DH11

BREAK&HALF DOP

MD-11-DZDHI-B

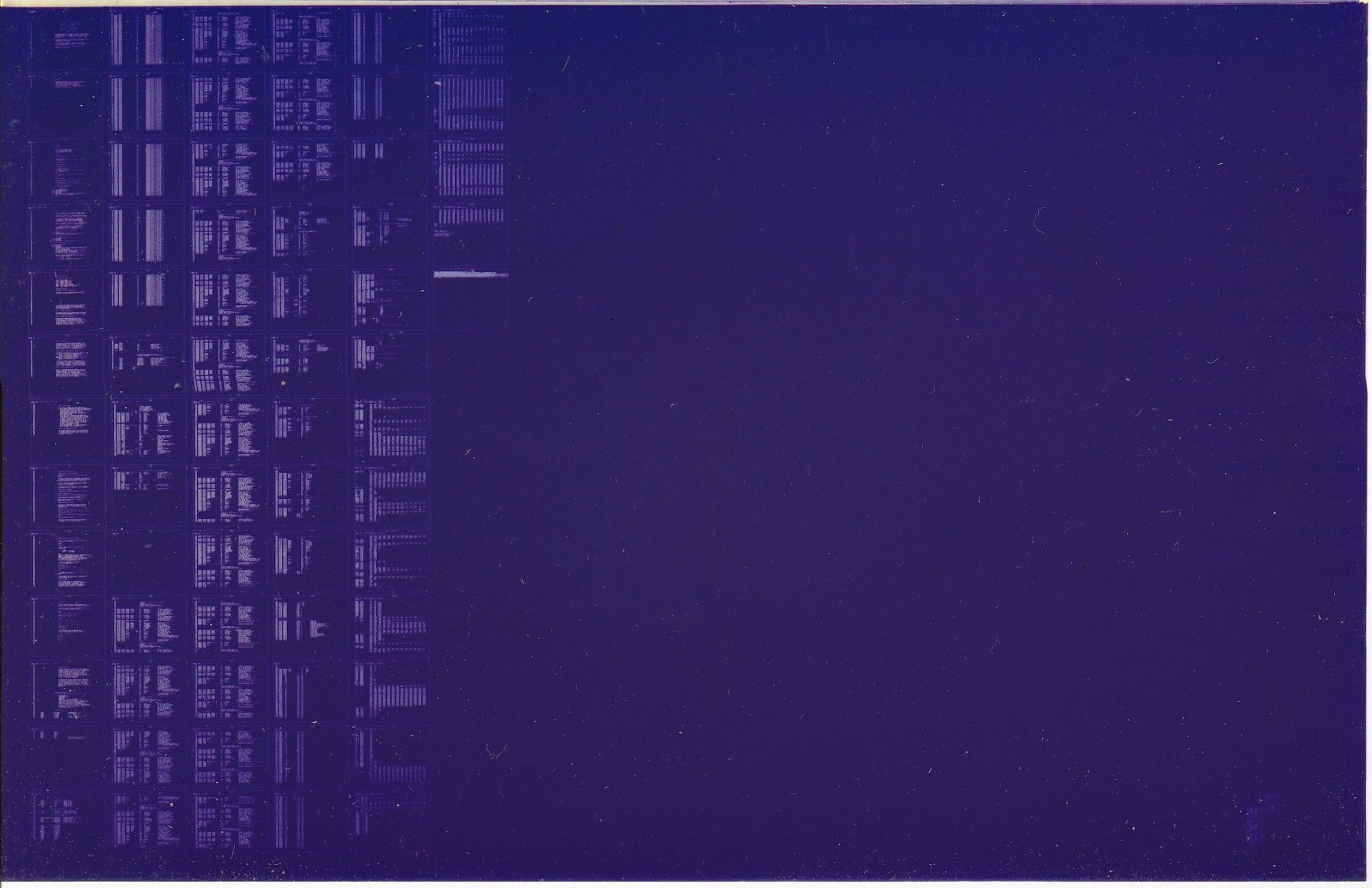
EP-DZDHI-B-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN U.S.A



```
                                .REM !
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DZDHI-B-D
PRODUCT NAME:	DH11 BREAK AND HALF-DUPLEX TEST
DATE:	MAY 1976
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	MICHAEL DAVIS

```
                                1.      ABSTRACT

                                        THE DH11 BREAK AND HALF DUPLEX TEST CHECKS THE BREAK
                                        CONTROL LOGIC OF THE DH11 AND VERIFIES THAT THE UARTS
                                        RECEIVE ONLY ONE BREAK CHARACTER  ON A GIVEN
                                        LINE NO MATTER HOW LONG BREAK IS ASSERTED.
                                        THE TEST ALSO VERIFIES THAT NO CHARACTERS ARE
                                        RECEIVED ON A A LINE IF THE HALF DUPLEX FUNCTION FOR
                                        THAT LINE IS SELECTED
```

```
                                2.      REQUIREMENTS

                                2.1     EQUIPMENT

                                        PDP-11 FAMILY STANDARD COMPUTER WITH 4KW OF MEMORY
                                        ASR-33 TELETYPE OR EQUIVALENT
                                        DH11 ASYNCHRONOUS MULTIPLEXER
                                        DH11 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 BREAK AND HALF-DUPLEX 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 BREAK AND HALF-DUPLEX 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 BREAK AND HALF-DUPLEX 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
```

```
                                5.      OPERATING PROCEDURE

                                5.1     OPERATIONAL SWITCH SETTINGS

                                        SW15=1, HALT ON ERROR
                                        SW14=1, LOOP ON CURRENT TEST
                                        SW13=1, SUPPRESS ERROR TYPEOUT
                                        SW11=1, INHIBIT ITERATIONS
                                        SW10=1, ESCAPE TO NEXT TEST ON ERROR
                                        SW09=1, FREEZE VARIABLE PARAMETER IN CURRENT TEST
                                        SW01=1, START PROGRAM AT SELECTED TEST
                                        SW00=1, CHANGE PARAMETERS AT PROGRAM RESTART

                                5.2     SUBROUTINE ABSTRACTS

                                5.2.1   TRAPCATCHER (LOCATIONS 000000-000776)

                                        THIS ROUTINE IS USED TO INTERCEPT UNEXPECTED INTERRUPTS
                                        AND TRAPS.  THE AREA FROM 000000-000776 IS LOADED
                                        WITH THE FOLLOWING SEQUENCE

                                                2
                                                0
                                                4
                                                0
                                                ...
                                                772
                                                0
                                                776
                                                0

                                        IF AN UNEXPECTED INTERRUPT OR TRAP OCCURS, THE PROGRAM
                                        WILL HALT WITH THE PC 2 GREATER THAN THE ADDRESS TO
                                        WHICH THE PROGRAM TRAPPED.  THE PROCESSOR STACK MAY
                                        BE EXAMINED TO DETERMINE WHERE THE PROGRAM WAS WHEN THE
                                        TRAP OR INTERRUPT OCCURED.

                                5.2.2   START   (PROGRAM INITIALIZATION)

                                        THIS ROUTINE INITIALIZES ALL PROGRAM FLAGS AND COUNTERS,
                                        TYPES THE PROGRAM TITLE MESSAGE, AND INPUTS THE VECTOR
                                        AND CONTROL REGISTER ADDRESSES OF THE DH11 TO BE TESTED.

