHTLVL/
                                                        ;INDIRECT POINTERS                                                                      ; 3

                        013746  177560                  TKCSR:  177560
                        013750  177562                  TKDBR:  177562
                        013752  177564                  TPCSR:  177564
                        013754  177566                  TPDBR:  177566
                                                        .IRP    A       <DHSCR,DHNRC,DHLPR,DHBA,DHBC,DHBAR,DHBCR,DHSSR,DHSLR,DHRVEC,DHRLVL,DHTVEC,DH
TLVL>
                                                        A:      0
                                                        .ENDM
                        013756  000000                  DHSCR:  0
                        013760  000000                  DHNRC:  0
                        013762  000000                  DHLPR:  0
                        013764  000000                  DHBA:   0
                        013766  000000                  DHBC:   0
                        013770  000000                  DHBAR:  0
                        013772  000000                  DHBCR:  0
                        013774  000000                  DHSSR:  0
                        013776  000000                  DHSLR:  0
                        014000  000000                  DHRVEC: 0
                        014002  000000                  DHRLVL: 0
                        014004  000000                  DHTVEC: 0
                        014006  000000                  DHTLVL: 0
                      2 014010                          .VARIA  ↑/TCONST,TIME1,TIME2,TEMP1,TEMP2/
                                                        ;PROGRAM VARIABLES

                        014010  000000                  ERRFLG: 0               ;ERROR FLAG
                        014012  000000                  PASCNT: 0               ;PASS COUNT
                        014014  000000                  ERRCNT: 0               ;ERROR COUNT
                        014016  000000                  RETRN:  0               ;SCOPE RETURN ADDRESS FOR TEST LOOPING
                        014020  000000                  ESCAPE: 0               ;ADDRESS FOR ERROR ESCAPE
                        014022  000000                  FREEZ1: 0               ;DATA LOOPING RETURN ADDRESS
                        014024  000000                  ICOUNT: 0               ;ITERATION COUNT FOR TEST IN PROGRESS
                        014026  000000                  LPCNT:  0               ;NUMBER OF ITERATIONS THIS TEST
                        014030  000000                  SAVR0:  0               ;R0 SAVE AREA
                        014032  000000                  SAVR1:  0               ;R1 SAVE AREA
                        014034  000000                  SAVR2:  0               ;R2 SAVE AREA                                                   ; 3
                        014036  000000                  SAVR3:  0               ;R3 SAVE ARE
                        014040  000000                  SAVR4:  0               ;R4 SAVE AREA
                        014042  000000                  SAVR5:  0               ;R5 SAVE AREA
                        014044  000000                  SAVSP:  0               ;STACK POINTER SAVE AREA
                        014046  000000                  SAVPC:  0               ;CALLING ROUTINE SAVE AREA
                        014050  000000                  INIFLG: 0               ;PROGRAM INITIALIZATION FLAG
                        014052  000000                  STFLG:  0               ;PROGRAM START FLAG
                        014054  000000                  LAST:   0               ;LAST ERROR PC
                                                        .IRP    A       <TCONST,TIME1,TIME2,TEMP1,TEMP2>
                                                        A:      0
                                                        .ENDM
                        014056  000000                  TCONST: 0
                        014060  000000                  TIME1:  0
                        014062  000000                  TIME2:  0
                        014064  000000                  TEMP1:  0
                        014066  000000                  TEMP2:  0
```

```
      1 014070                          .PFAIL
                                        ;ENTER HERE ON POWER FAILURE

 014070  010046                 PFAIL:  MOV     R0,-(SP)            ;SAVE R0-R5 ON PROCESSOR STACK
 014072  010146                         MOV     R1,-(SP)
 014074  010246                         MOV     R2,-(SP)
 014076  010346                         MOV     R3,-(SP)
 014100  010446                         MOV     R4,-(SP)
 014102  010546                         MOV     R5,-(SP)
 014104  016746  163714                 MOV     24,-(SP)
 014110  010667  177730                 MOV     SP,SAVSP            ;SAVE STACK POINTER
 014114  012767  014126  163702         MOV     @RESTART,24         ;SET UP FOR POWER UP TRAP       ; 3
 014122  000000                         HALT                        ;HALT ON POWER DOWN NORMAL
 014124  000777                         BR      .

                                        ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED

 014126  016706  177712         RESTAR: MOV     SAVSP,SP            ;RESTORE STACK POINTER
 014132  012605                         MOV     (SP)+,R5            ;RESTORE R0-R5
 014134  012604                         MOV     (SP)+,R4
 014136  012603                         MOV     (SP)+,R3
 014140  012602                         MOV     (SP)+,R2
 014142  012601                         MOV     (SP)+,R1
 014144  012600                         MOV     (SP)+,R0
 014146  012767  014070  163650         MOV     @PFAIL,24           ;SET UP FOR POWER FAILURE
 014154  012767  000340  163614         MOV     @340,PS
 014162  012706  015120                 MOV     @STACK,SP
 014166  005067  000454                 CLR     TEMP
 014172  005267  000450                 INC     TEMP
 014176  001375                         BNE     .-4
 014200  104401                         TYPE                                            ; 5
 014202  014370                         MCRLF                                           ; 5
 014204  104402                         OCTASC
 014206  014230                         PFTAB
 014210  104401                         TYPE
 014212  014373                         MPFAIL
 014214  005067  177570                 CLR     ERRFLG
 014220  005067  177630                 CLR     LAST
 014224  000177  177566                 JMP     @RETRN
 014230  000001                 PFTAB:  1
 014232  000006  000002                 6.2
 014236  014016                         RETRN
```

