

301

[illegible][illegible]

REC:RU21-4:DAV:BB2-- distribution to RU21-4, slot 134

[illegible]

```
/usr/bin -c -d /var/log, /usr/bin -c -d /var/log Date: 03-Dec-76 09:50:37 Monitor IPC-D 602 (680) #START#
```

[illegible][illegible]

NYC:ML21-4:DAVIDB2-- distribution to MH21-4, slot 134

[illegible]

/usr/bin/gdb -x /dev/null /usr/sbin/sshd /usr/sbin/sshd:pcrcmd:0 Date 03-Dec-76 09:50:37 Monitor IPC-D 602 (680) #START#

The image displays three 10x10 grids illustrating the progression of a pattern. The first grid (left) features a central 'E' shape formed by 'E' characters, surrounded by 'S' characters. The second grid (middle) shows the 'E' shape evolving into a more complex, interconnected structure. The third grid (right) shows the pattern further refined, with 'O' characters appearing in the central area, suggesting a transition to a new state or a different set of rules.

```

LPTSP Version 6(100344) Running on MTA041
*START* User DAVIES TOM [400.2704] Job DZDHIC Seq. 11 Date 03-Dec-76 09:50:37 Monitor IPC-D 602 [680] *START*
Request created: 03-Dec-76 09:26:29 /TO:ML21-4:DAVIES -- distribution to ML21-4, slot 134

BUTLE B6K10B0DZDHICPR01H08990WICE05011 10091576 18992100:11821MIF:275.4F080SDM09:50:41
File will be RENAMED to (057) protection

```

The image displays three 10x10 grids illustrating the growth of a fractal pattern using the characters 'S', 'E', and 'O'. The pattern starts at the top-left corner (row 1, column 1) and expands outwards in a complex, branching manner.

- Grid 1 (S):** Shows the pattern using the character 'S'. It begins with a single 'S' at (1,1) and grows to fill a large, irregular shape, reaching the bottom-left corner by row 10.
- Grid 2 (E):** Shows the pattern using the character 'E'. It begins with a single 'E' at (1,1) and grows to fill a large, irregular shape, reaching the bottom-right corner by row 10.
- Grid 3 (O):** Shows the pattern using the character 'O'. It begins with a single 'O' at (1,1) and grows to fill a large, irregular shape, reaching the bottom-right corner by row 10.

The pattern's growth is characterized by a central vertical stem with horizontal branches that further subdivide, creating a complex, self-similar structure. The final shape in each grid is a large, irregular polygon that fills most of the 10x10 area, with the specific shape determined by the character used.

BULK BGKZB8ZDHIZPRINT4008BROW#ICE:AS011 106N1E576 10P02100:118ZIM17:275.4F080SDNORMAL09:50:44
 File will be RENAMED to '057' protection

E01

DZDHI MACY11 27.1006 29-SEP-76 14:44 PAGE 2
DZDHI P11 29-SEP-76 14:42

SEG 0001

.REM !

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDHI-C-D
PRODUCT NAME: DH11 BREAK AND HALF-DUPLEX TEST
DATE: DECEMBER 1976
MAINTAINER: DIAGNOSTIC GROUP

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1972, 1976 BY DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.

2. REQUIREMENTS

2.1 EQUIPMENT

POP-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 IS 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.

100
101
102
103
104
105
106
107
108
109
110
111
112
113
114

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 THIS 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 IMMEDIATELY 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 ENTERED 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.

223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
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. IF THE NUMBERS ARE NOT EQUAL, THE TEST CURRENTLY IN PROGRESS WILL BE REPEATED.

5.2.6 SCOP1R (FREEZE ON CURRENT DATA)

THE CALL TO THIS ROUTINE FOLLOWS IMMEDIATELY AFTER 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.

263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
5.2.7 ERRORS (ERROR HANDLER)

THIS ROUTINE IS ENTERED UPON ERROR DETECTION ONLY.
WITH ALL CONSOLE SWITCHES DOWN, THE ROUTINE PROCEEDS 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 FAILURE
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 RO. 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.