                                5.2.3   BEGIN   (PROGRAM START AND RESTART)

                                        THIS ROUTINE IS ENTERED IMMEDIATLY AFTER "START"
                                        AND EACH TIME A PROGRAM PASS HAS BEEN COMPLETED.
                                        THE ROUTINE SETS UP THE PROCESSOR STACK AND STATUS WORD
                                        AND THEN TRANSFERS CONTROL TO THE TEST AT WHICH
                                        TESTING WILL BEGIN.  IF SW01=0 WHEN THIS ROUTINE IS ENTERD
                                        TESTING WILL START AT T1 (TEST 1).  IF SW01=1 WHEN
                                        THIS ROUTINE IS ENTERED, TESTING WILL START AT THE PC
                                        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 SCOPIR (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.

```
297
298                                     5.3     PROGRAM AND OR OPERATOR ACTION
299
300                                     5.3.1   PROGRAM START WITH ALL SWITCHES DOWN
301
302                                     5.3.1.1 REFER TO SECTIONS 4.3.1 AND 4.3.2 FOR INITIAL PROGRAM
303                                             BEHAVIOR.
304
305                                     5.3.1.2 AFTER "R" HAS BEEN TYPED BY THE PROGRAM, TEST EXECUTION
306                                             WILL BEGIN.  EACH TEST WILL BE REPEATED A SELECTED NUMBER
307                                             OF ITERATIONS (SEE LISTING FOR EXACT NUMBER FOR EACH TEST)
308                                             AND THEN THE PROGRAM WILL PROCEED TO THE NEXT TEST.
309
310                                     5.3.1.3 WHEN ALL ITERATIONS HAVE BEEN COMPLETED, THE PROGRAM
311                                             WILL TYPE "DZDHI" AND THEN RESTART TESTING AT TEST 1
312                                             (LOCATION T1 IN THE PROGRAM).
313
314                                     5.3.1.4 IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE
315                                             ERROR MESSAGE, AND THEN CONTINUE THE TEST IN PROGRESS.
316
317                                     5.3.2   PROGRAM START WITH SW00=1
318
319                                             THE PROGRAM WILL PERFORM AS DESCRIBED IN 4.3.1 AND 5.3.1
320
321                                     5.3.3   PROGRAM START WITH SW01=1
322
323                                     5.3.3.1 REFER TO SECTION 4.3.4 FOR INITIAL PROGRAM BEHAVIOR
324
325                                     5.3.3.2 TEST EXECUTION WILL START AT THE ADDRESS SPECIFIED
326                                             AND WILL CONTINUE AS DESCRIBED IN 5.3.1.2
327
328                                     5.3.3.3 AFTER "DZDHI" HAS BEEN TYPED, THE PROGRAM WILL
329                                             RESUME TESTING AT TEST 1
330
331                                     5.3.4   PROGRAM OPERATION WITH SW15=1
332
333                                             SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR,
334                                             THE PROGRAM WILL HALT AFTER THE ERROR TYPEOUT, AND
335                                             THE PC+2 OF THE CALL TO THE ERROR ROUTINE WILL BE
336                                             DISPLAYED IN R0.
337
338                                     5.3.5   PROGRAM OPERATION WITH SW13=1
339
340                                             SAME AS 5.3.1 EXCEPT THAT NO ERROR TYPEOUTS WILL OCCUR
341
342                                     5.3.6   PROGRAM OPERATION WITH SW11=1
343
344                                             SAME AS 5.3.1 EXCEPT THAT EACH TEST WILL BE REPEATED ONCE
345                                             ONLY
346
347                                     5.3.7   PROGRAM OPERATION WITH SW10=1
348
349                                             SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR
350                                             THE CURRENT TEST WILL BE ABORTED, AND THE PROGRAM
351                                             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 DZDHI TO END OF TYPEOUT OF DZDHI)
                                        IS GIVEN FOR VARIOUS PROCESSORS IN THE TABLE BELOW

                                                        TIME
                                        PROCESSOR

                                        PDP-11/05,10

                                        PDP-11/20

                                        PDP-11/40

                                        PDP-11/45
```

```
                                9.      PROGRAM DESCRIPTION

                                        THE FIRST GROUP OF TESTS VERIFIES THAT ONLY ONE BREAK
                                        CHARACTER IS RECEIVED ON A GIVEN LINE, EVEN THOUGH BREAK
                                        IS ASSRETED FOR THAT LINE FOR 400 (OCTAL) CHARACTER TIMES.
                                        THERE IS AN INDIVIDUAL TEST LOOP FOR EACH LINE.
                                        THE TEST BEGINS BY FLUSING EACH UART BY TRANSMITTING
                                        TO NULL (0) CHARACTERS.  THE BREAK BIT IS THEN SET
                                        FOR THE LINE TO BE TESTED, TRANSMISSION OF THE BINARY COUNT
                                        PATTERN IS STARTED.  THE SILO IS THEN CHECKED TO VERIFY
                                        THAT ONLY ONE CHARACTER WAS RECEIVED
                                        AND THAT IT WAS A BREAK CHARACTER.

                                        THE SECOND GROUP OF TESTS VERIFIES THAT NO CHARACTERS
                                        ARE RECEIVED ON A SELECTED LINE IF THE HALF DUPLEX BIT
                                        IS SET FOR THE LINE TO BE TESTED.  THERE IS AN INDIVIDUAL
                                        TEST LOOP FOR EACH LINE TO BE TESTED.  A BINARY COUNT
                                        PATTERN IS THEN TRANSMITTED ON THE LINE TO BE TESTED.
                                        WHEN ALL CHARACTERS HAVE BEEN TRANSMITTED, THE CHARACTER
                                        AVAILABLE FLAG IS TESTED TO DETERMINE IF ANY CHARACTERS HAVE
                                        BEEN RECEIVED.

                                10.     LISTING


                                        !

                                ;DH11 BREAK AND HALF DUPLEX TEST
                                ;COPYRIGHT 1972, 1976, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE DH11 BREAK AND HALF DUPLEX 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>
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE " DZDHI "
                                        ;AND THEN RESUM TESTING


                                ;SWITCH REGISTER OPTIONS

100000                          SW15=100000             ;=1,HALT ON ERROR
040000                          SW14=40000              ;=1,LOOP ON CURRENT TEST
020000                          SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
010000                          SW12=10000
004000                          SW11=4000               ;=1,INHIBIT ITERATIONS
002000                          SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
001000                          SW09=1000               ;=1,LOOP WITH CURRENT DATA
```

```
   509          000400                  SW08=400
   510          000100                  SW06=100
   511          000040                  SW05=40
   512          000020                  SW04=20
   513          000010                  SW03=10
   514          000004                  SW02=4
   515          000002                  SW01=2
   516          000001                  SW00=1
   517
```

```
;RESTART PROGRAM AT SELECTED TEST
;RESELECT VECTOR AND CONTROL REGISTER
;ADDRESS AFTER PROGRAM RESTART
```

```
                                ;REGISTER DEFINITIONS

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

                                ;LOCATION EQUIVALENCIES

177570                          SWR=177570      ;CONSOLE SWITCH REGISTER
177570                          LIGHTS=177570   ;PDP-11/45 DISPLAY REGISTER
177776                          PS=177776       ;PROCESSOR STATUS WORD
014040                          STACK=ENDCOD+200;START OF PROCESSOR STACK

                                ;INSTRUCTION DEFINITIONS

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


100000                          BIT15=100000
040000                          BIT14=40000
020000                          BIT13=20000
010000                          BIT12=10000
004000                          BIT11=4000
002000                          BIT10=2000
001000                          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
```

```
555                                     ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
566                 000000      .=0
567   000000        000002              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
568   000002        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
569   000004        000006              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
570   000006        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
571   000010        000012              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
572   000012        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
573   000014        000016              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
574   000016        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
575   000020        000022              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
576   000022        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
577   000024        000026              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
578   000026        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
579   000030        000032              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
580   000032        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
581   000034        000036              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
582   000036        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
583   000040        000042              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
584   000042        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
585   000044        000046              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
586   000046        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
587   000050        000052              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
588   000052        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
589   000054        000056              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
590   000056        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
591   000060        000062              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
592   000062        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
593   000064        000066              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
594   000066        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
595   000070        000072              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
596   000072        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
597   000074        000076              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
598   000076        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
599   000100        000102              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
600   000102        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
601   000104        000106              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
602   000106        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
603   000110        000112              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
604   000112        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
605   000114        000116              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
606   000116        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
607   000120        000122              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
608   000122        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
609   000124        000126              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
610   000126        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
611   000130        000132              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
612   000132        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
613   000134        000136              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
614   000136        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
615   000140        000142              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
616   000142        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
617   000144        000146              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
618   000146        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
619   000150        000152              .+2             ;UNEXPECTED TRAP TO THIS LOCATION
620   000152        000000              HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
  621  000154  000156                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  622  000156  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  623  000160  000162                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  624  000162  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  625  000164  000166                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  626  000166  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  627  000170  000172                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  628  000172  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  629  000174  000176                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  630  000176  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  631  000200  000202                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  632  000202  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  633  000204  000206                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  634  000206  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  635  000210  000212                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  636  000212  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  637  000214  000216                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  638  000216  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  639  000220  000222                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  640  000222  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  641  000224  000226                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  642  000226  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  643  000230  000232                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  644  000232  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  645  000234  000236                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  646  000236  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  647  000240  000242                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  648  000242  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  649  000244  000246                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  650  000246  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  651  000250  000252                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  652  000252  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  653  000254  000256                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  654  000256  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  655  000260  000262                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  656  000262  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  657  000264  000266                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  658  000266  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  659  000270  000272                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  660  000272  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  661  000274  000276                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  662  000276  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  663  000300  000302                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  664  000302  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  665  000304  000306                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  666  000306  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  667  000310  000312                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  668  000312  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  669  000314  000316                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  670  000316  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  671  000320  000322                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  672  000322  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  673  000324  000326                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  674  000326  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
  675  000330  000332                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
  676  000332  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
```

```
  677  000334  000336                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  678  000336  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  679  000340  000342                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  680  000342  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  681  000344  000346                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  682  000346  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  683  000350  000352                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  684  000352  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  685  000354  000356                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  686  000356  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  687  000360  000362                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  688  000362  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  689  000364  000366                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  690  000366  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  691  000370  000372                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  692  000372  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  693  000374  000376                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  694  000376  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  695  000400  000402                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  696  000402  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  697  000404  000406                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  698  000406  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  699  000410  000412                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  700  000412  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  701  000414  000416                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  702  000416  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  703  000420  000422                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  704  000422  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  705  000424  000426                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  706  000426  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  707  000430  000432                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  708  000432  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  709  000434  000436                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  710  000436  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  711  000440  000442                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  712  000442  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  713  000444  000446                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  714  000446  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  715  000450  000452                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  716  000452  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  717  000454  000456                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  718  000456  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  719  000460  000462                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  720  000462  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  721  000464  000466                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  722  000466  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  723  000470  000472                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  724  000472  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  725  000474  000476                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  726  000476  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  727  000500  000502                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  728  000502  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  729  000504  000506                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  730  000506  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
  731  000510  000512                    .+2         ;UNEXPECTED TRAP TO THIS LOCATION
  732  000512  000000                    HALT        ;EXAMINE STACK TO FIND CAUSE
```

```
733  000514  000516                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
734  000516  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
735  000520  000522                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
736  000522  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
737  000524  000526                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
738  000526  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
739  000530  000532                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
740  000532  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
741  000534  000536                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
742  000536  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
743  000540  000542                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
744  000542  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
745  000544  000546                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
746  000546  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
747  000550  000552                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
748  000552  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
749  000554  000556                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
750  000556  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
751  000560  000562                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
752  000562  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
753  000564  000566                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
754  000566  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
755  000570  000572                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
756  000572  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
757  000574  000576                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
758  000576  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
759  000600  000602                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
760  000602  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
761  000604  000606                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
762  000606  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
763  000610  000612                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
764  000612  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
765  000614  000616                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
766  000616  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
767  000620  000622                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
768  000622  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
769  000624  000626                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
770  000626  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
771  000630  000632                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
772  000632  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
773  000634  000636                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
774  000636  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
775  000640  000642                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
776  000642  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
777  000644  000646                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
778  000646  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
779  000650  000652                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
780  000652  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
781  000654  000656                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
782  000656  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
783  000660  000662                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
784  000662  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
785  000664  000666                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
786  000666  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
787  000670  000672                    .+2     ;UNEXPECTED TRAP TO THIS LOCATION
788  000672  000000                    HALT    ;EXAMINE STACK TO FIND CAUSE
```

```
  789  000674  000676                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  790  000676  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  791  000700  000702                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  792  000702  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  793  000704  000706                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  794  000706  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  795  000710  000712                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  796  000712  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  797  000714  000716                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  798  000716  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  799  000720  000722                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  800  000722  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  801  000724  000726                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  802  000726  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  803  000730  000732                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  804  000732  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  805  000734  000736                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  806  000736  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  807  000740  000742                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  808  000742  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  809  000744  000746                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  810  000746  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  811  000750  000752                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  812  000752  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  813  000754  000756                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  814  000756  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  815  000760  000762                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  816  000762  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  817  000764  000766                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  818  000766  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  819  000770  000772                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  820  000772  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
  821  000774  000776                    .+2             ;UNEXPECTED TRAP TO THIS LOCATION
  822  000776  000000                    HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
DZDHI   MACY11 27(732)  22-MAR-76  10:10  PAGE 20
DZDHI8.PFC

                                        ;STANDARD INTERRUPT VECTORS


                000024          .=24
000024  012756                  PFAIL                   ;POWER FAIL HANDLER
000026  000340                  340                     ;SERVICE AT LEVEL 7
000030  011064                  ERRORS                  ;ERROR HANDLER
000032  000340                  340                     ;SERVICE AT LEVEL 7
000034  011266                  TRPSRV                  ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                  340                     ;SERVICE AT LEVEL 7
                000200          .=200
000200  000167  000574          JMP     START           ;GO TO START OF PROGRAM


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

                104400                  SCOPE=TRAP+Y            ;SCOPE LOOP AND ITERATION HANDLER
                104401                  TYPE=TRAP+Y             ;TELETYPE OUTPUT ROUTINE
                104402                  OCTASC=TRAP+Y           ;OCTAL TO ASCII CONVERSION
                104403                  INSTR=TRAP+Y            ;INPUT ASCII STRING
                104404                  INSTER=TRAP+Y           ;STRING INPUT ERROR
                104405                  PARAM=TRAP+Y            ;CONVERT STRING TO OCTAL, CHECK LIMITS
                104406                  SAVDSP=TRAP+Y           ;SAVE R0-R5, PC
                104407                  RESCS=TRAP+Y            ;RESTORE R0-R5
                104410                  SCOPE1=TRAP+Y           ;CHECK FOR FREEZE ON CURRENT DATA
                000046          .=46
000046  010732                  LOGICAL
                000052          .=52
000052  040000                  40000
```

```
                001000                  .=1000
                                        ;PROGRAM INITIALIZATION
                                        ;LOCK OUT INTERRUPTS
                                        ;SET UP PROCESSOR STACK
                                        ;SET UP POWER FAIL VECTOR
                                        ;CLEAR PROGRAM FLAGS AND COUNTS
                                        ;TYPE TITLE MESSAGE

001000  012767  000340  176770  START:  MOV     #340,PS         ;LOCK OUT INTERRUPTS
001006  012706  014040                  MOV     #STACK,SP       ;SET UP PROCESSOR STACK
001012  012737  012756  000024          MOV     #PFAIL,@#24     ;SET UP POWER FAIL TRAP
001020  005067  011170                  CLR     STFLG           ;CLEAR TEST START FLAG
001024  005067  011124                  CLR     PASCNT          ;CLEAR PASS COUNT
001030  005067  011122                  CLR     ERRCNT          ;CLEAR ERROR COUNT
001034  005067  011112                  CLR     ERRFLG          ;CLEAR ERROR FLAG
001040  005067  011106                  CLR     EPRFLG          ;CLEAR LAST ERROR PC
001044  104401  013122                  TYPE    ,MTITLE         ;TYPE TITLE MESSAGE
001050  005767  011136                  TST     INIFLG          ;CHECK INITIALIZATION FLAG
001054  001001                          BNE     VEC1            ;IF NOT 0, CHECK SWITCHES
                                                                ;FOR REINITIALIZATION
001056  000404                          BR      VEC2
001060  032767  000001  176502  VEC1:   BIT     #SW00,SWR       ;IF SW00=1, GET NEW VECTOR
001066  001445                          BEQ     BEGIN           ;AND CSR
001070  012701  000300          VEC2:   MOV     #300,R1
001074  012702  000302                  MOV     #302,R2
001100  012703  000004                  MOV     #4,R3
001104  010211                  1$:     MOV     R2,(R1)         ;RESTORE TRAPCATCHER
001106  005012                          CLR     (R2)            ;IN FLOATING VECTOR AREA
001110  060301                          ADD     R3,R1
001112  060302                          ADD     R3,R2
001114  020127  001000                  CMP     R1,#1000
001120  001371                          BNE     1$
001122  104403                          INSTR                   ;INPUT ADDRESS OF DEVICE VECTOR
001124  013170                          MVECTOR                 ;MESSAGE "VECTOR ADDRESS-"
001126  104405                          PARAM                   ;CONVERT STRING TO OCTAL
001130  000300                          300                     ;LOW LIMIT
001132  000770                          770                     ;HIGH  LIMIT
001134  012142                          DHRVEC                  ;LOCATIONS TO BE FILLED
001136     003                  .BYTE   3                       ;NUMBER OF LOCATIONS
001137     004                  .BYTE   4                       ;LSB MASK
001140  104403                          INSTR                   ;INPUT ADDRESS OF DEVICE CSR
001142  013212                          MREGAD                  ;MESSAGE "CONTROL REGISTER ADDRESS-"
001144  104405                          PARAM                   ;CONVERT STRING TO OCTAL
001146  000000                          0                       ;LOW LIMIT
001150  177776                          177776                  ;HIGH LIMIT
001152  012120                          DHSCR                   ;LOCATIONS TO BE FILLED
001154     007                  .BYTE   7                       ;NUMBER OF LOCATIONS
001155     010                  .BYTE   10                      ;LSB MASK
001156  016767  010754  010754          MOV     DHSSR,DHSLR     ;SET UP ADDRESS OF SILO
001164  005267  010750                  INC     DHSLR           ;STATUS REGISTER HIGH BYTE
001170  005767  011016                  TST     INIFLG          ;IF INITIALIZATION FLAG
001174  001002                          BNE     BEGIN           ;IS CLEARED
001176  005167  011010                  COM     INIFLG          ;SET IT

                                        ;PROGRAM START
```

```
                                                        ;CHECK FOR PROGRAM START AT SELECTED ADDRESS
                                                        
001202  012767  000340  176566  BEGIN:  MOV     #340,PS         ;LOCK OUT INTERRUPTS
001210  012706  014040                  MOV     #STACK,SP       ;SET UP PROCESSOR STACK
001214  032767  000002  176346          BIT     #SW01,SWR       ;IF SW01=1
001222  001410                          BEQ     1$              ;GET PC FOR PROGRAM START
001224  104403                          INSTR                   ;GET PC
001226  013356                          MTSTPC                  ;MESSAGE "TEST PC"
001230  104405                          PARAM                   ;CONVERT STRING TO OCTAL
001232  000000                          0
001234  017500                          17500
001236  000207                          RETURN
001240     001                  .BYTE   1
001241     001                  .BYTE   1
001242  000410                          BR      2$
001244  012767  001274  010706  1$:     MOV     #T1,RETURN      ;NORMAL START, TEST 1
001252  005767  010736                  TST     STFLG           ;IF LOOPING, BYPASS TYPEOUT
001256  001004                          BNE     3$
001260  005167  010730                  COM     STFLG
001264  104401  013352          2$:     TYPE    ,MR             ;TYPE "R" TO INDICATE START
001270  000177  010664          3$:     JMP     @RETURN         ;START TESTING
```

```
   932
```

```
                                        ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
                                        ;ON LINE 0
                                        ;SET BREAK BIT FOR LINE 0
                                        ;TRANSMIT BINARY COUNT PATTERN ON LINE 0
                                        ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
                                        ;AND THAT IT IS A BREAK
001274  012767  000340  176474  T1:    MOV     #340,PS          ;DISABLE ALL INTERRUPTS
001302  012767  012120  010656         MOV     #DHSCR,ICOUNT    ;SET UP FOR DHSCR ITERATIONS
001310  012767  000020  010644         MOV     #20,ESCAPE       ;SET UP TO ESCAPE TO NEXT TEST
001316  012767  001532  010640         MOV     #4$,FREEZ1       ;SET UP TO LOOP WITH DATA
001324  012777  004000  010566         MOV     #BIT11,@DHSCR    ;MASTER CLEAR INTERFACE
001332  004767  010670                 JSR     PC,CLRALL        ;CLEAR ALL BUS ADDRESS AND
                                                                 ;BUS ADDRESS MEMORY LOCATIONS
001336  012777  000000  010554         MOV     #0,@DHSCR        ;SELECT LINE 0
001344  012777  012222  010554         MOV     #NULL,@DHBA      ;SET UPT TO TRANSMIT 0 CHARACTER
001352  012777  177776  010550         MOV     #-2,@DHBC        ;TWO 0S WILL BE TRANSMITTED
001360  012777  033503  010536         MOV     #33503,@DHLPR    ;SET LINE SPEED=9600 BAUD
                                                                 ;CHARACTER LENGTH =8 BITS
001366  012777  000001  010536         MOV     #1,@DHBAR        ;SET BAR BIT FOR LINE 0
001374  122777  000002  010536  1$:    CMPB    #2,@DHSLR        ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
001402  001374                 BNE     1$
001404  012777  004000  010506         MOV     #BIT11,@DHSCR    ;MASTER CLEAR INTERFACE
001412  012777  000000  010500         MOV     #0,@DHSCR        ;SELECT LINE 0
001420  012777  012254  010500         MOV     #TBUF,@DHBA      ;SET UP TO TRANSMIT 400
001426  012777  177400  010474         MOV     #-400,@DHBC      ;(OCTAL) CHARACTERS
001434  012777  033503  010462         MOV     #33503,@DHLPR    ;LINE SPEED = 9600 BAUD
001442  012777  000001  010464         MOV     #1,@DHBCR        ;SET BREAK BIT FOR LINE 0
001450  012777  000001  010454         MOV     #1,@DHBAR        ;SET BAR BIT FOR LINE 0
001456  005777  010450          2$:    TST     @DHBAR           ;WAIT FOR ALL CHARACTERS
001462  001375                 BNE     2$                       ;TO BE TRANSMITTED
001464  122777  000001  010446         CMPB    #1,@DHSLR        ;CHECK TO SEE THAT ONLY
001472  001407                 BEQ     3$                       ;1 CHARACTER WAS RECEIVED
001474  017704  010436                 MOV     @DHSSR,R4        ;(R4)=ACTUAL RECEIVED DATA
001500  042704  000300                 BIC     #300,R4          ;CLEAR UNWANTED BITS
001504  012705  000400                 MOV     #400,R5          ;(R5)=EXPECTED SILO FILL LEVEL, 1
001510  104000                 HLT     0                        ;MORE THAN ONE CHARACTER RECEIVED, ERROR
001512  017704  010404          3$:    MOV     @DHNRC,R4        ;READ NEXT RECEIVED CHARACTER REGISTER
001516  026704  011174                 CMP     RWRDO,R4                 ;IS RECEIVED CHARACTER A BREAK
001522  001403                 BEQ     4$
001524  016705  011156                 MOV     RWRDO,R5         ;(R5)=EXPECTED RECEIVED CHARACTER
001530  104001                 HLT     1                        ;RECEIVED DATA ERROR
001532  104400          4$:    SCOPE                            ;CHECK FOR ITERATIONS, LOOP

                                        ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
                                        ;ON LINE 1
                                        ;SET BREAK BIT FOR LINE 1
                                        ;TRANSMIT BINARY COUNT PATTERN ON LINE 1
                                        ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
                                        ;AND THAT IT IS A BREAK
001534  012767  000340  176234  T2:    MOV     #340,PS          ;DISABLE ALL INTERRUPTS
001542  012767  012120  010416         MOV     #DHSCR,ICOUNT    ;SET UP FOR DHSCR ITERATIONS
001550  012767  000020  010404         MOV     #20,ESCAPE       ;SET UP TO ESCAPE TO NEXT TEST
001556  012767  001772  010400         MOV     #4$,FREEZ1       ;SET UP TO LOOP WITH DATA
```

```
  989  001564  012777  004000  010326          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
  990  001572  004767  010430                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
  991                                                                          ;BUS ADDRESS MEMORY LOCATIONS
  992  001576  012777  000001  010314          MOV     #1,@DHSCR               ;SELECT LINE 1
  993  001604  012777  012222  010314          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
  994  001612  012777  177776  010310          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
  995  001620  012777  033503  010276          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
  996                                                                          ;CHARACTER LENGTH =8 BITS
  997  001626  012777  000002  010276          MOV     #2,@DHBAR               ;SET BAR BIT FOR LINE 1
  998  001634  122777  000002  010276  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
  999  001642  001374                          BNE     1$
 1000  001644  012777  004000  010246          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 1001  001652  012777  000001  010240          MOV     #1,@DHSCR               ;SELECT LINE 1
 1002  001660  012777  012254  010240          MOV     #TBUF,@DHBA             ;SET UP TO TRANSM.  400
 1003  001666  012777  177400  010234          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
 1004  001674  012777  033503  010222          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
 1005  001702  012777  000002  010224          MOV     #2,@DHBCR               ;SET BREAK BIT FOR LINE 1
 1006  001710  012777  000002  010214          MOV     #2,@DHBAR               ;SET BAR BIT FOR LINE 1
 1007  001716  005777  010210          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
 1008  001722  001375                          BNE     2$                      ;TO BE TRANSMITTED
 1009  001724  122777  000001  010206          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
 1010  001732  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
 1011  001734  017704  010176                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
 1012  001740  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
 1013  001744  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
 1014  001750  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
 1015  001752  017704  010144          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
 1016  001756  026704  010736                  CMP     RWRD1,R4                ;IS RECEIVED CHARACTER A BREAK
 1017  001762  001403                          BEQ     4$
 1018  001764  016705  010730                  MOV     RWRD1,R5                ;(R5)=EXPECTED RECEIVED CHARACTER
 1019  001770  104001                          HLT     1                       ;RECEIVED DATA ERROR
 1020  001772  104400          4$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
 1021
 1022                                  ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
 1023                                  ;ON LINE 2
 1024                                  ;SET BREAK BIT FOR LINE 2
 1025                                  ;TRANSMIT BINARY COUNT PATTERN ON LINE 2
 1026                                  ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
 1027                                  ;AND THAT IT IS A BREAK
 1028
 1029  001774  012767  000340  175774  T3:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
 1030  002002  012767  012120  010156          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
 1031  002010  012767  000020  010144          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
 1032  002016  012767  002232  010140          MOV     #4$,FREEZ1              ;SET UP TO LOOP WITH DATA
 1033  002024  012777  004000  010066          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 1034  002032  004767  010170                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
 1035                                                                          ;BUS ADDRESS MEMORY LOCATIONS
 1036  002036  012777  000002  010054          MOV     #2,@DHSCR               ;SELECT LINE 2
 1037  002044  012777  012222  010054          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
 1038  002052  012777  177776  010050          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
 1039  002060  012777  033503  010036          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
 1040                                                                          ;CHARACTER LENGTH =8 BITS