```
                                        .MSG    ↑/DH11 SPEED SELECTION LOGIC TEST/,↑/CZDHD-DO/
014240  015   012   012   MTITLE: .ASCIZ  <15><12><12>/DH11 SPEED SELECTION LOGIC TEST /<15><12>
014243  104   110   061
014246  061   040   123
014251  120   105   105
014254  104   040   123
014257  105   114   105
014262  103   124   111
014265  117   116   040
014270  114   117   107
014273  111   103   040
014276  124   105   123
014301  124   040   015
014304  012   000
014306  015   012   126   MVECTO: .ASCIZ  <15><12>/VECTOR ADDRESS-/
014311  105   103   124
014314  117   122   040
014317  101   104   104
014322  122   105   123
014325  123   055   000
014330  015   012   103   MREGAD: .ASCIZ  <15><12>/CONTROL REGISTER ADDRESS-/
014333  117   116   124
014336  122   117   114
014341  040   122   105
014344  107   111   123
014347  124   105   122
014352  040   101   104
014355  104   122   105
014360  123   123   055
014363  000
014364  040   040   077   MQM:    .ASCIZ  /  ?/
014367  000
014370  015   012   000   MCRLF:  .ASCIZ   <15><12>

014373  040   040   120   MPFAIL: .ASCIZ  /  POWER FAILURE, PROGRAM RESTART AT TEST IN PROGRESS/
014376  117   127   105
014401  122   040   106
014404  101   111   114
014407  125   122   105
014412  054   040   120
014415  122   117   107
014420  122   101   115
014423  040   122   105
014426  123   124   101
014431  122   124   040
014434  101   124   040
014437  124   105   123
014442  124   040   111
014445  116   040   120
014450  122   117   107
014453  122   105   123
014456  123   000
014460  015   012   103   MEPASS: .ASCIZ  <15><12>/CZDHD-DO/
014463  132   104   110
014466  104   055   104
014471  060   000
014473  015   012   120   PASTXT: .ASCIZ  <15><12>/PASS COUNT = /                                  ; 5
```

```
        014476   101   123   123
        014501   040   103   117
        014504   125   116   124
        014507   040   075   040
        014512   000
        014513   015   012   122  MR:      .ASCIZ  <15><12>/R/
        014516   000
        014517   015   012   124  MTSTPC:  .ASCIZ  <15><12>/TEST PC-/
        014522   105   123   124
        014525   040   120   103
        014530   055   000
                                        .EVEN
      2 014532   116   117   040  EM1:     .ASCIZ  /NO CLOCK/<15><12>/LINE  SPEED/
        014535   103   114   117
        014540   103   113   015
        014543   012   114   111
        014546   116   105   040
        014551   040   123   120
        014554   105   105   104
        014557   000
      3 014560   124   111   115  EM2:     .ASCIZ  /TIMING ERROR/<15><12>/LINE  SPEED/
        014563   111   116   107
        014566   040   105   122
        014571   122   117   122
        014574   015   012   114
        014577   111   116   105
        014602   040   040   123
        014605   120   105   105
        014610   104   000
      4
      5 014612                     .EVEN
                                   .TRPTAB

                                           ;TABLE OF POINTERS FOR TRAP DECODING

        014612  012600             TRPTAB: SCOPER
        014614  013254                     TYPER
        014616  013574                     OCTASN
        014620  013306                     INSTRG
        014622  013400                     INSTRE
        014624  013410                     PARAMS
        014626  013162                     SV05P
        014630  013222                     RS05
        014632  012700                     SCOP1R
      6 014634                     .BUFFER

                                           ;BUFFERS FOR INPUT-OUTPUT

        014634  000000             INBUF:  0
                014646                     .=.+10
        014646  000000             TEMP:   0
                014660                     .=.+10

        014660  000000             MDATA:  0
                014672                     .=.+10
      7 014672                     .ERRTAB

                                           ;TABLE OF POINTERS TO ERROR MESSAGES AND DATA
```

```
    014672                          ERRTAB:
 8  014672  000000                          0                       ;NO MESSAGE
 9  014674  000000                          0                       ;NO DATA
10  014676  014532                  ET1:    EM1                     ;NO CLOCK ERROR
11  014700  014706                          DT1
12  014702  014560                  ET2:    EM2                     ;TIMING ERROR
13  014704  014706                          DT1
14
15                                          ;DATA TABLES FOR ERROR OUTPUT
16
17  014706  000002                  DT1:    2                       ;2 DATA WORDS WILL BE TYPED
18  014710     002      004         .BYTE   2,4                     ;TWO DIGITS, 4 SPACES
19  014712  014042                          SAVR5           ;LINE UNDER TEST
20  014714     002      000         .BYTE   2,0                     ;TWO DIGITS, NO SPACES
21  014716  014040                          SAVR4                   ;SELECTED SPEED
22  014720                          .ENDCOD
    014720  000000                  ENDCOD: 0
23          000001                  .END
```

SYMBOL TABLE