- 296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
- 5.3 PROGRAM AND OR OPERATOR ACTION
 - 5.3.1 PROGRAM START WITH ALL SWITCHES DOWN
 - 5.3.1.1 REFER TO SECTIONS 4.3.1 AND 4.3.2 FOR INITIAL PROGRAM BEHAVIOR.
 - 5.3.1.2 AFTER "R" HAS BEEN TYPED BY THE PROGRAM, TEST EXECUTION WILL BEGIN. EACH TEST WILL BE REPEATED A SELECTED NUMBER OF ITERATIONS (SEE LISTING FOR EXACT NUMBER FOR EACH TEST) AND THEN THE PROGRAM WILL PROCEED TO THE NEXT TEST.
 - 5.3.1.3 WHEN ALL ITERATIONS HAVE BEEN COMPLETED, THE PROGRAM WILL TYPE "DZDHI" AND THEN RESTART TESTING AT TEST 1 (LOCATION T1 IN THE PROGRAM).
 - 5.3.1.4 IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE, AND THEN CONTINUE THE TEST IN PROGRESS.
 - 5.3.2 PROGRAM START WITH SW00=1
THE PROGRAM WILL PERFORM AS DESCRIBED IN 4.3.1 AND 5.3.1
 - 5.3.3 PROGRAM START WITH SW01=1
 - 5.3.3.1 REFER TO SECTION 4.3.4 FOR INITIAL PROGRAM BEHAVIOR
 - 5.3.3.2 TEST EXECUTION WILL START AT THE ADDRESS SPECIFIED AND WILL CONTINUE AS DESCRIBED IN 5.3.1.2
 - 5.3.3.3 AFTER "DZDHI" HAS BEEN TYPED, THE PROGRAM WILL RESUME TESTING AT TEST 1
 - 5.3.4 PROGRAM OPERATION WITH SW15=1
SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR, THE PROGRAM WILL HALT AFTER THE ERROR TYPEOUT, AND THE PC+2 OF THE CALL TO THE ERROR ROUTINE WILL BE DISPLAYED IN RO.
 - 5.3.5 PROGRAM OPERATION WITH SW13=1
SAME AS 5.3.1 EXCEPT THAT NO ERROR TYPEOUTS WILL OCCUR
 - 5.3.6 PROGRAM OPERATION WITH SW11=1
SAME AS 5.3.1 EXCEPT THAT EACH TEST WILL BE REPEATED ONCE ONLY
 - 5.3.7 PROGRAM OPERATION WITH SW10=1
SAME AS 5.3.1, EXCEPT THAT IN THE CASE OF AN ERROR THE CURRENT TEST WILL BE ABORTED, AND THE PROGRAM WILL PROCEED TO THE NEXT TEST IN SEQUENCE.

5. (CONT'D)

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

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

6. ERRORS

6.1 ERROR HALTS

THE ERROR MESSAGE FORMAT FOR ALL ERROR TYPEOUTS
IS AS FOLLOWS

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

WHERE

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

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

6.1.1 ERROR DESCRIPTIONS

SEE LISTING FOR DETAILS OF ERRORS

6.2 ERROR RECOVERY

6.2.1 SW15=0

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

6.2.2 SW15=1

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

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.

351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405

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 ASSERTED FOR THAT LINE FOR 400 OCTAL CHARACTER TIMES. THERE IS AN INDIVIDUAL TEST LOOP FOR EACH LINE. THE TEST BEGINS BY FLUSHING 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

:000000

:040000

:020000

:010000

:004000

:002000

:001000

SW15=100000 :=1, HALT ON ERROR

SW14=40000 :=1, LOOP ON CURRENT TEST

SW13=20000 :=1, INHIBIT ERROR TYPEOUT

SW12=10000

SW11=4000 :=1, INHIBIT ITERATIONS

SW10=2000 :=1, ESCAPE TO NEXT TEST ON ERROR

SW9=1000 :=1, LOOP WITH CURRENT DATA

C02

220H1 MACY11 27.006 29-SEP-76 14:44 PAGE 13
220H1C.P11 29-SEP-76 14:42

SEG 0012

508
509
510
511
512
513
514
515
516
517
518

000400
000100
000400
000020
000010
000040
000000
000000
000000
000000
000000

SW08=400
SW06=100
SW05=40
SW04=20
SW03=0
SW02=4
SW01=2
SW00=1

: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
012540	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
02626	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