 1041  002066  012777  000004  010036          MOV     #4,@DHBAR               ;SET BAR BIT FOR LINE 2
 1042  002074  122777  000002  010036  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
 1043  002102  001374                          BNE     1$
 1044  002104  012777  004000  010006          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
```

```
1045  002112  012777  000002  010000          MOV     #2,@DHSCR               ;SELECT LINE 2
1046  002120  012777  012254  010000          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1047  002126  012777  177400  007774          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1048  002134  012777  033503  007762          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1049  002142  012777  000004  007764          MOV     #4,@DHBCR               ;SET BREAK BIT FOR LINE 2
1050  002150  012777  000004  007754          MOV     #4,@DHBAR               ;SET BAR BIT FOR LINE 2
1051  002156  005777  007750          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1052  002162  001375                          BNE     2$                      ;TO BE TRANSMITTED
1053  002164  122777  000001  007746          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1054  002172  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1055  002174  017704  007736                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1056  002200  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1057  002204  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1058  002210  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1059  002212  017704  007704          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1060  002216  026704  010500                  CMP     RWRD2,R4                        ;IS RECEIVED CHARACTER A BREAK
1061  002222  001403                          BEQ     4$
1062  002224  016705  010472                  MOV     RWRD2,R5                ;(R5)=EXPECTED RECEIVED CHARACTER
1063  002230  104001                          HLT     1                       ;RECEIVED DATA ERROR
1064  002232  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1065
1066                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1067                                          ;ON LINE 3
1068                                          ;SET BREAK BIT FOR LINE 3
1069                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 3
107C                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1071                                          ;AND THAT IT IS A BREAK
1072
1073  002234  012767  000340  175534  T4:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1074  002242  012767  012120  007716          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1075  002250  012767  000020  007704          MOV     #2C,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1076  002256  012767  002472  007700          MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
1077  002264  012777  004000  007626          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1078  002272  004767  007730                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1079                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1080  002276  012777  000003  007614          MOV     #3,@DHSCR               ;SELECT LINE 3
1081  002304  012777  012222  007614          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1082  002312  012777  177776  007610          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1083  002320  012777  033503  007576          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1084                                                                          ;CHARACTER LENGTH =8 BITS
1085  002326  012777  000010  007576          MOV     #10,@DHBAR              ;SET BAR BIT FOR LINE 3
1086  002334  122777  000002  007576  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1087  002342  001374                          BNE     1$
1088  002344  012777  004000  007546          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1089  002352  012777  000003  007540          MOV     #3,@DHSCR               ;SELECT LINE 3
1090  002360  012777  012254  007540          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1091  002366  012777  177400  007534          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1092  002374  012777  033503  007522          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1093  002402  012777  000010  007524          MOV     #10,@DHBCR              ;SET BREAK BIT FOR LINE 3
1094  002410  012777  000010  007514          MOV     #10,@DHBAR              ;SET BAR BIT FOR LINE 3
1095  002416  005777  007510          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1096  002422  001375                          BNE     2$                      ;TO BE TRANSMITTED
1097  002424  122777  000001  007506          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1098  002432  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1099  002434  017704  007476                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1100  002440  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
```

```
1101  002444  012705  000400              MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1102  002450  104000                      HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1103  002452  017704  007444        3$:   MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1104  002456  026704  010242              CMP     RWRD3,R4                        ;IS RECEIVED CHARACTER A BREAK
1105  002462  001403                      BEQ     4$
1106  002464  016705  010234              MOV     RWRD3,R5                ;(R5)=EXPECTED RECEIVED CHARACTER
1107  002470  104001                      HLT     1                       ;RECEIVED DATA ERROR
1108  002472  104400                4$:   SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1109
1110                                      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1111                                      ;ON LINE 4
1112                                      ;SET BREAK BIT FOR LINE 4
1113                                      ;TRANSMIT BINARY COUNT PATTERN ON LINE 4
1114                                      ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1115                                      ;AND THAT IT IS A BREAK
1116
1117  002474  012767  000340  175274  T5:  MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1118  002502  012767  012120  007456        MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1119  002510  012767  000020  007444        MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1120  002516  012767  002732  007440        MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
1121  002524  012777  004000  007366        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1122  002532  004767  007470              JSR     PC,CLRALL               ;CLEAR ALL BUS AUDRESS AND
1123                                                                        ;BUS ADDRESS MEMORY LOCATIONS
1124  002536  012777  000004  007354        MOV     #4,@DHSCR               ;SELECT LINE 4
1125  002544  012777  012222  007354        MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1126  002552  012777  177776  007350        MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1127  002560  012777  033503  007336        MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1128                                                                        ;CHARACTER LENGTH =8 BITS
1129  002566  012777  000020  007336        MOV     #20,@DHBAR              ;SET BAR BIT FOR LINE 4
1130  002574  122777  000002  007336  1$:   CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1131  002602  001374                      BNE     1$
1132  002604  012777  004000  007306        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1133  002612  012777  000004  007300        MOV     #4,@DHSCR               ;SELECT LINE 4
1134  002620  012777  012254  007300        MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1135  002626  012777  177400  007274        MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1136  002634  012777  033503  007262        MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1137  002642  012777  000020  007264        MOV     #20,@DHBCR              ;SET BREAK BIT FOR LINE 4
1138  002650  012777  000020  007254        MOV     #20,@DHBAR              ;SET BAR BIT FOR LINE 4
1139  002656  005777  007250        2$:   TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1140  002662  001375                      BNE     2$                      ;TO BE TRANSMITTED
1141  002664  122777  000001  007246        CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1142  002672  001407                      BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1143  002674  017704  007236              MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1144  002700  042704  000300              BIC     #300,R4                 ;CLEAR UNWANTED BITS
1145  002704  012705  000400              MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1146  002710  104000                      HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1147  002712  017704  007204        3$:   MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1148  002716  026704  010004              CMP     RWRD4,R4                        ;IS RECEIVED CHARACTER A BREAK
1149  002722  001403                      BEQ     4$
1150  002724  016705  007776              MOV     RWRD4,R5                ;(R5)=EXPECTED RECEIVED CHARACTER
1151  002730  104001                      HLT     1                       ;RECEIVED DATA ERROR
1152  002736  104400                4$:   SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1153
1154                                      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1155                                      ;ON LINE 5
1156                                      ;SET BREAK BIT FOR LINE 5
```

```
1157                                                    ;TRANSMIT BINARY COUNT PATTERN ON LINE 5
1158                                                    ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1159                                                    ;AND THAT IT IS A BREAK
1160
1161  002734  012767  000340  175034  T6:   MOV   #340,PS          ;DISABLE ALL INTERRUPTS
1162  002742  012767  012120  007216        MOV   #DHSCR,ICOUNT    ;SET UP FOR DHSCR ITERATIONS
1163  002750  012767  000020  007204        MOV   #20,ESCAPE       ;SET UP TO ESCAPE TO NEXT TEST
1164  002756  012767  003172  007200        MOV   #4$,FREEZ1       ;SET UP TO LOOP WITH DATA
1165  002764  012777  004000  007126        MOV   #BIT11,@DHSCR    ;MASTER CLEAR INTERFACE
1166  002772  004767  007230                JSR   PC,CLRALL        ;CLEAR ALL BUS ADDRESS AND
1167                                                                ;BUS ADDRESS MEMORY LOCATIONS
1168  002776  012777  000005  007114        MOV   #5,@DHSCR        ;SELECT LINE 5
1169  003004  012777  012222  007114        MOV   #NULL,@DHBA      ;SET UPT TO TRANSMIT 0 CHARACTER
1170  003012  012777  177776  007110        MOV   #-2,@DHBC        ;TWO DS WILL BE TRANSMITTED
1171  003020  012777  033503  007076        MOV   #33503,@DHLPR    ;SET LINE SPEED=9600 BAUD
1172                                                                ;CHARACTER LENGTH =8 BITS
1173  003026  012777  000040  007076        MOV   #40,@DHBAR       ;SET BAR BIT FOR LINE 5
1174  003034  122777  000002  007076  1$:   CMPB  #2,@DHSLR        ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1175  003042  001374                BNE   1$
1176  003044  012777  004000  007046        MOV   #BIT11,@DHSCR    ;MASTER CLEAR INTERFACE
1177  003052  012777  000005  007040        MOV   #5,@DHSCR        ;SELECT LINE 5
1178  003060  012777  012254  007040        MOV   #TBUF,@DHBA      ;SET UP TO TRANSMIT 400
1179  003066  012777  177400  007034        MOV   #-400,@DHBC      ;(OCTAL) CHARACTERS
1180  003074  012777  033503  007022        MOV   #33503,@DHLPR    ;LINE SPEED = 9600 BAUD
1181  003102  012777  000040  007024        MOV   #40,@DHBCR       ;SET BREAK BIT FOR LINE 5
1182  003110  012777  000040  007014        MOV   #40,@DHBAR       ;SET BAR BIT FOR LINE 5
1183  003116  005777  007010          2$:   TST   @DHBAR           ;WAIT FOR ALL CHARACTERS
1184  003122  001375                BNE   2$                      ;TO BE TRANSMITTED
1185  003124  122777  000001  007006        CMPB  #1,@DHSLR        ;CHECK TO SEE THAT ONLY
1186  003132  001407                BEQ   3$                      ;1 CHARACTER WAS RECEIVED
1187  003134  017704  006776                MOV   @DHSSR,R4        ;(R4)=ACTUAL RECEIVED DATA
1188  003140  042704  000300                BIC   #300,R4          ;CLEAR UNWANTED BITS
1189  003144  012705  000400                MOV   #400,R5          ;(R5)=EXPECTED SILO FILL LEVEL, 1
1190  003150  104000                        HLT   0                ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1191  003152  017704  006744          3$:   MOV   @DHNRC,R4        ;READ NEXT RECEIVED CHARACTER REGISTER
1192  003156  026704  007546                CMP   RWRDS,R4                 ;IS RECEIVED CHARACTER A BREAK
1193  003162  001403                BEQ   4$
1194  003164  016705  007540                MOV   RWRDS,R5         ;(R5)=EXPECTED RECEIVED CHARACTER
1195  003170  104001                        HLT   1                ;RECEIVED DATA ERROR
1196  003172  104400          4$:   SCOPE                          ;CHECK FOR ITERATIONS, LOOP
1197
1198                                                    ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1199                                                    ;ON LINE 6
1200                                                    ;SET BREAK BIT FOR LINE 6
1201                                                    ;TRANSMIT BINARY COUNT PATTERN ON LINE 6
1202                                                    ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1203                                                    ;AND THAT IT IS A BREAK
1204
1205  003174  012767  000340  174574  T7:   MOV   #340,PS          ;DISABLE ALL INTERRUPTS
1206  003202  012767  012120  006756        MOV   #DHSCR,ICOUNT    ;SET UP FOR DHSCR ITERATIONS
1207  003210  012767  000020  006744        MOV   #20,ESCAPE       ;SET UP TO ESCAPE TO NEXT TEST
1208  003216  012767  003432  006740        MOV   #4$,FREEZ1       ;SET UP TO LOOP WITH DATA
1209  003224  012777  004000  006666        MOV   #BIT11,@DHSCR    ;MASTER CLEAR INTERFACE
1210  003232  004767  006770                JSR   PC,CLRALL        ;CLEAR ALL BUS ADDRESS AND
1211                                                                ;BUS ADDRESS MEMORY LOCATIONS
1212  003236  012777  000006  006654        MOV   #6,@DHSCR        ;SELECT LINE 6
```

```
1213  003244  012777  012222  006654          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1214  003252  012777  177776  006650          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1215  003260  012777  033503  006636          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1216                                                                          ;CHARACTER LENGTH =8 BITS
1217  003266  012777  000100  006636          MOV     #100,@DHBAR             ;SET BAR BIT FOR LINE 6
1218  003274  122777  000002  006636  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1219  003302  001374                          BNE     1$
1220  003304  012777  004000  006606          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1221  003312  012777  000006  006600          MOV     #6,@DHSCR               ;SELECT LINE 6
1222  003320  012777  012254  006600          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1223  003326  012777  177400  006574          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1224  003334  012777  033503  006562          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1225  003342  012777  000100  006564          MOV     #100,@DHSCR             ;SET BREAK BIT FOR LINE 6
1226  003350  012777  000100  006554          MOV     #100,@DHBAR             ;SET BAR BIT FOR LINE 6
1227  003356  005777  006550          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1228  003362  001375                          BNE     2$                      ;TO BE TRANSMITTED
1229  003364  122777  000001  006546          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1230  003372  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1231  003374  017704  006536                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1232  003400  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1233  003404  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1234  003410  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED. ERROR
1235  003412  017704  006504          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1236  003416  026704  007310                  CMP     RWR06,R4                ;IS RECEIVED CHARACTER A BREAK
1237  003422  001403                          BEQ     4$
1238  003424  016705  007302                  MOV     RWR06,R5                ;(R5)=EXPECTED RECEIVED CHARACTER
1239  003430  104001                          HLT     1                       ;RECEIVED DATA ERROR
1240  003432  104400          4$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
1241
1242                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1243                                          ;ON LINE 7
1244                                          ;SET BREAK BIT FOR LINE 7
1245                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 7
1246                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1247                                          ;AND THAT IT IS A BREAK
1248
1249  003434  012767  000340  174334  T10:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1250  003442  012767  012120  006516          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1251  003450  012767  000020  006504          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1252  003456  012767  003672  006500          MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
1253  003464  012777  004000  006426          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1254  003472  004767  006530                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1255                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1256  003476  012777  000007  006414          MOV     #7,@DHSCR               ;SELECT LINE 7
1257  003504  012777  012222  006414          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1258  003512  012777  177776  006410          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1259  003520  012777  033503  006376          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1260                                                                          ;CHARACTER LENGTH =8 BITS
1261  003526  012777  000200  006376          MOV     #200,@DHBAR             ;SET BAR BIT FOR LINE 7
1262  003534  122777  000002  006376  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1263  003542  001374                          BNE     1$
1264  003544  012777  004000  006346          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1265  003552  012777  000007  006340          MOV     #7,@DHSCR               ;SELECT LINE 7
1266  003560  012777  012254  006340          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1267  003566  012777  177400  006334          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1268  003574  012777  033503  006322          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
```

```
1269  003602  012777  000200  006324        MOV     #200,@DHBCR             ;SET BREAK BIT FOR LINE 7
1270  003610  012777  000200  006314        MOV     #200,@DHBAR             ;SET BAR BIT FOR LINE 7
1271  003616  005777  006310          2$:   TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1272  003622  001375                        BNE     2$                      ;TO BE TRANSMITTED
1273  003624  122777  000001  006306        CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1274  003632  001407                        BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1275  003634  017704  006276                MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1276  003640  042704  000300                BIC     #300,R4                 ;CLEAR UNWANTED BITS
1277  003644  012705  000400                MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1278  003650  104000                        HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1279  003652  017704  006244          3$:   MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1280  003656  026704  007052                CMP     RWRD7,R4                        ;IS RECEIVED CHARACTER A BREAK
1281  003662  001403                        BEQ     4$
1282  003664  016705  007044                MOV     RWRD7 R5                ;(R5)=EXPECTED RECEIVED CHARACTER
1283  003670  104001                        HLT     1                       ;RECEIVED DATA ERROR
1284  003672  104400          4$:   SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
1285
1286                                ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1287                                ;ON LINE 10
1288                                ;SET BREAK BIT FOR LINE 10
1289                                ;TRANSMIT BINARY COUNT PATTERN ON LINE 10
1290                                ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1291                                ;AND THAT IT IS A BREAK
1292
1293  003674  012767  000340  174074  T11:   MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1294  003702  012767  012120  006256        MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1295  003710  012767  000020  006244        MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1296  003716  012767  004132  006240        MOV     #4$,FREEZ1              ;SET UP TO LOOP WITH DATA
1297  003724  012777  004000  006136        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1298  003732  004767  006270                JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1299                                                                        ;BUS ADDRESS MEMORY LOCATIONS
1300  003736  012777  000010  006154        MOV     #10,@DHSCR              ;SELECT LINE 10
1301  003744  012777  012222  006154        MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1302  003752  012777  177776  006150        MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1303  003760  012777  033503  006136        MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1304                                                                        ;CHARACTER LENGTH =8 BITS
1305  003766  012777  000400  006136        MOV     #400,@DHBAR             ;SET BAR BIT FOR LINE 10
1306  003774  122777  000002  006136  1$:   CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1307  004002  001374                        BNE     1$
1308  004004  012777  004000  006106        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1309  004012  012777  000010  006100        MOV     #10,@DHSCR              ;SELECT LINE 10
1310  004020  012777  012254  006100        MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1311  004026  012777  177400  006074        MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1312  004034  012777  033503  006062        MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1313  004042  012777  000400  006064        MOV     #400,@DHBCR             ;SET BREAK BIT FOR LINE 10
1314  004050  012777  000400  006054        MOV     #400,@DHBAR             ;SET BAR BIT FOR LINE 10
1315  004056  005777  006050          2$:   TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1316  004062  001375                        BNE     2$                      ;TO BE TRANSMITTED
1317  004064  122777  000001  006046        CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1318  004072  001407                        BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1319  004074  017704  006036                MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1320  004100  042704  000300                BIC     #300,R4                 ;CLEAR UNWANTED BITS
1321  004104  012705  000400                MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1322  004110  104000                        HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1323  004112  017704  006004          3$:   MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1324  004116  026704  006614                CMP     RWRD10,R4                       ;IS RECEIVED CHARACTER A BREAK
```

```
1325  004122  001403                          BEQ     4$
1326  004124  016705  006606                  MOV     RWRD10,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1327  004130  104001                          HLT     1                       ;RECEIVED DATA ERROR
1328  004132  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1329
1330                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1331                                          ;ON LINE 11
1332                                          ;SET BREAK BIT FOR LINE 11
1333                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 11
1334                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1335                                          ;AND THAT IT IS A BREAK
1336
1337  004134  012767  000340  173634  T12:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1338  004142  012767  012120  006016          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1339  004150  012767  000020  006004          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1340  004156  012767  004372  006000          MOV     #4$,FREEZ1              ;SET UP TO LOOP WITH DATA
1341  004164  012777  004000  005726          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1342  004172  004767  006030                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1343                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1344  004176  012777  000011  005714          MOV     #11,@DHSCR              ;SELECT LINE 11
1345  004204  012777  012222  005714          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 3 CHARACTER
1346  004212  012777  177776  005710          MOV     #-2,@DHBC               ;TWO OS WILL BE TRANSMITTED
1347  004220  012777  033503  005676          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1348                                                                          ;CHARACTER LENGTH =8 BITS
1349  004226  012777  001000  005676          MOV     #1000,@DHBAR            ;SET BAR BIT FOR LINE 11
1350  004234  122777  000002  005676  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1351  004242  001374                          BNE     1$
1352  004244  012777  004000  005646          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1353  004252  012777  000011  005640          MOV     #11,@DHSCR              ;SELECT LINE 11
1354  004260  012777  012254  005640          MOV     #T6UF,@DHBA             ;SET UP TO TRANSMIT 400
1355  004266  012777  177400  005634          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1356  004274  012777  033503  005622          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1357  004302  012777  001000  005524          MOV     #1000,@DHBCR            ;SET BREAK BIT FOR LINE 11
1358  004310  012777  001000  005614          MOV     #1000,@DHBAR            ;SET BAR BIT FOR LINE 11
1359  004316  005777  005610          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1360  004322  001375                          BNE     2$                      ;TO BE TRANSMITTED
1361  004324  122777  000001  005606          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1362  004332  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1363  004334  017704  005576                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1364  004340  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1365  004344  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1366  004350  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1367  004352  017704  005544          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1368  004356  026704  006356                  CMP     RWRD11,R4                       ;IS RECEIVED CHARACTER A BREAK
1369  004362  001403                          BEQ     4$
1370  004364  016705  006350                  MOV     RWRD11,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1371  004370  104001                          HLT     1                       ;RECEIVED DATA ERROR
1372  004372  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1373
1374                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1375                                          ;ON LINE 12
1376                                          ;SET BREAK BIT FOR LINE 12
1377                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 12
1378                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1379                                          ;AND THAT IT IS A BREAK
1380
```

```
 1381  004374  012767  000340  173374  T13:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
 1382  004402  012767  012120  005556          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
 1383  004410  012767  000020  005544          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
 1384  004416  012767  004632  005540          MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
 1385  004424  012777  004000  005466          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 1386  004432  004767  005570                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
 1387                                                                          ;BUS ADDRESS MEMORY LOCATIONS
 1388  004436  012777  000012  005451          MOV     #12,@DHSCR              ;SELECT LINE 12
 1389  004444  012777  012222  005454          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
 1390  004452  012777  177776  005450          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
 1391  004460  012777  033503  005436          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
 1392                                                                          ;CHARACTER LENGTH =8 BITS
 1393  004466  012777  002000  005436          MOV     #2000,@DHBAR            ;SET BAR BIT FOR LINE 12
 1394  004474  122777  000002  005436  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
 1395  004502  001374                          BNE     1$
 1396  004504  012777  004000  005406          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 1397  004512  012777  000012  005400          MOV     #12,@DHSCR              ;SELECT LINE 12
 1398  004520  012777  012254  005400          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
 1399  004526  012777  177400  005374          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
 1400  004534  012777  033503  005362          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
 1401  004542  012777  002000  005364          MOV     #2000,@DHBCR            ;SET BREAK BIT FOR LINE 12
 1402  004550  012777  002000  005354          MOV     #2000,@DHBAR            ;SET BAR BIT FOR LINE 12
 1403  004556  005777  005350          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
 1404  004562  001375                          BNE     2$                      ;TO BE TRANSMITTED
 1405  004564  122777  000001  005346          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
 1406  004572  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
 1407  004574  017704  005336                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
 1408  004600  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
 1409  004604  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
 1410  004610  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
 1411  004612  017704  005324          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
 1412  004616  026704  006120                  CMP     RWRD12,R4               ;IS RECEIVED CHARACTER A BREAK
 1413  004622  001403                          BEQ     4$
 1414  004624  016705  006112                  MOV     RWRD12,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
 1415  004630  104001                          HLT     1                       ;RECEIVED DATA ERROR
 1416  004632  104400          4$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
 1417
 1418                                  ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
 1419                                  ;ON LINE 13
 1420                                  ;SET BREAK BIT FOR LINE 13
 1421                                  ;TRANSMIT BINARY COUNT PATTERN ON LINE 13
 1422                                  ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
 1423                                  ;AND THAT IT IS A BREAK
 1424
 1425  004634  012767  000340  173134  T14:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
 1426  004642  012767  012120  005316          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
 1427  004650  012767  000020  005304          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
 1428  004656  012767  005072  005300          MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
 1429  004664  012777  004000  005226          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 1430  004672  004767  005330                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
 1431                                                                          ;BUS ADDRESS MEMORY LOCATIONS
 1432  004676  012777  000013  005214          MOV     #13,@DHSCR              ;SELECT LINE 13
 1433  004704  012777  012222  005214          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
 1434  004712  012777  177776  005210          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
 1435  004720  012777  033503  005176          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
 1436                                                                          ;CHARACTER LENGTH =8 BITS
```

```
1437  004726  012777  004000  005176          MOV     #4000,@DHBAR            ;SET BAR BIT FOR LINE 13
1438  004734  122777  000002  005176  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1439  004742  001374                          BNE     1$
1440  004744  012777  004000  005146          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1441  004752  012777  000013  005140          MOV     #13,@DHSCR              ;SELECT LINE 13
1442  004760  012777  012254  005140          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1443  004766  012777  177400  005134          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1444  004774  012777  033503  005122          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1445  005002  012777  004000  005124          MOV     #4000,@DHBCR            ;SET BREAK BIT FOR LINE 13
1446  005010  012777  004000  005114          MOV     #4000,@DHBAR            ;SET BAR BIT FOR LINE 13
1447  005016  005777  005110          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1448  005022  001375                          BNE     2$                      ;TO BE TRANSMITTED
1449  005024  122777  000001  005106          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1450  005032  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1451  005034  017704  005076                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1452  005040  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1453  005044  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1454  005050  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1455  005052  017704  005044          3$:     MOV     @DHRBC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1456  005056  026704  005662                  CMP     RWRD13,R4                       ;IS RECEIVED CHARACTER A BREAK
1457  005062  001403                          BEQ     4$
1458  005064  016705  005654                  MOV     RWRD13,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1459  005070  104001                          HLT     1                       ;RECEIVED DATA ERROR
1460  005072  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1461
1462                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1463                                          ;ON LINE 14
1464                                          ;SET BREAK BIT FOR LINE 14
1465                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 14
1466                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1467                                          ;AND THAT IT IS A BREAK
1468
1469  005074  012767  000340  172674  T15:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1470  005102  012767  012120  005056          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1471  005110  012767  000020  005044          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1472  005116  012767  005332  005040          MOV     #4$,FREEZ1              ;SET UP TO LOOP WITH DATA
1473  005124  012777  004000  004766          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1474  005132  004767  005070                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1475                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1476  005136  012777  000014  004754          MOV     #14,@DHSCR              ;SELECT LINE 14
1477  005144  012777  012222  004754          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1478  005152  012777  177776  004750          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1479  005160  012777  033503  004736          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1480                                                                          ;CHARACTER LENGTH =8 BITS
1481  005166  012777  010000  004736          MOV     #10000,@DHBAR           ;SET BAR BIT FOR LINE 14
1482  005174  122777  000002  004736  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1483  005202  001374                          BNE     1$
1484  005204  012777  004000  004706          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1485  005212  012777  000014  004700          MOV     #14,@DHSCR              ;SELECT LINE 14
1486  005220  012777  012254  004700          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1487  005226  012777  177400  004674          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1488  005234  012777  033503  004662          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1489  005242  012777  010000  004664          MOV     #10000,@DHBCR           ;SET BREAK BIT FOR LINE 14
1490  005250  012777  010000  004654          MOV     #10000,@DHBAR           ;SET BAR BIT FOR LINE 14
1491  005256  005777  004650          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1492  005262  001375                          BNE     2$                      ;TO BE TRANSMITTED
```

```
1493  005264  122777  000001  004646          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1494  005272  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1495  005274  017704  004636                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1496  005300  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1497  005304  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1498  005310  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1499  005312  017704  004604          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1500  005316  026704  005424                  CMP     RWRD14,R4                       ;IS RECEIVED CHARACTER A BREAK
1501  005322  001403                          BEQ     4$
1502  005324  016705  005416                  MOV     RWRD14,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1503  005330  104001                          HLT     1                       ;RECEIVED DATA ERROR
1504  005332  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1505
1506                                          ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1507                                          ;ON LINE 15
1508                                          ;SET BREAK BIT FOR LINE 15
1509                                          ;TRANSMIT BINARY COUNT PATTERN ON LINE 15
1510                                          ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1511                                          ;AND THAT IT IS A BREAK
1512
1513  005334  012767  000340  172434  T16:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1514  005342  012767  012120  004616          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1515  005350  012767  000020  004604          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1516  005356  012767  005572  004600          MOV     #4$,FREEZI              ;SET UP TO LOOP WITH DATA
1517  005364  012777  004000  004526          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1518  005372  004767  004630                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1519                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1520  005376  012777  000015  004514          MOV     #15,@DHSCR              ;SELECT LINE 15
1521  005404  012777  012222  004514          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1522  005412  012777  177776  004510          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1523  005420  012777  033503  004476          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1524                                                                          ;CHARACTER LENGTH =8 BITS
1525  005426  012777  020000  004476          MOV     #20000,@DHBAR           ;SET BAR BIT FOR LINE 15
1526  005434  122777  000002  004476  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1527  005442  001374                          BNE     1$
1528  005444  012777  004000  004446          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1529  005452  012777  000015  004440          MOV     #15,@DHSCR              ;SELECT LINE 15
1530  005460  012777  012254  004440          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1531  005466  012777  177400  004434          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1532  005474  012777  033503  004422          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1533  005502  012777  020000  004424          MOV     #20000,@DHBCR           ;SET BREAK BIT FOR LINE 15
1534  005510  012777  020000  004414          MOV     #20000,@DHBAR           ;SET BAR BIT FOR LINE 15
1535  005516  005777  004410          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1536  005522  001375                          BNE     2$                      ;TO BE TRANSMITTED
1537  005524  122777  000001  004406          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1538  005532  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1539  005534  017704  004376                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1540  005540  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1541  005544  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1542  005550  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1543  005552  017704  004344          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1544  005556  026704  005166                  CMP     RWRD15,R4                       ;IS RECEIVED CHARACTER A BREAK
1545  005562  001403                          BEQ     4$
1546  005564  016705  005160                  MOV     RWRD15,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1547  005570  104001                          HLT     1                       ;RECEIVED DATA ERROR
1548  005572  104400                  4$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
1549
1550                                        ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1551                                        ;ON LINE 16
1552                                        ;SET BREAK BIT FOR LINE 16
1553                                        ;TRANSMIT BINARY COUNT PATTERN ON LINE 16
1554                                        ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1555                                        ;AND THAT IT IS A BREAK
1556
1557  005574  012767  000340  172174  T17:  MOV   #340,PS              ;DISABLE ALL INTERRUPTS
1558  005602  012767  012120  004356        MOV   #DHSCR,ICOUNT        ;SET UP FOR DHSCR ITERATIONS
1559  005610  012767  000020  004344        MOV   #20,ESCAPE           ;SET UP TO ESCAPE TO NEXT TEST
1560  005616  012767  006032  004340        MOV   #4$,FREEZ1           ;SET UP TO LOOP WITH DATA
1561  005624  012777  004000  004266        MOV   #BIT11,@DHSCR        ;MASTER CLEAR INTERFACE
1562  005632  004767  004370                JSR   PC,CLRALL            ;CLEAR ALL BUS ADDRESS AND
1563                                                                   ;BUS ADDRESS MEMORY LOCATIONS
1564  005636  012777  000016  004254        MOV   #16,@DHSCR           ;SELECT LINE 16
1565  005644  012777  012222  004254        MOV   #NULL,@DHBA          ;SET UPT TO TRANSMIT 0 CHARACTER
1566  005652  012777  177776  004250        MOV   #-2,@DHBC            ;TWO 0S WILL BE TRANSMITTED
1567  005660  012777  033503  004236        MOV   #33503,@DHLPR        ;SET LINE SPEED=9600 BAUD