```
ADRCNT= 013573        EOP     012500        MTSTPC  014517        SPACNT= 013743        T10     003376
BEGIN   001206        ERRCNT  014014        MVECTO  014306        STACK = 015120        T11     003620
BINWRD  013744        ERRFLG  014010        N       000001        START   001004        T12     004042
BITX  = 000000        ERRMSG  013036        OCTASC= 104402        STFLG   014052        T13     004264
BIT00 = 000001        ERRORS  012716        OCTASN  013574        SV05    013170        T14     004506
BIT01 = 000002        ERRTAB  014672        PARAM = 104405        SV05P   013162        T15     004730
BIT02 = 000004        ERTAB0  013122        PARAMS  013410        SWR     001000        T16     005152
BIT03 = 000010        ESCAPE  014020        PARAM1  013440        SW00  = 000001        T17     005374
BIT04 = 000020        ET1     014676        PARERR  013514        SW01  = 000002        T2      001622
BIT05 = 000040        ET2     014702        PASARG  012572        SW02  = 000004        T20     005616
BIT06 = 000100        EXITER  013100        PASCNT  014012        SW03  = 000010        T21     006040
BIT07 = 000200        FREEZ1  014022        PASTXT  014473        SW04  = 000020        T22     006262
BIT08 = 000400        HALTS   013060        PFAIL   014070        SW05  = 000040        T23     006504
BIT09 = 001000        HILIM   013566        PFTAB   014230        SW06  = 000100        T24     006726
BIT10 = 002000        ICOUNT  014024        POPR0 = 012600        SW08  = 000400        T25     007150
BIT11 = 004000        INBUF   014634        POP1SP= 005726        SW09  = 001000        T26     007372
BIT12 = 010000        INIFLG  014050        POP2SP= 022626        SW10  = 002000        T27     007614
BIT13 = 020000        INSTER= 104404        PS    = 177776        SW11  = 004000        T3      002044
BIT14 = 040000        INSTR = 104403        PUSHR0= 010046        SW12  = 010000        T30     010036
BIT15 = 100000        INSTRE  013400        PUSH1S= 005746        SW13  = 020000        T31     010260
CHRCNT  013742        INSTRG  013306        PUSH2S= 024646        SW14  = 040000        T32     010502
DATABP  013054        INSTR1  013320        RESREG  013056        SW15  = 100000        T33     010724
DEVADR  013570        INSTR2  013406        RESTAR  014126        TCONST  014056        T34     011146
DHBA    013764        LAST    014054        RESTRT  012566        TEMP    014646        T35     011370
DHBAR   013770        LIGHTS  001002        RES05 = 104407        TEMP1   014064        T36     011612
DHBC    013766        LIMITS  013520        RETRN   014016        TEMP2   014066        T37     012034
DHBCR   013772        LINE  = 000020        RS05    013222        TIME1   014060        T4      002266
DHLPR   013762        LOBITS  013572        SAVPC   014046        TIME2   014062        T40     012256
DHNRC   013760        LOGICA  012556        SAVR0   014030        TKCSR   013746        T5      002510
DHRLVL  014002        LOLIM   013564        SAVR1   014032        TKDBR   013750        T6      002732
DHRVEC  014000        LPCNT   014026        SAVR2   014034        TPCSR   013752        T7      003154
DHSCR   013756        MCRLF   014370        SAVR3   014036        TPDBR   013754        VEC1    001164
DHSLR   013776        MDATA   014660        SAVR4   014040        TRPOK   013142        VEC2    001174
DHSSR   013774        MEPASS  014460        SAVR5   014042        TRPSRV  013130        WRDCNT  013740
DHTLVL  014006        MPFAIL  014373        SAVSP   014044        TRPTAB  014612        X     = 000000
DHTVEC  014004        MQM     014364        SAV05P= 104406        TYPDAT  013040        XBIT  = 000000
DT1     014706        MR      014513        SCOPE = 104400        TYPE  = 104401        XLINE = 000020
EM1     014532        MREGAD  014330        SCOPER  012600        TYPER   013254        XN    = 000041
EM2     014560        MSG     013322        SCOPE1= 104410        TYPMSG  013016        Y     = 000011
ENDCOD  014720        MTITLE  014240        SCOP1R  012700        T1      001400

. ABS.  014722      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  17664 WORDS  ( 69 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  71 PAGES
CZDHDD.BIN,CZDHDD.SEQ=CZDHDD.DOC,DHMACA.MAC,CZDHDD.P11
```