E02

22041 MAC111 27.1006 29-SEP-76 14:44 PAGE 15
 220410.P11 29-SEP-76 14:42

SEG 0014

```

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

```

F02

DDH1 MACY11 27.1006 29-SEP-76 14:44 PAGE 16
 DDHIC.P11 29-SEP-76 14:42

SEG 0015

620	000154	000156	.+2	:UNEXPECTED TRAP TO THIS LOCATION
621	000156	000000	HALT	:EXAMINE STACK TO FIND CAUSE
622	000160	000162	.+2	:UNEXPECTED TRAP TO THIS LOCATION
623	000162	000000	HALT	:EXAMINE STACK TO FIND CAUSE
624	000164	000166	.+2	:UNEXPECTED TRAP TO THIS LOCATION
625	000166	000000	HALT	:EXAMINE STACK TO FIND CAUSE
626	000170	000172	.+2	:UNEXPECTED TRAP TO THIS LOCATION
627	000172	000000	HALT	:EXAMINE STACK TO FIND CAUSE
628	000174	000176	.+2	:UNEXPECTED TRAP TO THIS LOCATION
629	000176	000000	HALT	:EXAMINE STACK TO FIND CAUSE
630	000200	000202	.+2	:UNEXPECTED TRAP TO THIS LOCATION
631	000202	000000	HALT	:EXAMINE STACK TO FIND CAUSE
632	000204	000206	.+2	:UNEXPECTED TRAP TO THIS LOCATION
633	000206	000000	HALT	:EXAMINE STACK TO FIND CAUSE
634	000210	000212	.+2	:UNEXPECTED TRAP TO THIS LOCATION
635	000212	000000	HALT	:EXAMINE STACK TO FIND CAUSE
636	000214	000216	.+2	:UNEXPECTED TRAP TO THIS LOCATION
637	000216	000000	HALT	:EXAMINE STACK TO FIND CAUSE
638	000220	000222	.+2	:UNEXPECTED TRAP TO THIS LOCATION
639	000222	000000	HALT	:EXAMINE STACK TO FIND CAUSE
640	000224	000226	.+2	:UNEXPECTED TRAP TO THIS LOCATION
641	000226	000000	HALT	:EXAMINE STACK TO FIND CAUSE
642	000230	000232	.+2	:UNEXPECTED TRAP TO THIS LOCATION
643	000232	000000	HALT	:EXAMINE STACK TO FIND CAUSE
644	000234	000236	.+2	:UNEXPECTED TRAP TO THIS LOCATION
645	000236	000000	HALT	:EXAMINE STACK TO FIND CAUSE
646	000240	000242	.+2	:UNEXPECTED TRAP TO THIS LOCATION
647	000242	000000	HALT	:EXAMINE STACK TO FIND CAUSE
648	000244	000246	.+2	:UNEXPECTED TRAP TO THIS LOCATION
649	000246	000000	HALT	:EXAMINE STACK TO FIND CAUSE
650	000250	000252	.+2	:UNEXPECTED TRAP TO THIS LOCATION
651	000252	000000	HALT	:EXAMINE STACK TO FIND CAUSE
652	000254	000256	.+2	:UNEXPECTED TRAP TO THIS LOCATION
653	000256	000000	HALT	:EXAMINE STACK TO FIND CAUSE
654	000260	000262	.+2	:UNEXPECTED TRAP TO THIS LOCATION
655	000262	000000	HALT	:EXAMINE STACK TO FIND CAUSE
656	000264	000266	.+2	:UNEXPECTED TRAP TO THIS LOCATION
657	000266	000000	HALT	:EXAMINE STACK TO FIND CAUSE
658	000270	000272	.+2	:UNEXPECTED TRAP TO THIS LOCATION
659	000272	000000	HALT	:EXAMINE STACK TO FIND CAUSE
660	000274	000276	.+2	:UNEXPECTED TRAP TO THIS LOCATION
661	000276	000000	HALT	:EXAMINE STACK TO FIND CAUSE
662	000300	000302	.+2	:UNEXPECTED TRAP TO THIS LOCATION
663	000302	000000	HALT	:EXAMINE STACK TO FIND CAUSE
664	000304	000306	.+2	:UNEXPECTED TRAP TO THIS LOCATION
665	000306	000000	HALT	:EXAMINE STACK TO FIND CAUSE
666	000310	000312	.+2	:UNEXPECTED TRAP TO THIS LOCATION
667	000312	000000	HALT	:EXAMINE STACK TO FIND CAUSE
668	000314	000316	.+2	:UNEXPECTED TRAP TO THIS LOCATION
669	000316	000000	HALT	:EXAMINE STACK TO FIND CAUSE
670	000320	000322	.+2	:UNEXPECTED TRAP TO THIS LOCATION
671	000322	000000	HALT	:EXAMINE STACK TO FIND CAUSE
672	000324	000326	.+2	:UNEXPECTED TRAP TO THIS LOCATION
673	000326	000000	HALT	:EXAMINE STACK TO FIND CAUSE
674	000330	000332	.+2	:UNEXPECTED TRAP TO THIS LOCATION
675	000332	000000	HALT	:EXAMINE STACK TO FIND CAUSE