1568                                                                   ;CHARACTER LENGTH =8 BITS
1569  005666  012777  040000  004236        MOV   #40000,@DHBAR        ;SET BAR BIT FOR LINE 16
1570  005674  122777  000002  004236  1$:   CMPB  #2,@DHSLR            ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1571  005702  001374                       BNE   1$
1572  005704  012777  004000  004206        MOV   #BIT11,@DHSCR        ;MASTER CLEAR INTERFACE
1573  005712  012777  000016  004200        MOV   #16,@DHSCR           ;SELECT LINE 16
1574  005720  012777  012254  004200        MOV   #TBUF,@DHBA          ;SET UP TO TRANSMIT 400
1575  005726  012777  177400  004174        MOV   #-400,@DHBC          ;(OCTAL) CHARACTERS
1576  005734  012777  033503  004162        MOV   #33503,@DHLPR        ;LINE SPEED = 9600 BAUD
1577  005742  012777  040000  004164        MOV   #40000,@DHBCR        ;SET BREAK BIT FOR LINE 16
1578  005750  012777  040000  004154        MOV   #40000,@DHBAR        ;SET BAR BIT FOR LINE 16
1579  005756  005777  004150          2$:   TST   @DHBAR               ;WAIT FOR ALL CHARACTERS
1580  005762  001375                       BNE   2$                   ;TO BE TRANSMITTED
1581  005764  122777  000001  004146        CMPB  #1,@DHSLR            ;CHECK TO SEE THAT ONLY
1582  005772  001407                       BEQ   3$                   ;1 CHARACTER WAS RECEIVED
1583  005774  017704  004136                MOV   @DHSSR,R4            ;(R4)=ACTUAL RECEIVED DATA
1584  006000  042704  000300                BIC   #300,R4              ;CLEAR UNWANTED BITS
1585  006004  012705  000400                MOV   #400,R5              ;(R5)=EXPECTED SILO FILL LEVEL, 1
1586  006010  104000                       HLT   0                    ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1587  006012  017704  004104          3$:   MOV   @DHNRC,R4            ;READ NEXT RECEIVED CHARACTER REGISTER
1588  006016  026704  004730                CMP   RWRD16,R4                  ;IS RECEIVED CHARACTER A BREAK
1589  006022  001403                       BEQ   4$
1590  006024  016705  004722                MOV   RWRD16,R5            ;(R5)=EXPECTED RECEIVED CHARACTER
1591  006030  104001                       HLT   1                    ;RECEIVED DATA ERROR
1592  006032  104400          4$:   SCOPE                              ;CHECK FOR ITERATIONS, LOOP
1593
1594                                        ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1595                                        ;ON LINE 17
1596                                        ;SET BREAK BIT FOR LINE 17
1597                                        ;TRANSMIT BINARY COUNT PATTERN ON LINE 17
1598                                        ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1599                                        ;AND THAT IT IS A BREAK
1600
1601  006034  012767  000340  171734  T20:  MOV   #340,PS              ;DISABLE ALL INTERRUPTS
1602  006042  012767  012120  004116        MOV   #DHSCR,ICOUNT        ;SET UP FOR DHSCR ITERATIONS
1603  006050  012767  000020  004104        MOV   #20,ESCAPE           ;SET UP TO ESCAPE TO NEXT TEST
1604  006056  012767  006272  004100        MOV   #4$,FREEZ1           ;SET UP TO LOOP WITH DATA
```

```
1605  006064  012777  004000  004026          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1606  006072  004767  004130                  JSR     PC,CLRALL               ;CLEAR ALL BUS ADDRESS AND
1607                                                                          ;BUS ADDRESS MEMORY LOCATIONS
1608  006076  012777  000017  004014          MOV     #17,@DHSCR              ;SELECT LINE 17
1609  006104  012777  012222  004014          MOV     #NULL,@DHBA             ;SET UPT TO TRANSMIT 0 CHARACTER
1610  006112  012777  177776  004010          MOV     #-2,@DHBC               ;TWO 0S WILL BE TRANSMITTED
1611  006120  012777  033503  003776          MOV     #33503,@DHLPR           ;SET LINE SPEED=9600 BAUD
1612                                                                          ;CHARACTER LENGTH =8 BITS
1613  006126  012777  100000  003776          MOV     #100000,@DHBAR          ;SET BAR BIT FOR LINE 17
1614  006134  122777  000002  003776  1$:     CMPB    #2,@DHSLR               ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1615  006142  001374                          BNE     1$
1616  006144  012777  004000  003746          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1617  006152  012777  000017  003740          MOV     #17,@DHSCR              ;SELECT LINE 17
1618  006160  012777  012254  003740          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT 400
1619  006166  012777  177400  003734          MOV     #-400,@DHBC             ;(OCTAL) CHARACTERS
1620  006174  012777  033503  003722          MOV     #33503,@DHLPR           ;LINE SPEED = 9600 BAUD
1621  006202  012777  100000  003724          MOV     #100000,@DHBCR          ;SET BREAK BIT FOR LINE 17
1622  006210  012777  100000  003714          MOV     #100000,@DHBAR          ;SET BAR BIT FOR LINE 17
1623  006216  005777  003710          2$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS
1624  006222  001375                          BNE     2$                      ;TO BE TRANSMITTED
1625  006224  122777  000001  003706          CMPB    #1,@DHSLR               ;CHECK TO SEE THAT ONLY
1626  006232  001407                          BEQ     3$                      ;1 CHARACTER WAS RECEIVED
1627  006234  017704  003676                  MOV     @DHSSR,R4               ;(R4)=ACTUAL RECEIVED DATA
1628  006240  042704  000300                  BIC     #300,R4                 ;CLEAR UNWANTED BITS
1629  006244  012705  000400                  MOV     #400,R5                 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1630  006250  104000                          HLT     0                       ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1631  006252  017704  003644          3$:     MOV     @DHNRC,R4               ;READ NEXT RECEIVED CHARACTER REGISTER
1632  006256  026704  004472                  CMP     RWRD17,R4               ;IS RECEIVED CHARACTER A BREAK
1633  006262  001403                          BEQ     4$
1634  006264  016705  004464                  MOV     RWRD17,R5               ;(R5)=EXPECTED RECEIVED CHARACTER
1635  006270  104001                          HLT     1                       ;RECEIVED DATA ERROR
1636  006272  104400          4$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
1637
1638                                  ;SET HALF DUPLEX ON LINE 0
1639                                  ;TRANSMIT A BINARY COUNT PATTERN
1640                                  ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1641
1642  006274  012767  000340  171474  T21:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1643  006302  012767  012120  003656          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1644  006310  012767  000020  003644          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1645  006316  012767  006412  003640          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1646  006324  012777  004000  003566          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1647  006332  004767  003670                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1648                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1649  006336  012777  000000  003554          MOV     #0,@DHSCR               ;SELECT LINE 0
1650  006344  012777  012254  003554          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1651  006352  012777  177400  003550          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1652  006360  012777  073503  003536          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1653                                                                          ;LINE SPEED =9600 BAUD
1654                                                                          ;CHARACTER LENGTH = 8 BITS
1655  006366  012777  000001  003536          MOV     #1,@DHBAR               ;SET BAR BIT FOR LINE 0
1656  006374  005777  003532          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1657  006400  001375                          BNE     1$
1658  006402  105777  003512                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1659  006406  100002                          BPL     2$
1660  006410  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
```

```
1661  006412  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1662
1663                                          ;SET HALF DUPLEX ON LINE 1
1664                                          ;TRANSMITI A BINARY COUNT PATTERN
1665                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1666
1667  006414  012767  000340  171354  T22:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1668  006422  012767  012120  003536          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1669  006430  012767  000020  003524          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1670  006436  012767  006532  003520          MOV     #2$,FREEZI              ;SET UP TO LOOP WITH DATA
1671  006444  012777  004000  003446          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1672  006452  004767  003550                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1673                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1674  006456  012777  000001  003434          MOV     #1,@DHSCR               ;SELECT LINE 1
1675  006464  012777  012254  003434          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1676  006472  012777  177400  003430          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1677  006500  012777  073503  003416          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1678                                                                          ;LINE SPEED =9600 BAUD
1679                                                                          ;CHARACTER LENGTH = 8 BITS
1680  006506  012777  000002  003416          MOV     #2,@DHBAR               ;SET BAR BIT FOR LINE 1
1681  006514  005777  003412          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1682  006520  001375                          BNE     1$
1683  006522  105777  003372                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1684  006526  100001                          BPL     2$
1685  006530  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1686  006532  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1687
1688                                          ;SET HALF DUPLEX ON LINE 2
1689                                          ;TRANSMITI A BINARY COUNT PATTERN
1690                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1691
1692  006534  012767  000340  171234  T23:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1693  006542  012767  012120  003416          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1694  006550  012767  000020  003404          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1695  006556  012767  006652  003400          MOV     #2$,FREEZI              ;SET UP TO LOOP WITH DATA
1696  006564  012777  004000  003326          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1697  006572  004767  003430                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1698                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1699  006576  012777  000002  003314          MOV     #2,@DHSCR               ;SELECT LINE 2
1700  006604  012777  012254  003314          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1701  006612  012777  177400  003310          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1702  006620  012777  073503  003276          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1703                                                                          ;LINE SPEED =9600 BAUD
1704                                                                          ;CHARACTER LENGTH = 8 BITS
1705  006626  012777  000004  003276          MOV     #4,@DHBAR               ;SET BAR BIT FOR LINE 2
1706  006634  005777  003272          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1707  006640  001375                          BNE     1$
1708  006642  105777  003252                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1709  006646  100001                          BPL     2$
1710  006650  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1711  006652  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1712
1713                                          ;SET HALF DUPLEX ON LINE 3
1714                                          ;TRANSMITI A BINARY COUNT PATTERN
1715                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1716
```

```
1717  006654  012767  000340  171114  T24:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1718  006662  012767  012120  003276          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1719  006670  012767  000020  003264          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1720  006676  012767  006772  003260          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1721  006704  012777  004000  003206          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1722  006712  004767  003310                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1723                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1724  006716  012777  000003  003174          MOV     #3,@DHSCR               ;SELECT LINE 3
1725  006724  012777  012254  003174          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1726  006732  012777  177400  003170          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1727  006740  012777  073503  003156          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1728                                                                          ;LINE SPEED =9600 BAUD
1729                                                                          ;CHARACTER LENGTH = 8 BITS
1730  006746  012777  000010  003156          MOV     #10,@DHBAR              ;SET BAR BIT FOR LINE 3
1731  006754  005777  003152          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1732  006760  001375                          BNE     1$
1733  006762  105777  003132                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1734  006766  100001                          BPL     2$
1735  006770  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1736  006772  104400          2$:             SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1737
1738                                          ;SET HALF DUPLEX ON LINE 4
1739                                          ;TRANSMITI A BINARY COUNT PATTERN
1740                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1741
1742  006774  012767  000340  170774  T25:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1743  007002  012767  012120  003156          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1744  007010  012767  000020  003144          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1745  007016  012767  007112  003140          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1746  007024  012777  004000  003066          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1747  007032  004767  003170                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1748                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1749  007036  012777  000004  003054          MOV     #4,@DHSCR               ;SELECT LINE 4
1750  007044  012777  012254  003054          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1751  007052  012777  177400  003050          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1752  007060  012777  073503  003036          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1753                                                                          ;LINE SPEED =9600 BAUD
1754                                                                          ;CHARACTER LENGTH = 8 BITS
1755  007066  012777  000020  003036          MOV     #20,@DHBAR              ;SET BAR BIT FOR LINE 4
1756  007074  005777  003032          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1757  007100  001375                          BNE     1$
1758  007102  105777  003012                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1759  007106  100001                          BPL     2$
1760  007110  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1761  007112  104400          2$:             SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1762
1763                                          ;SET HALF DUPLEX ON LINE 5
1764                                          ;TRANSMITI A BINARY COUNT PATTERN
1765                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1766
1767  007114  012767  000340  170654  T26:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1768  007122  012767  012120  003036          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1769  007130  012767  000020  003024          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1770  007136  012767  007232  003020          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1771  007144  012777  004000  002746          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1772  007152  004767  003050                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
```

```
1773                                                                    ;AND BUS ADDRESS MEMORY LOCATIONS
1774  007156  012777  000005  002734          MOV     #5,@DHSCR         ;SELECT LINE 5
1775  007164  012777  012254  002734          MOV     #TBUF,@DHBA       ;SET UP TO TRANSMIT
1776  007172  012777  177400  002730          MOV     #-400,@DHBC       ;400 (OCTAL) CHARACTERS
1777  007200  012777  073503  002716          MOV     #73503,@DHLPR     ;SET RECEIVER BLIND
1778                                                                    ;LINE SPEED =9600 BAUD
1779                                                                    ;CHARACTER LENGTH = 8 BITS
1780  007206  012777  000040  002716          MOV     #40,@DHBAR        ;SET BAR BIT FOR LINE 5
1781  007214  005777  002712          1$:     TST     @DHBAR            ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1782  007220  001375                          BNE     1$
1783  007222  105777  002672                  TSTB    @DHSCR            ;WERE ANY CHARACTERS RECEIVED
1784  007226  100001                          BPL     2$
1785  007230  104002                          HLT     2                 ;RECEIVER NOT BLINDED, ERROR
1786  007232  104400                  2$:     SCOPE                     ;CHECK FOR ITERATIONS, LOOP
1787
1788                                          ;SET HALF DUPLEX ON LINE 6
1789                                          ;TRANSMITI A BINARY COUNT PATTERN
1790                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1791
1792  007234  012767  000340  170534  T27:     MOV     #340,PS           ;DISABLE ALL INTERRUPTS
1793  007242  012767  012120  002716          MOV     #DHSCR,ICOUNT     ;SET UP FOR DHSCR ITERATIONS
1794  007250  012767  000020  002704          MOV     #20,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1795  007256  012767  007352  002700          MOV     #2$,FREEZ1        ;SET UP TO LOOP WITH DATA
1796  007264  012777  004000  002626          MOV     #BIT11,@DHSCR     ;MASTER CLEAR INTERFACE
1797  007272  004767  002730                  JSR     PC,CLRALL         ;CLEAR ALL BYTE COUNT AND
1798                                                                    ;AND BUS ADDRESS MEMORY LOCATIONS
1799  007276  012777  000006  002614          MOV     #6,@DHSCR         ;SELECT LINE 6
1800  007304  012777  012254  002614          MOV     #TBUF,@DHBA       ;SET UP TO TRANSMIT
1801  007312  012777  177400  002610          MOV     #-400,@DHBC       ;400 (OCTAL) CHARACTERS
1802  007320  012777  073503  002576          MOV     #73503,@DHLPR     ;SET RECEIVER BLIND
1803                                                                    ;LINE SPEED =9600 BAUD
1804                                                                    ;CHARACTER LENGTH = 8 BITS
1805  007326  012777  000100  002576          MOV     #100,@DHBAR       ;SET BAR BIT FOR LINE 6
1806  007334  005777  002572          1$:     TST     @DHBAR            ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1807  007340  001375                          BNE     1$
1808  007342  105777  002552                  TSTB    @DHSCR            ;WERE ANY CHARACTERS RECEIVED
1809  007346  100001                          BPL     2$
1810  007350  104002                          HLT     2                 ;RECEIVER NOT BLINDED, ERROR
1811  007352  104400                  2$:     SCOPE                     ;CHECK FOR ITERATIONS, LOOP
1812
1813                                          ;SET HALF DUPLEX ON LINE 7
1814                                          ;TRANSMITI A BINARY COUNT PATTERN
1815                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1816
1817  007354  012767  000340  170414  T30:     MOV     #340,PS           ;DISABLE ALL INTERRUPTS
1818  007362  012767  012120  002576          MOV     #DHSCR,ICOUNT     ;SET UP FOR DHSCR ITERATIONS
1819  007370  012767  000020  002564          MOV     #20,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1820  007376  012767  007472  002560          MOV     #2$,FREEZ1        ;SET UP TO LOOP WITH DATA
1821  007404  012777  004000  002506          MOV     #BIT11,@DHSCR     ;MASTER CLEAR INTERFACE
1822  007412  004767  002610                  JSR     PC,CLRALL         ;CLEAR ALL BYTE COUNT AND
1823                                                                    ;AND BUS ADDRESS MEMORY LOCATIONS
1824  007416  012777  000007  002474          MOV     #7,@DHSCR         ;SELECT LINE 7
1825  007424  012777  012254  002474          MOV     #TBUF,@DHBA       ;SET UP TO TRANSMIT
1826  007432  012777  177400  002470          MOV     #-400,@DHBC       ;400 (OCTAL) CHARACTERS
1827  007440  012777  073503  002456          MOV     #73503,@DHLPR     ;SET RECEIVER BLIND
1828                                                                    ;LINE SPEED =9600 BAUD
```

```
1829                                                                    ;CHARACTER LENGTH = 8 BITS
1830  007446  012777  000200  002456          MOV     #200,@DHBAR       ;SET BAR BIT FOR LINE 7
1831  007454  005777  002452          1$:     TST     @DHBAR            ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1832  007460  001375                  BNE     1$
1833  007462  105777  002432          TSTB    @DHSCR                    ;WERE ANY CHARACTERS RECEIVED
1834  007466  100001                  BPL     2$
1835  007470  104002                  HLT     2                         ;RECEIVER NOT BLINDED, ERROR
1836  007472  104400          2$:     SCOPE                             ;CHECK FOR ITERATIONS, LOOP
1837
1838                                  ;SET HALF DUPLEX ON LINE 10
1839                                  ;TRANSMITI A BINARY COUNT PATTERN
1840                                  ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1841
1842  007474  012767  000340  170274  T31:    MOV     #340,PS           ;DISABLE ALL INTERRUPTS
1843  007502  012767  012120  002456          MOV     #DHSCR,ICOUNT     ;SET UP FOR DHSCR ITERATIONS
1844  007510  012767  000020  002444          MOV     #20,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1845  007516  012767  007612  002440          MOV     #2$,FREEZI        ;SET UP TO LOOP WITH DATA
1846  007524  012777  004000  002366          MOV     #BIT11,@DHSCR     ;MASTER CLEAR INTERFACE
1847  007532  004767  002470                  JSR     PC,CLRALL         ;CLEAR ALL BYTE COUNT AND
1848                                                                    ;AND BUS ADDRESS MEMORY LOCATIONS
1849  007536  012777  000010  002354          MOV     #10,@DHSCR        ;SELECT LINE 10
1850  007544  012777  012254  002354          MOV     #TBUF,@DHBA       ;SET UP TO TRANSMIT
1851  007552  012777  177400  002350          MOV     #-400,@DHBC       ;400 (OCTAL) CHARACTERS
1852  007560  012777  073503  002336          MOV     #73503,@DHLPR     ;SET RECEIVER BLIND
1853                                                                    ;LINE SPEED =9600 BAUD
1854                                                                    ;CHARACTER LENGTH = 8 BITS
1855  007566  012777  000400  002336          MOV     #400,@DHBAR       ;SET BAR BIT FOR LINE 10
1856  007574  005777  002332          1$:     TST     @DHBAR            ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1857  007600  001375                  BNE     1$
1858  007602  105777  002312          TSTB    @DHSCR                    ;WERE ANY CHARACTERS RECEIVED
1859  007606  100001                  BPL     2$
1860  007610  104002                  HLT     2                         ;RECEIVER NOT BLINDED, ERROR
1661  007612  104400          2$:     SCOPE                             ;CHECK FOR ITERATIONS, LOOP
1862
1863                                  ;SET HALF DUPLEX ON LINE 11
1864                                  ;TRANSMITI A BINARY COUNT PATTERN
1865                                  ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1866
1867  007614  012767  000340  170154  T32:    MOV     #340,PS           ;DISABLE ALL INTERRUPTS
1868  007622  012767  012120  002336          MOV     #DHSCR,ICOUNT     ;SET UP FOR DHSCR ITERATIONS
1869  007630  012767  000020  002324          MOV     #20,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1870  007636  012767  007732  002320          MOV     #2$,FREEZI        ;SET UP TO LOOP WITH DATA
1871  007644  012777  004000  002246          MOV     #BIT11,@DHSCR     ;MASTER CLEAR INTERFACE
1872  007652  004767  002350                  JSR     PC,CLRALL         ;CLEAR ALL BYTE COUNT AND
1873                                                                    ;AND BUS ADDRESS MEMORY LOCATIONS
1874  007656  012777  000011  002234          MOV     #11,@DHSCR        ;SELECT LINE 11
1875  007664  012777  012254  002234          MOV     #TBUF,@DHBA       ;SET UP TO TRANSMIT
1876  007672  012777  177400  002230          MOV     #-400,@DHBC       ;400 (OCTAL) CHARACTERS
1877  007700  012777  073503  002216          MOV     #73503,@DHLPR     ;SET RECEIVER BLIND
1878                                                                    ;LINE SPEED =9600 BAUD
1879                                                                    ;CHARACTER LENGTH = 8 BITS
1880  007706  012777  001000  002216          MOV     #1000,@DHBAR      ;SET BAR BIT FOR LINE 11
1881  007714  005777  002212          1$:     TST     @DHBAR            ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1882  007720  001375                  BNE     1$
1883  007722  105777  002172          TSTB    @DHSCR                    ;WERE ANY CHARACTERS RECEIVED
1884  007726  100001                  BPL     2$
```

```
1885  007730  104002                        HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1886  007732  104400              2$:       SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1887
1888                                        ;SET HALF DUPLEX ON LINE 12
1889                                        ;TRANSMIT A BINARY COUNT PATTERN
1890                                        ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1891
1892  007734  012767  000340  170034  T33:  MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1893  007742  012767  012120  002216        MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1894  007750  012767  000020  002204        MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1895  007756  012767  010052  002200        MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1896  007764  012777  004000  002126        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1897  007772  004767  002230                JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1898                                                                        ;AND BUS ADDRESS MEMORY LOCATIONS
1899  007776  012777  000012  002114        MOV     #12,@DHSCR              ;SELECT LINE 12
1900  010004  012777  012254  002114        MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1901  010012  012777  177400  002110        MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1902  010020  012777  073503  002076        MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1903                                                                        ;LINE SPEED =9600 BAUD
1904                                                                        ;CHARACTER LENGTH = 8 BITS
1905  010026  012777  002000  002076        MOV     #2000,@DHBAR            ;SET BAR BIT FOR LINE 12
1906  010034  005777  002072        1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1907  010040  001375                BNE     1$
1908  010042  105777  002052              TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1909  010046  100001                BPL     2$
1910  010050  104002                HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1911  010052  104400              2$:       SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1912
1913                                        ;SET HALF DUPLEX ON LINE 13
1914                                        ;TRANSMIT A BINARY COUNT PATTERN
1915                                        ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1916
1917  010054  012767  000340  167714  T34:  MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1918  010062  012767  012120  002076        MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1919  010070  012767  000020  002064        MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1920  010076  012767  010172  002060        MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1921  010104  012777  004000  002006        MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1922  010112  004767  002110                JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1923                                                                        ;AND BUS ADDRESS MEMORY LOCATIONS
1924  010116  012777  000013  001774        MOV     #13,@DHSCR              ;SELECT LINE 13
1925  010124  012777  012254  001774        MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1926  010132  012777  177400  001770        MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1927  010140  012777  073503  001756        MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1928                                                                        ;LINE SPEED =9600 BAUD
1929                                                                        ;CHARACTER LENGTH = 8 BITS
1930  010146  012777  004000  001756        MOV     #4000,@DHBAR            ;SET BAR BIT FOR LINE 13
1931  010154  005777  001752        1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1932  010160  001375                BNE     1$
1933  010162  105777  001732              TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1934  010166  100001                BPL     2$
1935  010170  104002                HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1936  010172  104400              2$:       SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1937
1938                                        ;SET HALF DUPLEX ON LINE 14
1939                                        ;TRANSMIT A BINARY COUNT PATTERN
1940                                        ;VERIFY THAT NO CHARACTERS ARE RECEIVED
```

```
:941
1942  010174  012767  000340  167574  T35:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1943  010202  012767  012120  001756          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1944  010210  012767  000020  001744          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1945  010216  012767  010312  001740          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1946  010224  012777  004000  001666          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1947  010232  004767  001770                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1948                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1949  010236  012777  000014  001654          MOV     #14,@DHSCR              ;SELECT LINE 14
1950  010244  012777  012254  001654          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1951  010252  012777  177400  001650          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1952  010260  012777  073503  001636          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1953                                                                          ;LINE SPEED =9600 BAUD
1954                                                                          ;CHARACTER LENGTH = 8 BITS
1955  010266  012777  010000  001636          MOV     #10000,@DHBAR           ;SET BAR BIT FOR LINE 14
1956  010274  005777  001632          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1957  010300  001375                          BNE     1$
1958  010302  105777  001612                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1959  010306  100001                          BPL     2$
1960  010310  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1961  010312  104400          2$:             SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1962
1963                                  ;SET HALF DUPLEX ON LINE 15
1964                                  ;TRANSMIT A BINARY COUNT PATTERN
1965                                  ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1966
1967  010314  012767  000340  167454  T36:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1968  010322  012767  012120  001636          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1969  010330  012767  000020  001624          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1970  010336  012767  010432  001620          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1971  010344  012777  004000  001546          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1972  010352  004767  001650                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
1973                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
1974  010356  012777  000015  001534          MOV     #15,@DHSCR              ;SELECT LINE 15
1975  010364  012777  012254  001534          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
1976  010372  012777  177400  001530          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
1977  010400  012777  073503  001516          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
1978                                                                          ;LINE SPEED =9600 BAUD
1979                                                                          ;CHARACTER LENGTH = 8 BITS
1980  010406  012777  020000  001516          MOV     #20000,@DHBAR           ;SET BAR BIT FOR LINE 15
1981  010414  005777  001512          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1982  010420  001375                          BNE     1$
1983  010422  105777  001472                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
1984  010426  100001                          BPL     2$
1985  010430  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
1986  010432  104400          2$:             SCOPE                           ;CHECK FOR ITERATIONS, LOOP
1987
1988                                  ;SET HALF DUPLEX ON LINE 16
1989                                  ;TRANSMIT A BINARY COUNT PATTERN
1990                                  ;VERIFY THAT NO CHARACTERS ARE RECEIVED
1991
1992  010434  012767  000340  167334  T37:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1993  010442  012767  012120  001516          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
1994  010450  012767  000020  001504          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
1995  010456  012767  010532  001500          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
1996  010464  012777  004000  001426          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
```

```
 1997  010472  004767  001530                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
 1998                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
 1999  010476  012777  000016  001414          MOV     #16,@DHSCR              ;SELECT LINE 16
 2000  010504  012777  012254  001414          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
 2001  010512  012777  177400  001410          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
 2002  010520  012777  073503  001376          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
 2003                                                                          ;LINE SPEED =9600 BAUD
 2004                                                                          ;CHARACTER LENGTH = 8 BITS
 2005  010526  012777  040000  001376          MOV     #40000,@DHBAR           ;SET BAR BIT FOR LINE 16
 2006  010534  005777  001372          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
 2007  010540  001375                          BNE     1$
 2008  010542  105777  001352                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
 2009  010546  100001                          BPL     2$
 2010  010550  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
 2011  010552  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
 2012
 2013                                          ;SET HALF DUPLEX ON LINE 17
 2014                                          ;TRANSMITI A BINARY COUNT PATTERN
 2015                                          ;VERIFY THAT NO CHARACTERS ARE RECEIVED
 2016
 2017  010554  012767  000340  167214  T40:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
 2018  010562  012767  012120  001376          MOV     #DHSCR,ICOUNT           ;SET UP FOR DHSCR ITERATIONS
 2019  010570  012767  000020  001364          MOV     #20,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
 2020  010576  012767  010672  001360          MOV     #2$,FREEZ1              ;SET UP TO LOOP WITH DATA
 2021  010604  012777  004000  001306          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
 2022  010612  004767  001410                  JSR     PC,CLRALL               ;CLEAR ALL BYTE COUNT AND
 2023                                                                          ;AND BUS ADDRESS MEMORY LOCATIONS
 2024  010616  012777  000017  001274          MOV     #17,@DHSCR              ;SELECT LINE 17
 2025  010624  012777  012254  001274          MOV     #TBUF,@DHBA             ;SET UP TO TRANSMIT
 2026  010632  012777  177400  001270          MOV     #-400,@DHBC             ;400 (OCTAL) CHARACTERS
 2027  010640  012777  073503  001256          MOV     #73503,@DHLPR           ;SET RECEIVER BLIND
 2028                                                                          ;LINE SPEED =9600 BAUD
 2029                                                                          ;CHARACTER LENGTH = 8 BITS
 2030  010646  012777  100000  001256          MOV     #100000,@DHBAR          ;SET BAR BIT FOR LINE 17
 2031  010654  005777  001252          1$:     TST     @DHBAR                  ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
 2032  010660  001375                          BNE     1$
 2033  010662  105777  001232                  TSTB    @DHSCR                  ;WERE ANY CHARACTERS RECEIVED
 2034  010666  100001                          BPL     2$
 2035  010670  104002                          HLT     2                       ;RECEIVER NOT BLINDED, ERROR
 2036  010672  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
2037
2038
2039                                            ;END OF PASS
2040                                            ;TYPE NAME OF TEST
2041                                            ;UPDATE PASS COUNT
2042                                            ;CHECK FOR EXIT TO ACT-11
2043                                            ;RESTART TEST
2044
2045  010674  104401                  EOP:      TYPE                              ;TYPE NAME OF TEST
2046  010676  013342                            MEPASS
2047  010700  005067  001312                   CLR       LAST                    ;CLEAR LAST ERROR PC
2048  010704  005067  001242                   CLR       ERRFLG                  ;CLEAR ERROR FLAG
2049  010710  005267  001240                   INC       PASCNT                  ;UPDATE PASS COUNT
2050  010714  016767  001234  166646          MOV       PASCNT,LIGHTS           ;DISPLAY PASS COUNT
2051  010722  013701  000042                   MOV       @#42,R1                 ;CHECK FOR ACT-11 OR DDP
2052  010726  001405                           BEQ       RESTRT                  ;IF NOT, CONTINUE TESTING
2053  010730  000005                           RESET
2054  010732  004711                  LOGICAL:           JSR       PC,(R1)
2055  010734  000240                           NOP
2056  010736  000240                           NOP
2057  010740  000240                           NOP
2058  010742  000167  170234          RESTRT:  JMP       BEGIN
2059
2060                                            ;CHECK FOR LOOP ON CURRENT TEST
2061                                            ;CHECK FOR ITERATION SUPPRESSION
2062
2063  010746  032767  002000  166614  SCOPER:  BIT       #SW10,SWR
2064  010754  001030                           BNE       4$
2065  010756  032767  040000  166604  1$:      BIT       #SW14,SWR
2066  010764  001021                           BNE       3$
2067  010766  032767  004000  166574           BIT       #SW11,SWR
2068  010774  001006                           BNE       2$
2069  010776  005267  001166                   INC       LPCNT
2070  011002  026767  001162  001156          CMP       LPCNT,ICOUNT
2071  011010  001007                           BNE       3$
2072  011012  005067  001152          2$:      CLR       LPCNT
2073  011016  005067  001130                   CLR       ERRFLG
2074  011022  011667  001132                   MOV       (SP),RETURN
2075  011026  000002                           RTI
2076  011030  016716  001124          3$:      MOV       RETURN,(SP)
2077  011034  000002                           RTI
2078  011036  005767  001110          4$:      TST       ERRFLG
2079  011042  001745                           BEQ       1$
2080  011044  000762                           BR        2$
2081
2082                                            ;CHECK FOR FREEZE ON CURRENT DATA
2083
2084  011046  032767  001000  166514  SCOP1R:  BIT       #SW09,SWR
2085  011054  001402                           BEQ       1$
2086  011056  016716  001102                   MOV       FREEZ1,(SP)
2087  011062  000002                  1$:      RTI
```

```
2088
2089                                                    ;ERROR HANDLER
2090
2091  011064  032767  020000  166476  ERRORS: BIT     #SW13.SWR
2092  011072  001051                          BNE     HALTS
2093  011074  021667  001116                  CMP     (SP),LAST
2094  011100  001404                          BEQ     1$
2095  011102  011667  001110                  MOV     (SP),LAST
2096  011106  005067  001040                  CLR     ERRFLG
2097  011112  104406                  1$:     SAVOSP
2098  011114  011605                          MOV     (SP),R5
2099  011116  162705  000002                  SUB     #2,R5
2100  011122  011504                          MOV     (R5),R4
2101  011124  006304                          ASL     R4
2102  011126  006304                          ASL     R4
2103  011130  042704  177001                  BIC     #177001,R4
2104  011134  062704  013452                  ADD     #ERRTAB,R4
2105  011140  012467  000034                  MOV     (R4)+,ERRMSG
2106  011144  011467  000042                  MOV     (R4),DATABP
2107  011150  005767  000776                  TST     ERRFLG
2108  011154  001403                          BEQ     TYPMSG
2109  011156  005767  000030                  TST     DATABP
2110  011162  001007                          BNE     TYPDAT
2111  011164  104402                  TYPMSG: OCTASC
2112  011166  011260                          ERTABO
2113  011170  012767  000001  000754          MOV     #1,ERRFLG
2114  011176  104 01                          TYPE
2115  011200  000000                  ERRMSG: 0
2116  011202  005767  000004          TYPDAT: TST     DATABP
2117  011206  001402                          BEQ     RESREG
2118  011210  104402                          OCTASC
2119  011212  000000                  DATABP: 0
2120  011214  104407                  RESREG: RESO5
2121  011216  005767  166346          HALTS:  TST     SWR
2122  011222  100005                          BPL     EXITER
2123  011224  010046                          PUSHRO
2124  011226  016600  000002                  MOV     2(SP),R0
2125  011232  000000                          HALT
2126  011234  01260C                          POPRO
2127  011236  005267  000714          EXITER: INC     ERRCNT
2128  011242  032767  002000  166320          BIT     #SW10,SWR
2129  011250  001402                          BEQ     1$
2130  011252  016716  000704                  MOV     ESCAPE,(SP)
2131  011256  000002                  1$:     RTI
2132  011260  000001                  ERTABO: 1
2133  011262     006     002                  .BYTE   6,2
2134  011264  012210                          SAVPC
```

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

011266  011646                        TRPSRV: MOV     (SP),-(SP)              ;GET PC OF RETURN
011270  162716  000002                        SUB     #2,(SP)                 ;=PC OF TRAP
011274  017616  000000                        MOV     @(SP),(SP)              ;GET TRP
011300  006316                        TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
011302  042716  177001                        BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
011306  062716  013372                        ADD     #TRPTAB,(SP)            ;POINTER TO SUBROUTINE ADDRESS
011312  017616  000000                        MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
011316  000136                                JMP     @(SP)+                  ;GO TO SUBROUTINE

                                                ;SAVE PC OF TEST THAT FAILED AND R0-R5

011320  016667  000004  000662        SVOSP:  MOV     4(SP),SAVPC

                                                ;SAVE R0-R5

011326  010567  000652                SVOS:   MOV     R5,SAVR5
011332  010467  000644                        MOV     R4,SAVR4
011336  010367  000636                        MOV     R3,SAVR3
011342  010267  000630                        MOV     R2,SAVR2
011346  010167  000622                        MOV     R1,SAVR1
011352  010067  000614                        MOV     R0,SAVR0
011356  000002                                RTI
                                                ;RESTORE R0-R5

011360  016700  000606                RSOS:   MOV     SAVR0,R0
011364  016701  000604                        MOV     SAVR1,R1
011370  016702  000602                        MOV     SAVR2,R2
011374  016703  000600                        MOV     SAVR3,R3
011400  016704  000576                        MOV     SAVR4,R4
011404  016705  000574                        MOV     SAVR5,R5
011410  000002                                RTI
```

```
 2171
 2172                                                ;TELETYPE OUTPUT ROUTINE
 2173
 2174  011412  017605  000000          TYPER:  MOV     @(SP),R5
 2175  011416  062716  000002                  ADD     #2,(SP)
 2176  011422  105777  000466          1$:     TSTB    @TPCSR
 2177  011426  100375                          BPL     1$
 2178  011430  105715                          TSTB    (R5)
 2179  011432  001001                          BNE     2$
 2180  011434  000002                          RTI
 2181  011436  112577  000454          2$:     MOVB    (R5)+,@TPDBR
 2182  011442  000767                          BR      1$
 2183
 2184                                                ;ASCII STRING INPUT ROUTINE
 2185
 2186  011444  017667  000000  000006  INSTRG: MOV     @(SP),MSG
 2187  011452  062716  000002                  ADD     #2,(SP)
 2188  011456  104401                  INSTR1: TYPE
 2189  011460  000000                  MSG:    0
 2190  011462  012704  013414                  MOV     #INBUF,R4
 2191  011466  012703  000007                  MOV     #7,R3
 2192  011472  105777  000412          1$:     TSTB    @TKCSR
 2193  011476  100375                          BPL     1$
 2194  011500  117714  000406                  MOVB    @TKDBR,(R4)
 2195  011504  142714  000200                  BICB    #200,(R4)
 2196  011510  122427  000015                  CMPB    (R4)+,#15
 2197  011514  001413                          BEQ     INSTR2
 2198  011516  117777  000370  000372          MOVB    @TKDBR,@TPDBR
 2199  011524  105777  000364          2$:     TSTB    @TPCSR
 2200  011530  100375                          BPL     2$
 2201  011532  005303                          DEC     R3
 2202  011534  001356                          BNE     1$
 2203  011536  104401                  INSTRE: TYPE
 2204  011540  013246                          MOM
 2205  011542  000745                          BR      INSTR1
 2206  011544  000002                  INSTR2: RTI
```

```
2207
2208                                            ;CONVERT ASCII STRING TO OCTAL
2209
2210  011546  011605                PARAMS: MOV     (SP),R5
2211  011550  012567  000146               MOV     (R5)+,LOLIM
2212  011554  012567  000144               MOV     (R5)+,HILIM
2213  011560  012567  000142               MOV     (R5)+,DEVADR
2214  011564  112567  000140               MOVB    (R5)+,LOBITS
2215  011570  112567  000135               MOVB    (R5)+,ADRCNT
2216  011574  010516                       MOV     R5,(SP)
2217  011576  005005                PARAM1: CLR     R5
2218  011600  012704  013414               MOV     #INBUF,R4
2219  011604  122714  000015               CMPB    #15,(R4)
2220  011610  001420                       BEQ     PARERR
2221  011612  121427  000060         1$:   CMPB    (R4),#60
2222  011616  002415                       BLT     PARERR
2223  011620  121427  000067               CMPB    (R4),#67
2224  011624  003012                       BGT     PARERR
2225  011626  142714  000060               BICB    #60,(R4)
2226  011632  152405                       BISB    (R4)+,R5
2227  011634  122714  000015               CMPB    #15,(R4)
2228  011640  001406                       BEQ     LIMITS
2229  011642  006305                       ASL     R5
2230  011644  006305                       ASL     R5
2231  011646  006305                       ASL     R5
2232  011650  000760                       BR      1$
2233  011652  104404                PARERR: INSTER
2234  011654  000750                       BR      PARAM1
2235
2236                                            ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
2237
2238  011656  020567  000042         LIMITS: CMP     R5,HILIM
2239  011662  101373                       BHI     PARERR
2240  011664  020567  000032               CMP     R5,LOLIM
2241  011670  103770                       BLO     PARERR
2242  011672  136705  000032               BITB    LOBITS,R5
2243  011676  001365                       BNE     PARERR
2244
2245                                            ;STORE NUMBER AT SPECIFIED ADDRESS
2246
2247  011700  016704  000022               MOV     DEVADR,R4
2248  011704  010524                1$:    MOV     R5,(R4)+
2249  011706  062705  000002               ADD     #2,R5
2250  011712  105367  000013               DECB    ADRCNT
2251  011716  001372                       BNE     1$
2252  011720  000002                       RTI
2253  011722  000000                LOLIM:  0
2254  011724  000000                HILIM:  0
2255  011726  000000                DEVADR: 0
2256  011730  000000                LOBITS: 0
2257          011731                ADRCNT=LOBITS+1
```

```
2258
2259                                        ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
2260
2261  011732  104401              OCTASN:   TYPE
2262  011734  013252                        MCRLF
2263  011736  017601  000000                MOV     @(SP),R1
2264  011742  062716  000002                ADD     #2,(SP)
2265  011746  012167  000130                MOV     (R1)+,WRDCNT
2266  011752  112167  000126        1$:     MOVB    (R1)+,CHRCNT
2267  011756  112167  000123                MOVB    (R1)+,SPACNT
2268  011762  013167  000120                MOV     @(R1)+,BINWRD
2269  011766  016704  000114        2$:     MOV     BINWRD,R4
2270  011772  116705  000106                MOVB    CHRCNT,R5
2271  011776  012700  013426                MOV     #TEMP,R0
2272  012002  010403                3$:     MOV     R4,R3
2273  012004  042703  177770                BIC     #177770,R3
2274  012010  062703  000260                ADD     #260,R3
2275  012014  110320                        MOVB    R3,(R0)+
2276  012016  006204                        ASR     R4
2277  012020  006204                        ASR     R4
2278  012022  006204                        ASR     R4
2279  012024  005305                        DEC     R5
2280  012026  001365                        BNE     3$
2281  012030  012703  013440                MOV     #MDATA,R3
2282  012034  114023                4$:     MOVB    -(R0),(R3)+
2283  012036  105367  000042                DECB    CHRCNT
2284  012042  001374                        BNE     4$
2285  012044  105767  000035                TSTB    SPACNT
2286  012050  001405                        BEQ     6$
2287  012052  112723  000240        5$:     MOVB    #240,(R3)+
2288  012056  105367  000023                DECB    SPACNT
2289  012062  001373                        BNE     5$
2290  012064  105013                6$:     CLRB    (R3)
2291  012066  104401                        TYPE
2292  012070  013440                        MDATA
2293  012072  005367  000004                DEC     WRDCNT
2294  012076  001325                        BNE     1$
2295  012100  000002                        RTI
2296  012102  000000                WRDCNT: 0
2297  012104  000000                CHRCNT: 0
2298          012105                SPACNT=CHRCNT+1
2299  012106  000000                BINWRD: 0
```

```
2300                                            ;INDIRECT POINTERS
2301
2302  012110  177560                TKCSR:  177560
2303  012112  177562                TKDBR:  177562
2304  012114  177564                TPCSR:  177564
2305  012116  177566                TPDBR:  177566
2306  012120  000000                DHSCR:  0
2307  012122  000000                DHNRC:  0
2308  012124  000000                DHLPR:  0
2309  012126  000000                DHBA:   0
2310  012130  000000                DHBC:   0
2311  012132  000000                DHBAR:  0
2312  012134  000000                DHBCR:  0
2313  012136  000000                DHSSR:  0
2314  012140  000000                DHSLR:  0
2315  012142  000000                DHRVEC: 0
2316  012144  000000                DHRLVL: 0
2317  012146  000000                DHTVEC: 0
2318  012150  000000                DHTLVL: 0
2319                                            ;PROGRAM VARIABLES
2320
2321  012152  000000                ERRFLG: 0               ;ERROR FLAG
2322  012154  000000                PASCNT: 0               ;PASS COUNT
2323  012156  000000                ERRCNT: 0               ;ERROR COUNT
2324  012160  000000                RETURN: 0               ;SCOPE RETURN ADDRESS FOR TEST LOOPING
2325  012162  000000                ESCAPE: 0               ;ADDRESS FOR ERROR ESCAPE
2326  012164  000000                FREEZI: 0               ;DATA LOOPING RETURN ADDRESS
2327  012166  000000                ICOUNT: 0               ;ITERATION COUNT FOR TEST IN PROGRESS
2328  012170  000000                LPCNT:  0               ;NUMBER OF ITERATIONS THIS TEST
2329  012172  000000                SAVR0:  0               ;R0 SAVE AREA
2330  012174  000000                SAVR1:  0               ;R1 SAVE AREA
2331  012176  000000                SAVR2:  0               ;R2 SAVE AREA
2332  012200  000000                SAVR3:  0               ;R3 SAVE ARE
2333  012202  000000                SAVR4:  0               ;R4 SAVE AREA
2334  012204  000000                SAVR5:  0               ;R5 SAVE AREA
2335  012206  000000                SAVSP:  0               ;STACK POINTER SAVE AREA
2336  012210  000000                SAVPC:  0               ;CALLING ROUTINE SAVE AREA
2337  012212  000000                INIFLG: 0               ;PROGRAM INITIALIZATION FLAG
2338  012214  000000                STFLG:  0               ;PROGRAM START FLAG
2339  012216  000000                LAST:   0               ;LAST ERROR PC
2340  012220  000000                ENDFLG: 0
2341  012222  000000                NULL:   0
2342  012224  000000                LINACT: 0
```

```
 2343
 2344                                                ;CLEAR ALL BYTE COUNT AND BUS ADDRESS REGISTERS
 2345
 2346  012226  012700  000020          CLRALL: MOV     #20,R0
 2347  012232  005077  177670          1$:     CLR     @DHBA
 2348  012236  005077  177666                  CLR     @DHBC
 2349  012242  005277  177652                  INC     @DHSCR
 2350  012246  005300                          DEC     R0
 2351  012250  001370                          BNE     1$
 2352  012252  000207                          RTS     PC
 2353  012254     000                  TBUF:   .BYTE   0
 2354  012255     001                  .BYTE   TDAT
 2355  012256     002                  .BYTE   TDAT
 2356  012257     003                  .BYTE   TDAT
 2357  012260     004                  .BYTE   TDAT
 2358  012261     005                  .BYTE   TDAT
 2359  012262     006                  .BYTE   TDAT
 2360  012263     007                  .BYTE   TDAT
 2361  012264     010                  .BYTE   TDAT
 2362  012265     011                  .BYTE   TDAT
 2363  012266     012                  .BYTE   TDAT
 2364  012267     013                  .BYTE   TDAT
 2365  012270     014                  .BYTE   TDAT
 2366  012271     015                  .BYTE   TDAT
 2367  012272     016                  .BYTE   TDAT
 2368  012273     017                  .BYTE   TDAT
 2369  012274     020                  .BYTE   TDAT
 2370  012275     021                  .BYTE   TDAT
 2371  012276     022                  .BYTE   TDAT
 2372  012277     023                  .BYTE   TDAT
 2373  012300     024                  .BYTE   TDAT
 2374  012301     025                  .BYTE   TDAT
 2375  012302     026                  .BYTE   TDAT
 2376  012303     027                  .BYTE   TDAT
 2377  012304     030                  .BYTE   TDAT
 2378  012305     031                  .BYTE   TDAT
 2379  012306     032                  .BYTE   TDAT
 2380  012307     033                  .BYTE   TDAT
 2381  012310     034                  .BYTE   TDAT
 2382  012311     035                  .BYTE   TDAT
 2383  012312     036                  .BYTE   TDAT
 2384  012313     037                  .BYTE   TDAT
 2385  012314     040                  .BYTE   TDAT
 2386  012315     041                  .BYTE   TDAT
 2387  012316     042                  .BYTE   TDAT
 2388  012317     043                  .BYTE   TDAT
 2389  012320     044                  .BYTE   TDAT
 2390  012321     045                  .BYTE   TDAT
 2391  012322     046                  .BYTE   TDAT
 2392  012323     047                  .BYTE   TDAT
 2393  012324     050                  .BYTE   TDAT
 2394  012325     051                  .BYTE   TDAT
 2395  012326     052                  .BYTE   TDAT
 2396  012327     053                  .BYTE   TDAT
 2397  012330     054                  .BYTE   TDAT
 2398  012331     055                  .BYTE   TDAT
```

```
 2399  012332  056                    .BYTE   TDAT
 2400  012333  057                    .BYTE   TDAT
 2401  012334  060                    .BYTE   TDAT
 2402  012335  061                    .BYTE   TDAT
 2403  012336  062                    .BYTE   TDAT
 2404  012337  063                    .BYTE   TDAT
 2405  012340  064                    .BYTE   TDAT
 2406  012341  065                    .BYTE   TDAT
 2407  012342  066                    .BYTE   TDAT
 2408  012343  067                    .BYTE   TDAT
 2409  012344  070                    .BYTE   TDAT
 2410  012345  071                    .BYTE   TDAT
 2411  012346  072                    .BYTE   TDAT
 2412  012347  073                    .BYTE   TDAT
 2413  012350  074                    .BYTE   TDAT
 2414  012351  075                    .BYTE   TDAT
 2415  012352  076                    .BYTE   TDAT
 2416  012353  077                    .BYTE   TDAT
 2417  012354  100                    .BYTE   TDAT
 2418  012355  101                    .BYTE   TDAT
 2419  012356  102                    .BYTE   TDAT
 2420  012357  103                    .BYTE   TDAT
 2421  012360  104                    .BYTE   TDAT
 2422  012361  105                    .BYTE   TDAT
 2423  012362  106                    .BYTE   TDAT
 2424  012363  107                    .BYTE   TDAT
 2425  012364  110                    .BYTE   TDAT
 2426  012365  111                    .BYTE   TDAT
 2427  012366  112                    .BYTE   TDAT
 2428  012367  113                    .BYTE   TDAT
 2429  012370  114                    .BYTE   TDAT
 2430  012371  115                    .BYTE   TDAT
 2431  012372  116                    .BYTE   TDAT
 2432  012373  117                    .BYTE   TDAT
 2433  012374  120                    .BYTE   TDAT
 2434  012375  121                    .BYTE   TDAT
 2435  012376  122                    .BYTE   TDAT
 2436  012377  123                    .BYTE   TDAT
 2437  012400  124                    .BYTE   TDAT
 2438  012401  125                    .BYTE   TDAT
 2439  012402  126                    .BYTE   TDAT
 2440  012403  127                    .BYTE   TDAT
 2441  012404  130                    .BYTE   TDAT
 2442  012405  131                    .BYTE   TDAT
 2443  012406  132                    .BYTE   TDAT
 2444  012407  133                    .BYTE   TDAT
 2445  012410  134                    .BYTE   TDAT
 2446  012411  135                    .BYTE   TDAT
 2447  012412  136                    .BYTE   TDAT
 2448  012413  137                    .BYTE   TDAT
 2449  012414  140                    .BYTE   TDAT
 2450  012415  141                    .BYTE   TDAT
 2451  012416  142                    .BYTE   TDAT
 2452  012417  143                    .BYTE   TDAT
 2453  012420  144                    .BYTE   TDAT
 2454  012421  145                    .BYTE   TDAT
```

```
 2455  012422     146                .BYTE   TDAT
 2456  012423     147                .BYTE   TDAT
 2457  012424     150                .BYTE   TDAT
 2458  012425     151                .BYTE   TDAT
 2459  012426     152                .BYTE   TDAT
 2460  012427     153                .BYTE   TDAT
 2461  012430     154                .BYTE   TDAT
 2462  012431     155                .BYTE   TDAT
 2463  012432     156                .BYTE   TDAT
 2464  012433     157                .BYTE   TDAT
 2465  012434     160                .BYTE   TDAT
 2466  012435     161                .BYTE   TDAT
 2467  012436     162                .BYTE   TDAT
 2468  012437     163                .BYTE   TDAT
 2469  012440     164                .BYTE   TDAT
 2470  012441     165                .BYTE   TDAT
 2471  012442     166                .BYTE   TDAT
 2472  012443     167                .BYTE   TDAT
 2473  012444     170                .BYTE   TDAT
 2474  012445     171                .BYTE   TDAT
 2475  012446     172                .BYTE   TDAT
 2476  012447     173                .BYTE   TDAT
 2477  012450     174                .BYTE   TDAT
 2478  012451     175                .BYTE   TDAT
 2479  012452     176                .BYTE   TDAT
 2480  012453     177                .BYTE   TDAT
 2481  012454     200                .BYTE   TDAT
 2482  012455     201                .BYTE   TDAT
 2483  012456     202                .BYTE   TDAT
 2484  012457     203                .BYTE   TDAT
 2485  012460     204                .BYTE   TDAT
 2486  012461     205                .BYTE   TDAT
 2487  012462     206                .BYTE   TDAT
 2488  012463     207                .BYTE   TDAT
 2489  012464     210                .BYTE   TDAT
 2490  012465     211                .BYTE   TDAT
 2491  012466     212                .BYTE   TDAT
 2492  012467     213                .BYTE   TDAT
 2493  012470     214                .BYTE   TDAT
 2494  012471     215                .BYTE   TDAT
 2495  012472     216                .BYTE   TDAT
 2496  012473     217                .BYTE   TDAT
 2497  012474     220                .BYTE   TDAT
 2498  012475     221                .BYTE   TDAT
 2499  012476     222                .BYTE   TDAT
 2500  012477     223                .BYTE   TDAT
 2501  012500     224                .BYTE   TDAT
 2502  012501     225                .BYTE   TDAT
 2503  012502     226                .BYTE   TDAT
 2504  012503     227                .BYTE   TDAT
 2505  012504     230                .BYTE   TDAT
 2506  012505     231                .BYTE   TDAT
 2507  012506     232                .BYTE   TDAT
 2508  012507     233                .BYTE   TDAT
 2509  012510     234                .BYTE   TDAT
 2510  012511     235                .BYTE   TDAT
```

```
 2511  012512  236                .BYTE   TDAT
 2512  012513  237                .BYTE   TDAT
 2513  012514  240                .BYTE   TDAT
 2514  012515  241                .BYTE   TDAT
 2515  012516  242                .BYTE   TDAT
 2516  012517  243                .BYTE   TDAT
 2517  012520  244                .BYTE   TDAT
 2518  012521  245                .BYTE   TDAT
 2519  012522  246                .BYTE   TDAT
 2520  012523  247                .BYTE   TDAT
 2521  012524  250                .BYTE   TDAT
 2522  012525  251                .BYTE   TDAT
 2523  012526  252                .BYTE   TDAT
 2524  012527  253                .BYTE   TDAT
 2525  012530  254                .BYTE   TDAT
 2526  012531  255                .BYTE   TDAT
 2527  012532  256                .BYTE   TDAT
 2528  012533  257                .BYTE   TDAT
 2529  012534  260                .BYTE   TDAT
 2530  012535  261                .BYTE   TDAT
 2531  012536  262                .BYTE   TDAT
 2532  012537  263                .BYTE   TDAT
 2533  012540  264                .BYTE   TDAT
 2534  012541  265                .BYTE   TDAT
 2535  012542  266                .BYTE   TDAT
 2536  012543  267                .BYTE   TDAT
 2537  012544  270                .BYTE   TDAT
 2538  012545  271                .BYTE   TDAT
 2539  012546  272                .BYTE   TDAT
 2540  012547  273                .BYTE   TDAT
 2541  012550  274                .BYTE   TDAT
 2542  012551  275                .BYTE   TDAT
 2543  012552  276                .BYTE   TDAT
 2544  012553  277                .BYTE   TDAT
 2545  012554  300                .BYTE   TDAT
 2546  012555  301                .BYTE   TDAT
 2547  012556  302                .BYTE   TDAT
 2548  012557  303                .BYTE   TDAT
 2549  012560  304                .BYTE   TDAT
 2550  012561  305                .BYTE   TDAT
 2551  012562  306                .BYTE   TDAT
 2552  012563  307                .BYTE   TDAT
 2553  012564  310                .BYTE   TDAT
 2554  012565  311                .BYTE   TDAT
 2555  012566  312                .BYTE   TDAT
 2556  012567  313                .BYTE   TDAT
 2557  012570  314                .BYTE   TDAT
 2558  012571  315                .BYTE   TDAT
 2559  012572  316                .BYTE   TDAT
 2560  012573  317                .BYTE   TDAT
 2561  012574  320                .BYTE   TDAT
 2562  012575  321                .BYTE   TDAT
 2563  012576  322                .BYTE   TDAT
 2564  012577  323                .BYTE   TDAT
 2565  012600  324                .BYTE   TDAT
 2566  012601  325                .BYTE   TDAT
```

```
2567  012602  326                 .BYTE   TDAT
2568  012603  327                 .BYTE   TDAT
2569  012604  330                 .BYTE   TDAT
2570  012605  331                 .BYTE   TDAT
2571  012606  332                 .BYTE   TDAT
2572  012607  333                 .BYTE   TDAT
2573  012610  334                 .BYTE   TDAT
2574  012611  335                 .BYTE   TDAT
2575  012612  336                 .BYTE   TDAT
2576  012613  337                 .BYTE   TDAT
2577  012614  340                 .BYTE   TDAT
2578  012615  341                 .BYTE   TDAT
2579  012616  342                 .BYTE   TDAT
2580  012617  343                 .BYTE   TDAT
2581  012620  344                 .BYTE   TDAT
2582  012621  345                 .BYTE   TDAT
2583  012622  346                 .BYTE   TDAT
2584  012623  347                 .BYTE   TDAT
2585  012624  350                 .BYTE   TDAT
2586  012625  351                 .BYTE   TDAT
2587  012626  352                 .BYTE   TDAT
2588  012627  353                 .BYTE   TDAT
2589  012630  354                 .BYTE   TDAT
2590  012631  355                 .BYTE   TDAT
2591  012632  356                 .BYTE   TDAT
2592  012633  357                 .BYTE   TDAT
2593  012634  360                 .BYTE   TDAT
2594  012635  361                 .BYTE   TDAT
2595  012636  362                 .BYTE   TDAT
2596  012637  363                 .BYTE   TDAT
2597  012640  364                 .BYTE   TDAT
2598  012641  365                 .BYTE   TDAT
2599  012642  366                 .BYTE   TDAT
2600  012643  367                 .BYTE   TDAT
2601  012644  370                 .BYTE   TDAT
2602  012645  371                 .BYTE   TDAT
2603  012646  372                 .BYTE   TDAT
2604  012647  373                 .BYTE   TDAT
2605  012650  374                 .BYTE   TDAT
2606  012651  375                 .BYTE   TDAT
2607  012652  376                 .BYTE   TDAT
2608  012653  377                 .BYTE   TDAT
2609                              .EVEN
2610  012654  000000              RBUF:   0
2611          012716              .=.+40
```

```
  2612  012716  120000                  RWRD0:  120000
  2613  012720  120400                  RWRD1:  120400
  2614  012722  121000                  RWRD2:  121000
  2615  012724  121400                  RWRD3:  121400
  2616  012726  122000                  RWRD4:  122000
  2617  012730  122400                  RWRD5:  122400
  2618  012732  123000                  RWRD6:  123000
  2619  012734  123400                  RWRD7:  123400
  2620  012736  124000                  RWRD10: 124000
  2621  012740  124400                  RWRD11: 124400
  2622  012742  125000                  RWRD12: 125000
  2623  012744  125400                  RWRD13: 125400
  2624  012746  126000                  RWRD14: 126000
  2625  012750  126400                  RWRD15: 126400
  2626  012752  127000                  RWRD16: 127000
  2627  012754  127400                  RWRD17: 127400
```

```
2628                                            ;ENTER HERE ON POWER FAILURE
2629
2630
2631  012756  010046                  PFAIL:    MOV     R0,-(SP)              ;SAVE R0-R5 ON PROCESSOR STACK
2632  012760  010146                            MOV     R1,-(SP)
2633  012762  010246                            MOV     R2,-(SP)
2634  012764  010346                            MOV     R3,-(SP)
2635  012766  010446                            MOV     R4,-(SP)
2636  012770  010546                            MOV     R5,-(SP)
2637  012772  016746  165026                    MOV     24,-(SP)
2638  012776  010667  177204                    MOV     SP,SAVSP              ;SAVE STACK POINTER
2639  013002  012767  013014  165014            MOV     #RESTART,24           ;SET UP FOR POWER UP TRAP
2640  013010  000000                            HALT                          ;HALT ON POWER DOWN NORMAL
2641  013012  000777                            BR      .
2642
2643                                            ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
2644
2645  013014  016706  177166          RESTAR:   MOV     SAVSP,SP              ;RESTORE STACK POINTER
2646  013020  012605                            MOV     (SP)+,R5              ;RESTORE R0-R5
2647  013022  012604                            MOV     (SP)+,R4
2648  013024  012603                            MOV     (SP)+,R3
2649  013026  012602                            MOV     (SP)+,R2
2650  013030  012601                            MOV     (SP)+,R1
2651  013032  012600                            MOV     (SP)+,R0
2652  013034  012767  012756  164762            MOV     #PFAIL,24             ;SET UP FOR POWER FAILURE
2653  013042  012767  000340  164726            MOV     #340,PS
2654  013050  012706  014040                    MOV     #STACK,SP
2655  013054  005067  000346                    CLR     TEMP
2656  013060  005267  000342                    INC     TEMP
2657  013064  001375                            BNE     .-4
2658  013066  104402                            OCTASC
2659  013070  013112                            PFTAB
2660  013072  104401                            TYPE
2661  013074  013255                            MPFAIL
2662  013076  005067  177050                    CLR     ERRFLG
2663  013102  005067  177110                    CLR     LAST
2664  013106  000177  177046                    JMP     @RETURN
2665  013112  000001                  PFTAB:    1
2666  013114  000006  000002                    6,2
2667  013120  000207                            RETURN
```

```
2668  013122  005015  042012  030510  MTITLE: .ASCIZ  <15><12><12>/DH11 HALF-DUPLEX AND BREAK TEST /<15><12>
2669  013130  020061  040510  043114
2670  013136  042055  050125  042514
2671  013144  020120  047101  020104
2672  013152  051102  040505  020113
2673  013160  042524  052123  006440
2674  013166  000012
2675  013170  005015  042526  052103  MVECTO: .ASCIZ  <15><12>/VECTOR ADDRESS-/
2676  013176  051117  040440  042104
2677  013204  042522  051523  000055
2678  013212  005015  047503  052116  MREGAD: .ASCIZ  <15><12>/CONTROL REGISTER ADDRESS-/
2679  013220  047522  020114  042522
2680  013226  044507  052123  051105
2681  013234  040440  042104  042522
2682  013242  051523  000055
2683  013246  020040  000077          MQM:    .ASCIZ  / ?/
2684  013252  005015  000             MCRLF:  .ASCIZ  <15><12>
2685  013256  000040  050040  053517  MPFAIL: .ASCIZ  /  POWER FAILURE, PROGRAM RESTART AT TEST IN PROGRESS/
2686  013264  051105  043040  044501
2687  013272  052514  042522  020054
2688  013276  051120  043517  040522
2689  013304  020115  042522  052123
2690  013312  051101  020124  052101
2691  013320  052040  051505  020124
2692  013326  047111  050040  047522
2693  013334  051107  051505  000123
2694  013342  005015  055104  044104  MEPASS: .ASCIZ  <15><12>/DZDHI/
2695  013350  000111
2696  013352  005015  000122          MR:     .ASCIZ  <15><12>/R/
2697  013356  005015  042524  052123  MTSTPC: .ASCIZ  <15><12>/TEST PC-/
2698  013364  050040  026503  000
2699          013372                  .EVEN
2700
2701                                  ;TABLE OF POINTERS FOR TRAP DECODING
2702
2703  013372  010746                  TRPTAB: SCOPER
2704  013374  011412                          TYPER
2705  013376  011732                          OCTASN
2706  013400  011444                          INSTRG
2707  013402  011536                          INSTRE
2708  013404  011546                          PARAMS
2709  013406  011320                          SVOSP
2710  013410  011360                          RSOS
2711  013412  011046                          SCOP1R
2712
2713                                  ;BUFFERS FOR INPUT-OUTPUT
2714
2715  013414  000000                  INBUF:  0
2716          013426                  .=.+10
2717  013426  000000                  TEMP:   0
2718          013440                  .=.+10
2719  013440  000000                  MDATH:  0
2720          013452                  .=.+10
2721
2722                                  ;TABLE OF POINTERS TO ERROR MESSAGES AND DATA
2723
```

```
 2724  013452                          ERRTAB:
 2725  013452  013466                          EM1
 2726  013454  013626                          DT1
 2727  013456  013542                          EM2
 2728  013460  013626                          DT1
 2729  013462  013600                          EM3
 2730  013464  000000                          0
 2731  013466  047515  042522  052040  EM1:    .ASCIZ  /MORE THAN 1 CHARACTER RECEIVED/<15><12>/EXP     REC/
 2732  013474  040510  020116  020061
 2733  013502  044103  051101  041501
 2734  013510  042524  020122  042522
 2735  013516  042503  053111  042105
 2736  013524  005015  054105  020120
 2737  013532  020040  020040  042522
 2738  013540  000103
 2739  013542  051102  040505  020113  EM2:    .ASCIZ  /BREAK DATA ERROR/<15><12>/EXP     REC/
 2740  013550  040504  040524  042440
 2741  013556  051122  051117  005015
 2742  013564  054105  020120  020040
 2743  013572  020040  042522  000103
 2744  013600  042522  042503  053111  EM3:    .ASCIZ  /RECEIVER NOT BLINDED/
 2745  013606  051105  047040  052117
 2746  013614  041040  044514  042116
 2747  013622  042105     000
 2748          013626                  .EVEN
 2749  013626  000002                  DT1:    2
 2750  013630     006  002             .BYTE   6,2
 2751  013632  012204                          SAVR5
 2752  013634     006  002             .BYTE   6,2
 2753  013636  012202                          SAVR4
 2754  013640  000000                  ENDCOD: 0
 2755          000001                  .END
```

```
ADRCNT= 011731           2215*  2250*  2257#
BEGIN   001202            878    907    913#  2058
BINWRD  012106           2268*  2269   2299#
BITX  = 000000            933#   977#  1021#  1065#  1109#  1153#  1197#  1241#  1285#  1329#  1373#  1417#  1461#
                         1505#  1549#  1593#  1637#  1662#  1687#  1712#  1737#  1762#  1787#  1812#  1837#  1862#
                         1887#  1912#  1937#  1962#  1987#  2012#  2037#
BIT00 = 000001            564#
BIT01 = 000002            563#
BIT02 = 000004            562#
BIT03 = 000010            561#
BIT04 = 000020            560#
BIT05 = 000040            559#
BIT06 = 000100            558#
BIT07 = 000200            557#
BIT08 = 000400            556#
BIT09 = 001000            555#
BIT10 = 002000            554#
BIT11 = 004000            553#   945    956    989   1000   1033   1044   1077   1088   1121   1132   1165   1176
                         1209   1220   1253   1264   1297   1308   1341   1352   1385   1396   1429   1440   1473
                         1484   1517   1528   1561   1572   1605   1616   1646   1671   1696   1721   1746   1771
                         1796   1821   1846   1871   1896   1921   1946   1971   1996   2021
BIT12 = 010000            552#
BIT13 = 020000            551#
BIT14 = 040000            550#
BIT15 = 100000            549#
CHRCNT  012104           2266*  2270   2283*  2297#  2298
CLRALL  012226            946    990   1034   1078   1122   1166   1210   1254   1298   1342   1386   1430   1474
                         1518   1562   1606   1647   1672   1697   1722   1747   1772   1797   1822   1847   1872
                         1897   1922   1947   1972   1997   2022   2346#
DATA  = 127400           2612#
DATABP  011212           2106*  2109   2116   2119#
DATAX = 127400           2612#  2613#  2614#  2615#  2616#  2617#  2618#  2619#  2620#  2621#  2622#  2623#  2624#
                         2625#  2626#  2627#
DEVADR  011726           2213*  2247   2255#
DHBA    012126            949*   958*   993*  1002*  1037*  1046*  1081*  1090*  1125*  1134*  1169*  1178*  1213*
                         1222*  1257*  1266*  1301*  1310*  1345*  1354*  1389*  1398*  1433*  1442*  1477*  1486*
                         1521*  1530*  1565*  1574*  1609*  1618*  1650*  1675*  1700*  1725*  1750*  1775*  1800*
                         1825*  1850*  1875*  1900*  1925*  1950*  1975*  2000*  2025*  2309#  2347#
DHBAR   012132            953*   962*   963    997*  1006*  1007   1041*  1050*  1051   1085*  1094*  1095   1129*
                         1138*  1139   1173*  1182*  1183   1217*  1226*  1227   1261*  1270*  1271   1305*  1314*
                         1315   1349*  1358*  1359   1393*  1402*  1403   1437*  1446*  1447   1481*  1490*  1491
                         1525*  1534*  1535   1569*  1578*  1579   1613*  1622*  1623   1655*  1656   1680*  1681
                         1705*  1706   1730*  1731   1755*  1756   1780*  1781   1805*  1806   1830*  1831   1855*
                         1856   1880*  1881   1905*  1906   1930*  1931   1955*  1956   1980*  1981   2005*  2006
                         2030*  2031   2311#
DHBC    012130            950*   959*   994*  1003*  1038*  1047*  1082*  1091*  1126*  1135*  1170*  1179*  1214*
                         1223*  1258*  1267*  1302*  1311*  1346*  1355*  1390*  1399*  1434*  1443*  1478*  1487*
                         1522*  1531*  1566*  1575*  1610*  1619*  1651*  1676*  1701*  1726*  1751*  1776*  1801*
                         1826*  1851*  1876*  1901*  1926*  1951*  1976*  2001*  2026*  2310#  2348#
DHBCR   012134            961*  1005*  1049*  1093*  1137*  1181*  1225*  1269*  1313*  1357*  1401*  1445*  1489*
                         1533*  1577*  1621*  2312#
DHLPR   012124            951*   960*   995*  1004*  1039*  1048*  1083*  1092*  1127*  1136*  1171*  1180*  1215*
                         1224*  1259*  1268*  1303*  1312*  1347*  1356*  1391*  1400*  1435*  1444*  1479*  1488*
                         1523*  1532*  1567*  1576*  1611*  1620*  1652*  1677*  1702*  1727*  1752*  1777*  1802*
                         1827*  1852*  1877*  1902*  1927*  1952*  1977*  2002*  2027*  2308#
DHNRC   012122            971   1015   1059   1103   1147   1191   1235   1279   1323   1367   1411   1455   1499
```

```
                    1543    1587    1631    2307#
DHRLVL  012144      2316#
DHRVEC  012142       893    2315#
DHSCR   012120       901     942     945*    948*    956*    957*    986     989*    992*   1000*   1001*   1030    1033*
                    1036*   1044*   1045*   1074    1077*   1080*   1088*   1089*   1118    1121*   1124*   1132*   1133*
                    1162    1165*   1168*   1176*   1177*   1206    1209*   1212*   1220*   1221*   1250    1253*   1256*
                    1264*   1265*   1294    1297*   1300*   1308*   1309*   1338    1341*   1344*   1352*   1353*   1382
                    1385*   1388*   1396*   1397*   1426    1429*   1432*   1440*   1441*   1470    1473*   1476*   1484*
                    1485*   1514    1517*   1520*   1528*   1529*   1558    1561*   1564*   1572*   1573*   1602    1605*
                    1608*   1616*   1617*   1643    1646*   1649*   1658    1668    1671*   1674*   1683    1693    1696*
                    1699*   1708    1718    1721*   1724*   1733    1743    1746*   1749*   1758    1768    1771*   1774*
                    1783    1793    1796*   1799*   1808    1818    1821*   1824*   1833    1843    1846*   1849*   1858
                    1868    1871*   1874*   1883    1893    1896*   1899*   1908    1918    1921*   1924*   1933    1943
                    1946*   1949*   1958    1968    1971*   1974*   1983    1993    1996*   1999*   2008    2018    2021*
                    2024*   2033    2306#   2349*
DHSLR   012140       904*    905*    954     965     998    1009    1042    1053    1086    1097    1130    1141    1174
                    1185    1218    1229    1262    1273    1306    1317    1350    1361    1394    1405    1438    1449
                    1482    1493    1526    1537    1570    1581    1614    1625    2314#
DHSSR   012136       904     967    1011    1055    1099    1143    1187    1231    1275    1319    1363    1407    1451
                    1495    1539    1583    1627    2313#
DHTLVL  012150      2318#
DHTVEC  012146      2317#
DT1     013626      2726    2728    2749#
EM1     013466      2725    2731#
EM2     013542      2727    2739#
EM3     013600      2729    2744#
ENDCOD  013640       536    2754#
ENDFLG  012220      2340#
EOP     010674      2045#
ERRCNT  012156       869*   2127*   2323#
ERRFLG  012152       870*    871*   2048*   2073*   2078    2096*   2107    2113*   2321#   2662*
ERRMSG  011200      2105*   2115#
ERRORS  011064       829    2091#
ERRTAB  013452      2104    2724#
ERTABO  011250      2112    2132#
ESCAPE  012162       943*    987*   1031*   1075*   1119*   1163*   1207*   1251*   1295*   1339*   1383*   1427*   1471*
                    1515*   1559*   1603*   1644*   1669*   1694*   1719*   1744*   1769*   1794*   1819*   1844*   1869*
                    1894*   1919*   1944*   1969*   1994*   2019*   2130    2325#
EXITER  011236      2122    2127#
FREEZI  012164       944*    988*   1032*   1076*   1120*   1164*   1208*   1252*   1296*   1340*   1384*   1428*   1472*
                    1516*   1560*   1604*   1645*   1670*   1695*   1720*   1745*   1770*   1795*   1820*   1845*   1870*
                    1895*   1920*   1945*   1970*   1995*   2020*   2086    2326#
HALTS   011216      2092    2121#
HILIM   011724      2212*   2238    2254#
ICOUNT  012166       942*    986*   1030*   1074*   1118*   1162*   1206*   1250*   1294*   1338*   1382*   1426*   1470*
                    1514*   1558*   1602*   1643*   1668*   1693*   1718*   1743*   1768*   1793*   1818*   1843*   1868*
                    1893*   1918*   1943*   1968*   1993*   2018*   2070    2327#
INBUF   013414      2190    2218    2715#
INIFLG  012212       673     906     908*   2337#
INSTER= 104404       846#   2233
INSTR = 104403       845#    888     896     917
INSTRE  011536      2203#   2707
INSTRG  011444      2186#   2706
INSTRI  011456      2188#   2205
INSTR2  011544      2197    2206#
K     = 000020       933#   1637#   2612#
```

```
KX     = 000020          933#   977#  1021#  1065#  1109#  1153#  1197#  1241#  1285#  1329#  1373#  1417#  1461#
                        1505#  1549#  1593#  1637#  1662#  1687#  1712#  1737#  1762#  1787#  1812#  1837#  1862#
                        1887#  1912#  1937#  1962#  1987#  2012#  2037#  2612#  2613#  2614#  2615#  2616#  2617#
                        2618#  2619#  2620#  2621#  2622#  2623#  2624#  2625#  2626#  2627#  2628#
LAST     012216        2047*  2093   2095*  2339#  2663*
LIGHTS=  177570         534#  2050*
LIMITS   011656        2228   2238#
LINACT   012224        2342#
LINE   = 000020          933#   977#  1021#  1065#  1109#  1153#  1197#  1241#  1285#  1329#  1373#  1417#  1461#
                        1505#  1549#  1593#  1637#  1662#  1687#  1712#  1737#  1762#  1787#  1812#  1837#  1862#
                        1887#  1912#  1937#  1962#  1987#  2012#  2037#
LOBITS   011730        2214*  2242   2256#  2257
LOGICA   010732         852   2054#
LOLIM    011722        2211*  2240   2253#
LPCNT    012170        2069*  2070   2072*  2328#
MCRLF    013252        2262   2684#
MDATA    013440        2281   2292   2719#
MEPASS   013342        2046   2694#
MPFAIL   013255        2661   2685#
MQM      013246        2204   2683#
MR       013352         930   2696#
MREGAD   013212         897   2678#
MSG      011460        2186*  2189#
MTITLE   013122         872   2668#
MTSTPC   013356         918   2697#
MVECTO   013170         889   2675#
N      = 000001           1#
NULL     012222         949    993   1037   1081   1125   1169   1213   1257   1301   1345   1389   1433   1477
                       1521   1565   1609   2341#
OCTASC=  104402         844#  2111   2118   2658
OCTASN   011732        2261#  2705
PARAM  = 104405         847#   890    898    919
PARAMS   011546        2210#  2708
PARAM1   011576        2217#  2234
PARERR   011652        2220   2222   2224   2233#  2239   2241   2243
PASCNT   012154         868*  2049*  2050   2322#
PC     =%000007         529#   946*   990*  1034*  1078*  1122*  1166*  1210*  1254*  1298*  1342*  1386*  1430*
                       1474*  1518*  1562*  1606*  1647*  1672*  1697*  1722*  1747*  1772*  1797*  1822*  1847*
                       1872*  1897*  1922*  1947*  1972*  1997*  2022*  2054*  2352*
PFAIL    012756         827    866   2631#  2652
PFTAB    013112        2659   2665#
POPRO  = 012600         543#  2126
POP1SP=  005726         541#
POP2SP=  022626         545#
PS     = 177776         535#   864*   913*   941*   985*  1029*  1073*  1117*  1161*  1205*  1249*  1293*  1337*
                       1381*  1425*  1469*  1513*  1557*  1601*  1642*  1667*  1692*  1717*  1742*  1767*  1792*
                       1817*  1842*  1867*  1892*  1917*  1942*  1967*  1992*  2017*  2653*
PUSHRO=  010046         542#  2123
PUSH1S=  005746         540#
PUSH2S=  024646         544#
RBUF     012654        2610#
RESREG   011214        2117   2120#
RESTAR   013014        2639   2645#
RESTRT   010742        2052   2058#
RESOS  = 104407         849#  2120
RETURN   012160         926*   931   2074*  2076   2324#  2664
```

K05


```
RSOS    011360          2164#   2710
RWR00   012716           972     974   2612#
RWR01   012720          1016    1018   2613#
RWR010  012736          1324    1326   2620#
RWR011  012740          1368    1370   2621#
RWR012  012742          1412    1414   2622#
RWR013  012744          1456    1458   2623#
RWR014  012746          1500    1502   2624#
RWR015  012750          1544    1546   2625#
RWR016  012752          1588    1590   2626#
RWR017  012754          1632    1634   2627#
RWR02   012722          1060    1062   2614#
RWR03   012724          1104    1106   2615#
RWR04   012726          1148    1150   2616#
RWR05   012730          1192    1194   2617#
RWR06   012732          1236    1238   2618#
RWR07   012734          1280    1282   2619#
R0      =%000000         522#   2124*  2160    2164*  2271*  2275*  2282    2346*  2350*  2631    2651*
R1      =%000001         523#    879*   882*   884*   886   2051*  2054    2159   2165*  2263*  2265    2266    2267
                        2268    2632   2650*
R2      =%000002         524#    880*   882    883*   885*  2158   2166*   2633   2649*
R3      =%000003         525#    881*   884    885   2157   2167*  2191*   2201*  2272*  2273*  2274*   2275    2281*
                        2282*   2287*  2290*  2634   2648*
R4      =%000004         526#    967*   968*   971*   972   1011*  1012*   1015*  1016   1055*  1056*   1059*   1060
                        1099*   1100*  1103*  1104   1143*  1144*  1147*   1148   1187*  1188*  1191*   1192    1?:1*
                        1232*   1235*  1236   1275*  1276*  1279*  1280    1319*  1320*  1323*  1324    1363*   1364*
                        1367*   1368   1407*  1408*  1411*  1412   1451*   1452*  1455*  1456   1495*   1496*   1499*
                        1500    1539*  1540*  1543*  1544   1583*  1584*   1587*  1588   1627*  1628*   1631*   1632
                        2100*   2101*  2102*  2103*  2104*  2105   2106    2156   2168*  2190*  2194*   2195*   2196
                        2218*   2219   2221   2223   2225*  2226   2227    2247*  2248*  2269*  2272    2276*   2277*
                        2278*   2635   2647*
RS      =%000005         527#    969*   974*  1013*  1018*  1057*  1062*   1101*  1106*  1145*  1150*   1189*   1194*
                        1233*   1238*  1277*  1282*  1321*  1326*  1365*   1370*  1409*  1414*  1453*   1458*   1497*
                        1502*   1541*  1546*  1585*  1590*  1629*  1634*   2098*  2099*  2100   2155    2169*   2174*
                        2178    2181   2210*  2211   2212   2213   2214    2215   2216   2217*  2226*   2229*   2230*
                        2231*   2238   2240   2242   2248   2249*  2270*   2279*  2636   2646*
SAVPC   012210          2134    2151*  2336#
SAVR0   012172          2160*   2164   2329#
SAVR1   012174          2159*   2165   2330#
SAVR2   012176          2158*   2166   2331#
SAVR3   012200          2157*   2167   2332#
SAVR4   012202          2156*   2168   2333#  2753
SAVR5   012204          2155*   2169   2334#  2751
SAVSP   012206          2335#   2638*  2645
SAVUSP= 104406           848#   2097
SCOPE  = 104400          842#    976   1020   1064   1108   1152   1196    1240   1284   1328   1372    1416    1460
                        1504    1548   1592   1636   1661   1686   1711    1736   1761   1786   1811    1836    1861
                        1886    1911   1936   1961   1986   2011   2036
SCOPER  010746          2063#   2703
SCOPE1= 104410           850#
SCOPIR  011046          2084#   2711
SP      =%000006         528#    865*   914*  2071   2076*  2086*  2093    2095   2098   2124   2130*   2140*   2141*
                        2142*   2143*  2144*  2145*  2146*  2147   2151    2174   2175*  2186   2187*   2210    2216*
                        2263    2264*  2631*  2632*  2633*  2634*  2635*   2636*  2637*  2638   2645*   2646    2647
                        2648    2649   2650   2651   2654*
SPACNT= 012105          2267*   2285   2288*  2298#
```

DZDHIB.PFC CROSS REFERENCE TABLE -- USER SYMBOLS

```
STACK = 014040            536#    865     914     2654
START   001000            834     864#
STFLG   012214            867*    927     929*    2338#
SVOS    011326           2155#
SVOSP   011320           2151#   2709
SWR   = 177570            533#    877     915     2063    2065    2067    2084    2091    2121    2128
SW00  = 000001            516#    877
SW01  = 000002            515#    915
SW02  = 000004            514#
SW03  = 000010            513#
SW04  = 000020            512#
SW05  = 000040            511#
SW06  = 000100            510#
SW08  = 000400            509#
SW09  = 001000            508#   2084
SW10  = 002000            507#   2063    2128
SW11  = 004000            506#   2067
SW12  = 010000            505#
SW13  = 020000            504#   2091
SW14  = 040000            503#   2065
SW15  = 100000            502#
TBUF    012254            958    1002    1046    1090    1134    1178    1222    1266    1310    1354    1398    1442    1486
                         1530    1574    1618    1650    1675    1700    1725    1750    1775    1800    1825    1850    1875
                         1900    1925    1950    1975    2000    2025   2353#
TDAT  = 000400           2353#   2354   2355#   2356#   2357#   2358#   2359#   2360#   2361#   2362#   2363#   2364#   2365#
                         2366#   2367#   2368#   2369#   2370#   2371#   2372#   2373#   2374#   2375#   2376#   2377#   2378#
                         2379#   2380#   2381#   2382#   2383#   2384#   2385#   2386#   2387#   2388#   2389#   2390#   2391#
                         2392#   2393#   2394#   2395#   2396#   2397#   2398#   2399#   2400#   2401#   2402#   2403#   2404#
                         2405#   2406#   2407#   2408#   2409#   2410#   2411#   2412#   2413#   2414#   2415#   2416#   2417#
                         2418#   2419#   2420#   2421#   2422#   2423#   2424#   2425#   2426#   2427#   2428#   2429#   2430#
                         2431#   2432#   2433#   2434#   2435#   2436#   2437#   2438#   2439#   2440#   2441#   2442#   2443#
                         2444#   2445#   2446#   2447#   2448#   2449#   2450#   2451#   2452#   2453#   2454#   2455#   2456#
                         2457#   2458#   2459#   2460#   2461#   2462#   2463#   2464#   2465#   2466#   2467#   2468#   2469#
                         2470#   2471#   2472#   2473#   2474#   2475#   2476#   2477#   2478#   2479#   2480#   2481#   2482#
                         2483#   2484#   2485#   2486#   2487#   2488#   2489#   2490#   2491#   2492#   2493#   2494#   2495#
                         2496#   2497#   2498#   2499#   2500#   2501#   2502#   2503#   2504#   2505#   2506#   2507#   2508#
                         2509#   2510#   2511#   2512#   2513#   2514#   2515#   2516#   2517#   2518#   2519#   2520#   2521#
                         2522#   2523#   2524#   2525#   2526#   2527#   2528#   2529#   2530#   2531#   2532#   2533#   2534#
                         2535#   2536#   2537#   2538#   2539#   2540#   2541#   2542#   2543#   2544#   2545#   2546#   2547#
                         2548#   2549#   2550#   2551#   2552#   2553#   2554#   2555#   2556#   2557#   2558#   2559#   2560#
                         2561#   2562#   2563#   2564#   2565#   2566#   2567#   2568#   2569#   2570#   2571#   2572#   2573#
                         2574#   2575#   2576#   2577#   2578#   2579#   2580#   2581#   2582#   2583#   2584#   2585#   2586#
                         2587#   2588#   2589#   2590#   2591#   2592#   2593#   2594#   2595#   2596#   2597#   2598#   2599#
                         2600#   2601#   2602#   2603#   2604#   2605#   2606#   2607#   2608#   2609#
TEMP    013426           2271    2655#   2656#   2717#
TKCSR   012110           2192    2302#
TKDBR   012112           2194    2198    2303#
TPCSR   012114           2176    2199    2304#
TPDBR   012116           2181#   2196#   2305#
TRPOK   011300           2143#
TRPSRV  011266            831    2140#
TRPTAB  013372           2145    2703#
TYPDAT  011202           2110    2116#
TYPE  = 104401            843#    872     930     2045    2114    2188    2203    2261    2291    2660
TYPER   011412           2174#   2704
TYPMSG  011164           2108    2111#
```

DZDHIB.PFC CROSS REFERENCE TABLE -- USER SYMBOLS

```
T1      001274            92#     941#
T10     003434           1249#
T11     003674           12?3#
T12     004134           1337#
T13     004374           1381#
T14     004634           1?25#
T15     005074           1?69#
T16     005334           1513#
T17     005574           1557#
T2      001534            985#
T20     006034           1601#
T21     006274           1642#
T22     006414           1667#
T23     006534           1692#
T24     006654           1717#
T25     006774           1742#
T26     007114           1767#
T27     007234           1792#
T3      001774           1029#
T30     007354           1817#
T31     007474           1842#
T32     007614           1867#
T33     007734           1892#
T34     010054           1917#
T35     010174           1942#
T36     010314           1967#
T37     010434           1992#
T4      002234           1073#
T40     010554           20?7#
T5      002474           1117#
T6      002734           1161#
T7      003174           1205#
VEC1    001060            874     877#
VEC2    001070            876     879#
WRDCNT  012102           2265*   2293*   2296#
X       = 000000            1#
XBIT    = 000000C         933#    1637#
XLINE   = 00002?          933#    1637#
XN      = 000041            1#     941     945#     985     989#    1029    1033#    1073    1077#    1117    1121#    1161    1165#
                         1205    1209#    1249    1253#    1293    1297#    1337    1341#    1381    1385#    1425    1429#    1469
                         1473#    1513    1517#    1557    1561#    1601    1605#    1642    1646#    1667    1671#    1692    1696#
                         1717    1721#    1742    1746#    1767    1771#    1792    1796#    1817    1821#    1842    1846#    1867
                         1871#    1892    1896#    1917    1921#    1942    1946#    1967    1971#    1992    1996#    2017    2021#
Y       = 000011            1#     842     843#     844#     845#     846#     847#     848#     849#     850#     851#
.       = 013642          566#     567     569     571     573     575     577     579     581     583     585     587     589
                          591     593     595     597     599     601     603     605     607     609     611     613     615
                          617     619     621     623     625     627     629     631     633     635     637     639     641
                          643     645     647     649     651     653     655     657     659     661     663     665     667
                          669     671     673     675     677     679     681     683     685     687     689     691     693
                          695     697     699     701     703     705     707     709     711     713     715     717     719
                          721     723     725     727     729     731     733     735     737     739     741     743     745
                          747     749     751     753     755     757     759     761     763     765     767     769     771
                          773     775     777     779     781     783     785     787     789     791     793     795     797
                          799     801     803     805     807     809     811     813     815     817     819     821     826#
                          833#    851#    853#    855#   2611#   2641    2657    2699#   2716#   2718#   2720#   2748#
```

```
BLIND1   932#  1637  1662  1687  1712  1737  1762  1787  1812  1837  1862  1887  1912  1937  1962
        1987  2012
BREAK1   932#   933   977  1021  1065  1109  1153  1197  1241  1285  1329  1373  1417  1461  1505
        1549  1593
CCLRAL   933#  2343
CODEM1   855#   904
HLT      546#   970   975  1014  1019  1058  1063  1102  1107  1146  1151  1190  1195  1234  1239
        1278  1283  1322  1327  1366  1371  1410  1415  1454  1459  1498  1503  1542  1547  1586
        1591  1630  1635  1660  1685  1710  1735  1760  1785  1810  1835  1860  1885  1910  1935
        1960  1985  2010  2035
SSETAL   932#
TRFDEF     1#   842   843   844   845   846   847   848   849   850
TS         1#   941   985  1029  1073  1117  1161  1205  1249  1293  1337  1381  1425  1469  1513
        1557  1601  1642  1667  1692  1717  1742  1767  1792  1817  1842  1867  1892  1917  1942
        1967  1992  2017
WORDS    2612#  2613  2614  2615  2616  2617  2618  2619  2620  2621  2622  2623  2624  2625  2626
        2627
.BUFFE     1#  2712
.CATCH     1#   565
.ENDCO     1#  2754
.EOP       1#  2078
.ERROR     1#  2188
.ERRTA     1#  2721
.HEADE     1#   480
.INSTR     1#  2183
.MSG       1#  2668
.OCTAS     1#  2258
.PARAM     1#  2207
.PFAIL     1#  2628
.POINT     1#  2300
.RESRE     1#  2162
.SAVRE     1#  2148
.SCOPE     1#  2059
.SCOP1     1#  2081
.SETVE     1#   823
.START     1#   855
.SYMBO     1#   499
.TRPDE     1#   837
.TRPSR     1#  2135
.TRPTA     1#  2700
.TYPER     1#  2171
.VARIA     1#  2319
```

```
ADD      884   885  2104  2145  2175  2187  2249  2264  2274
ASL     2101  2102  2143  2229  2230  2231
ASR     2276  2277  2278
BEQ      878   916   966   973  1010  1017  1054  1061  1098  1105  1142  1149  1186  1193  1230
        1237  1274  1281  1318  1325  1362  1369  1406  1413  1450  1457  1494  1501  1538  1545
        1582  1589  1626  1633  2052  2079  2085  2094  2108  2117  2129  2197  2220  2228  2286
BGT     2224
BHI     2239
BIC      968  1012  1056  1100  1144  1188  1232  1276  1320  1364  1408  1452  1496  1540  1584
        1628  2103  2144  2273
BICB    2195  2225
BISB    2226
BIT      877   915  2063  2065  2067  2084  2091  2128
BITB    2242
BLO     2241
BLT     2222
BNE      874   887   907   928   955   964   999  1008  1043  1052  1087  1096  1131  1140  1175
        1184  1219  1228  1263  1272  1307  1316  1351  1360  1395  1404  1439  1448  1483  1492
        1527  1536  1571  1580  1615  1624  1657  1682  1707  1732  1757  1782  1807  1832  1957
        1882  1907  1932  1957  1982  2007  2032  2064  2066  2068  2071  2092  2110  2179  2202
        2243  2251  2280  2284  2289  2294  2351  2657
BPL     1659  1684  1709  1734  1759  1784  1809  1834  1859  1884  1909  1934  1959  1984  2009
        2034  2122  2177  2193  2200
BR       876   925  2090  2182  2205  2232  2234  2641
CLR      867   868   869   870   871   883  2047  2048  2072  2073  2096  2217  2347  2348  2655
        2662  2663
CLRB    2290
CMP      886   972  1016  1060  1104  1148  1192  1236  1280  1324  1368  1412  1456  1500  1544
        1588  1632  2070  2093  2238  2240
CMPB     954   965   998  1009  1042  1053  1086  1097  1130  1141  1174  1185  1218  1229  1262
        1273  1306  1317  1350  1361  1394  1405  1438  1449  1482  1493  1526  1537  1570  1581
        1614  1625  2196  2219  2221  2223  2227
COM      908   929
DEC     2201  2279  2293  2350
DECB    2250  2283  2288
EMT      546
HALT     568   570   572   574   576   578   580   582   584   586   588   590   592   594   596
        598   600   602   604   606   608   610   612   614   616   618   620   622   624   626
        628   630   632   634   636   638   640   642   644   646   648   650   652   654   656
        658   660   662   664   666   668   670   672   674   676   678   680   682   684   686
        688   690   692   694   696   698   700   702   704   706   708   710   712   714   716
        718   720   722   724   726   729   730   732   734   736   738   740   742   744   746
        748   750   752   754   756   759   760   762   764   766   768   770   772   774   776
        778   780   782   784   786   788   790   792   794   796   798   800   802   804   806
        808   810   812   814   816   818   820   822  2125  2640
INC      905  2049  2069  2127  2349  2656
JMP      834   951  2058  2147  2564
JSR      946   990  1034  1078  1122  1166  1210  1254  1298  1342  1386  1430  1474  1518  1562
        1606  1647  1672  1697  1722  1747  1772  1797  1822  1847  1872  1897  1922  1947  1972
        1997  2022  2054
MOV      864   865   866   879   880   881   882   904   913   914   926   941   942   943   944
        945   948   949   950   951   953   956   957   958   959   960   961   962   967   969
        971   974   985   986   987   988   989   992   993   994   995   997  1000  1001  1002
       1003  1004  1005  1006  1011  1013  1...  1018  1029  1030  1031  1032  1033  1036  1037
       1038  1039  1041  1044  1045  1046  1047  1048  1049  1050  1055  1057  1059  1062  1073
       1074  1075  1076  1077  1080  1081  1082  1083  1085  1088  1089  1090  1091  1092  1093
```

```
                1094  1099  1101  1103  1106  1117  1118  1119  1120  1121  1124  1125  1126  1127  1129
                1132  1133  1134  1135  1136  1137  1138  1143  1145  1147  1153  1161  1162  1163  1164
                1165  1168  1169  1170  1171  1173  1176  1177  1178  1179  1180  1181  1182  1187  1189
                1191  1194  1205  1206  1207  1208  1209  1212  1213  1214  1215  1217  1220  1221  1222
                1223  1224  1225  1226  1231  1233  1235  1238  1249  1250  1251  1252  1253  1256  1257
                1258  1259  1261  1264  1265  1266  1267  1268  1269  1270  1275  1277  1279  1282  1293
                1294  1295  1296  1297  1300  1301  1302  1303  1305  1308  1309  1310  1311  1312  1313
                1314  1319  1321  1323  1326  1337  1338  1339  1340  1341  1344  1345  1346  1347  1349
                1352  1353  1354  1355  1356  1357  1358  1363  1365  1367  1370  1381  1382  1383  1384
                1385  1388  1389  1390  1391  1393  1396  1397  1398  1399  1400  1401  1402  1407  1409
                1411  1414  1425  1426  1427  1428  1429  1432  1433  1434  1435  1437  1440  1441  1442
                1443  1444  1445  1446  1451  1453  1455  1458  1469  1470  1471  1472  1473  1476  1477
                1478  1479  1481  1484  1485  1486  1487  1488  1489  1490  1495  1497  1499  1502  1513
                1514  1515  1516  1517  15[illegible]  1521  1522  1523  1525  1528  1529  1530  1531  1532  1533
                1534  1539  1541  1543  1546  1557  1558  1559  1560  1561  1564  1565  1566  1567  1569
                1572  1573  1574  1575  1576  1577  1578  1583  1585  1587  1590  1601  1602  1603  1604
                1605  1608  1609  1610  1611  1613  1616  1617  1618  1619  1620  1621  1622  1627  1629
                1631  1634  1642  1643  1644  1645  1646  1649  1650  1651  1652  1655  1667  1668  1669
                1670  1671  1674  1675  1676  1677  1680  1692  1693  1694  1695  1696  1699  1700  1701
                1702  1705  1717  1718  1719  1720  1721  1724  1725  1726  1727  1730  1742  1743  1744
                1745  1746  1749  1750  1751  1752  1755  1757  1768  1769  1770  1771  1774  1775  1776
                1777  1780  1792  1793  1794  1795  1796  1799  1800  1801  1802  1805  1817  1818  1819
                1820  1821  1824  1825  1826  1827  1830  1842  1843  1844  1845  1846  1849  1850  1851
                1852  1855  1867  1868  1869  1870  1871  1874  1875  1876  1877  1880  1892  1893  1894
                1995  1896  1899  1900  1901  1902  1905  1917  1918  19[illegible]9  1920  1921  1924  1925  1926
                1927  1930  1942  1943  1944  1945  1946  1949  1950  1951  1952  1955  1967  1968  1969
                1970  1971  1974  1975  1976  1977  1980  1992  1993  1994  1995  1996  1999  2000  2001
                2002  2005  2017  2018  2019  2020  2021  2024  2025  2026  2027  2030  2050  2051  2074
                2076  2086  2095  2098  2100  2105  2106  2113  2124  2[illegible]30  2140  2142  2146  2151  2155
                2156  2157  2158  2159  2160  2164  2165  2166  2167  2168  2169  2174  2186  2190  2191
                2210  2211  2212  2213  2216  2218  2247  2248  2263  2265  2268  2269  2271  2272  2281
                2346  2631  2632  2633  2634  2635  2636  2637  2638  2639  2645  2646  2647  2648  2649
                2650  2651  2652  2653  2654
MOVB            2181  2194  2198  2214  2215  2266  2267  2270  2275  2282  2287
NOP             2055  2056  2057
RESET           2053
RETURN           922  2667
RTI             2075  2077  2087  2131  2161  2170  2180  2206  2252  2295
RTS             23[illegible]2
SUB             2099  2141
TRAP             842   843   844   845   846   847   848   849   850
TST              873   906   927   963  1007  1051  1095  1139  1183  1227  1271  1315  1359  1403  1447
                1491  1535  1579  1623  1656  1681  1706  1731  1756  1781  1806  1831  1856  1881  1906
                1931  1956  1981  2006  2031  2078  2107  2109  2116  2121
TSTB            1658  1683  1708  1733  1758  1783  1808  1833  1858  1883  1908  1933  1958  1983  2008
                2033  2176  2178  2192  2199  2285
.ASCIZ          2668  2675  2678  2683  2684  2685  2694  2696  2697  2731  2739  2744
.BYTE            894   895   902   903   923   924  2133  2353  2354  2355  2356  2357  2358  2359  2360
                2361  2362  2363  2364  2365  2366  2367  2368  2369  2370  2371  2372  2373  2374  2375
                2376  2377  2378  2379  2380  2381  2382  2383  2384  2385  2386  2387  23[illegible]8  2389  2390
                2391  2392  2393  2394  2395  2396  2397  2398  2399  2400  2401  2402  2403  2404  2405
                2406  2407  2408  2409  2410  2411  2412  2413  2414  2415  2416  2417  2418  2419  2420
                2421  2422  2423  2424  2425  2426  2427  2428  2429  2430  2431  2432  2433  2434  2435
                2436  2437  2438  2439  2440  2441  2442  2443  2444  2445  2446  2447  2448  2449  2450
                2451  2452  2453  2454  2455  2456  2457  2458  2459  2460  2461  2462  2463  2464  2465
                2466  2467  2468  2469  2470  2471  2472  2473  2474  2475  2476  2477  2478  2479  2480
```

D06


```
                2481  2482  2483  2484  2485  2486  2487  2488  2489  2490  2491  2492  2493  2494  2495
                2496  2497  2498  2499  2500  2501  2502  2503  2504  2505  2506  2507  2508  2509  2510
                2511  2512  2513  2514  2515  2516  2517  2518  2519  2520  2521  2522  2523  2524  2525
                2536  2527  2528  2529  2530  2531  2532  2533  2534  2535  2536  2537  2538  2539  2540
                2541  2542  2543  2544  2545  2546  2547  2548  2549  2550  2551  2552  2553  2554  2555
                2556  2557  2558  2559  2560  2561  2562  2563  2564  2565  2566  2567  2568  2569  2570
                2571  2572  2573  2574  2575  2576  2577  2578  2579  2580  2581  2582  2583  2584  2585
                2586  2587  2588  2589  2590  2591  2592  2593  2594  2595  2596  2597  2598  2599  2600
                2601  2602  2603  2604  2605  2606  2607  2608  2750  2752
.ENABL           499
.END            2755
.ENDC            876   877   904   906   945   989  1033  1077  1121  1165  1209  1253  1297  1341  1385
                1429  1473  1517  1561  1605  1646  1671  1696  1721  1746  1771  1796  1821  1846  1871
                1896  1921  1946  1971  1996  2021
.EQLIV           546
.EVEN           2609  2699  2748
.IF              874   876   904   944   988  1032  1076  1120  1164  1208  1252  1296  1340  1384  1428
                1472  1516  1560  1604  1645  1670  1695  1720  1745  1770  1795  1820  1845  1870  1895
                1920  1945  1970  1995  2020
.IFF             876   877
.IIF             863
.IRP            2306  2340
.LIST              1   480   499   843   844   845   846   847   848   849   850   851   933   945   977
                 989  1021  1033  1065  1077  1109  1121  1153  1165  1197  1209  1241  1253  1285  1297
                1329  1341  1373  1385  1417  1429  1461  1473  1505  1517  1549  1561  1593  1605  1637
                1646  1662  1671  1687  1696  1712  1721  1737  1746  1762  1771  1787  1796  1812  1821
                1837  1846  1862  1871  1887  1896  1912  1921  1937  1946  1962  1971  1987  1996  2012
                2021  2037  2353  2355  2356  2357  2358  2359  2360  2361  2362  2363  2364  2365  2366
                2367  2368  2369  2370  2371  2372  2373  2374  2375  2376  2377  2378  2379  2380  2381
                2382  2383  2384  2385  2386  2387  2388  2389  2390  2391  2392  2393  2394  2395  2396
                2397  2398  2379  2400  2401  2402  2403  2404  2405  2406  2407  2408  2409  2410  2411
                2412  2413  2414  2415  2416  2417  2418  2419  2420  2421  2422  2423  2424  2425  2426
                2427  2428  2429  2430  2431  2432  2433  2434  2435  2436  2437  2438  2439  2440  2441
                2442  2443  2444  2445  2446  2447  2448  2449  2450  2451  2452  2453  2454  2455  2456
                2457  2458  2459  2460  2461  2462  2463  2464  2465  2466  2467  2468  2469  2470  2471
                2472  2473  2474  2475  2476  2477  2478  2479  2480  2481  2482  2483  2484  2485  2486
                2487  2488  2489  2490  2491  2492  2493  2494  2495  2496  2497  2498  2499  2500  2501
                2502  2503  2504  2505  2506  2507  2508  2509  2510  2511  2512  2513  2514  2515  2516
                2517  2518  2519  2520  2521  2522  2523  2524  2525  2526  2527  2528  2529  2530  2531
                2532  2533  2534  2535  2536  2537  2538  2539  2540  2541  2542  2543  2544  2545  2546
                2547  2548  2549  2550  2551  2552  2553  2554  2555  2556  2557  2558  2559  2560  2561
                2562  2563  2564  2565  2566  2567  2568  2569  2570  2571  2572  2573  2574  2575  2576
                2577  2578  2579  2580  2581  2582  2583  2584  2585  2586  2587  2588  2589  2590  2591
                2592  2593  2594  2595  2596  2597  2598  2599  2600  2601  2602  2603  2604  2605  2606
                2607  2608  2609  2612  2613  2614  2615  2616  2617  2618  2619  2620  2621  2622  2623
                2624  2625  2626  2627  2628
.MACRO           1   855   932   933  2612
.NLIST           1   480   499   843   844   845   846   847   848   849   850   851   933   945   977
                 989  1021  1033  1065  1077  1109  1121  1153  1165  1197  1209  1241  1253  1285  1297
                1329  1341  1373  1385  1417  1429  1461  1473  1505  1517  1549  1561  1593  1605  1637
                1646  1662  1671  1687  1696  1712  1721  1737  1746  1762  1771  1787  1796  1812  1821
                1837  1846  1862  1871  1887  1896  1912  1921  1937  1946  1962  1971  1987  1996  2012
                2021  2037  2353  2355  2356  2357  2358  2359  2360  2361  2362  2363  2364  2365  2366
                2367  2368  2369  2370  2371  2372  2373  2374  2375  2376  2377  2378  2379  2380  2381
                2382  2383  2384  2385  2386  2387  2388  2389  2390  2391  2392  2393  2394  2395  2396
                2397  2398  2399  2400  2401  2402  2403  2404  2405  2406  2407  2408  2409  2410  2411
```

E06

```
DZDHI   MACY11 27(732)  22-MAR-76  10:10  PAGE 73
DZDHIB.PFC      CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

          2412    2413    2414    2415    2416    2417    2418    2419    2420    2421    2422    2423    2424    2425    2426
          2427    2428    2429    2430    2431    2432    2433    2434    2435    2436    2437    2438    2439    2440    2441
          2442    2443    2444    2445    2446    2447    2448    2449    2450    2451    2452    2453    2454    2455    2456
          2457    2458    2459    2460    2461    2462    2463    2464    2465    2466    2467    2468    2469    2470    2471
          2472    2473    2474    2475    2476    2477    2478    2479    2480    2481    2482    2483    2484    2485    2486
          2487    2488    2489    2490    2491    2492    2493    2494    2495    2496    2497    2498    2499    2500    2501
          2502    2503    2504    2505    2506    2507    2508    2509    2510    2511    2512    2513    2514    2515    2516
          2517    2518    2519    2520    2521    2522    2523    2524    2525    2526    2527    2528    2529    2530    2531
          2532    2533    2534    2535    2536    2537    2538    2539    2540    2541    2542    2543    2544    2545    2546
          2547    2548    2549    2550    2551    2552    2553    2554    2555    2556    2557    2558    2559    2560    2561
          2562    2563    2564    2565    2566    2567    2568    2569    2570    2571    2572    2573    2574    2575    2576
          2577    2578    2579    2580    2581    2582    2583    2584    2585    2586    2587    2588    2589    2590    2591
          2592    2593    2594    2595    2596    2597    2598    2599    2600    2601    2602    2603    2604    2605    2606
          2607    2608    2609    2612    2613    2614    2615    2616    2617    2618    2619    2620    2621    2622    2623
          2624    2625    2626    2627    2628
.PAGE      518     565     823     855    2088    2135    2171    2207    2258    2300    2343    2628    2668
.REM         1
.REPT      567     933    1637    2354    2612
.TITLE     499


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *DZDHIB,DZDHIB.SEQ/CRF/SOL/PAGNUM=UTIL2.P11,DZDHIB.PFC
 RUN-TIME:  12 22 4 SECONDS
 RUN-TIME RATIO:  115/40=2.8
 CORE USED:  11K  (21 PAGES)
```

F06