G02

DDMI MACY11 27(1006) 29-SEP-76 14:44 PAGE 17
 DDMIC.P11 29-SEP-76 14:42

SEG 0016

676	000334	000336	.+2	:UNEXPECTED TRAP TO THIS LOCATION
677	000336	000000	HALT	:EXAMINE STACK TO FIND CAUSE
678	000340	000342	.+2	:UNEXPECTED TRAP TO THIS LOCATION
679	000342	000000	HALT	:EXAMINE STACK TO FIND CAUSE
680	000344	000346	.+2	:UNEXPECTED TRAP TO THIS LOCATION
681	000346	000000	HALT	:EXAMINE STACK TO FIND CAUSE
682	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
683	000352	000000	HALT	:EXAMINE STACK TO FIND CAUSE
684	000354	000356	.+2	:UNEXPECTED TRAP TO THIS LOCATION
685	000356	000000	HALT	:EXAMINE STACK TO FIND CAUSE
686	000360	000362	.+2	:UNEXPECTED TRAP TO THIS LOCATION
687	000362	000000	HALT	:EXAMINE STACK TO FIND CAUSE
688	000364	000366	.+2	:UNEXPECTED TRAP TO THIS LOCATION
689	000366	000000	HALT	:EXAMINE STACK TO FIND CAUSE
690	000370	000372	.+2	:UNEXPECTED TRAP TO THIS LOCATION
691	000372	000000	HALT	:EXAMINE STACK TO FIND CAUSE
692	000374	000376	.+2	:UNEXPECTED TRAP TO THIS LOCATION
693	000376	000000	HALT	:EXAMINE STACK TO FIND CAUSE
694	000400	000402	.+2	:UNEXPECTED TRAP TO THIS LOCATION
695	000402	000000	HALT	:EXAMINE STACK TO FIND CAUSE
696	000404	000406	.+2	:UNEXPECTED TRAP TO THIS LOCATION
697	000406	000000	HALT	:EXAMINE STACK TO FIND CAUSE
698	000410	000412	.+2	:UNEXPECTED TRAP TO THIS LOCATION
699	000412	000000	HALT	:EXAMINE STACK TO FIND CAUSE
700	000414	000416	.+2	:UNEXPECTED TRAP TO THIS LOCATION
701	000416	000000	HALT	:EXAMINE STACK TO FIND CAUSE
702	000420	000422	.+2	:UNEXPECTED TRAP TO THIS LOCATION
703	000422	000000	HALT	:EXAMINE STACK TO FIND CAUSE
704	000424	000426	.+2	:UNEXPECTED TRAP TO THIS LOCATION
705	000426	000000	HALT	:EXAMINE STACK TO FIND CAUSE
706	000430	000432	.+2	:UNEXPECTED TRAP TO THIS LOCATION
707	000432	000000	HALT	:EXAMINE STACK TO FIND CAUSE
708	000434	000436	.+2	:UNEXPECTED TRAP TO THIS LOCATION
709	000436	000000	HALT	:EXAMINE STACK TO FIND CAUSE
710	000440	000442	.+2	:UNEXPECTED TRAP TO THIS LOCATION
711	000442	000000	HALT	:EXAMINE STACK TO FIND CAUSE
712	000444	000446	.+2	:UNEXPECTED TRAP TO THIS LOCATION
713	000446	000000	HALT	:EXAMINE STACK TO FIND CAUSE
714	000450	000452	.+2	:UNEXPECTED TRAP TO THIS LOCATION
715	000452	000000	HALT	:EXAMINE STACK TO FIND CAUSE
716	000454	000456	.+2	:UNEXPECTED TRAP TO THIS LOCATION
717	000456	000000	HALT	:EXAMINE STACK TO FIND CAUSE
718	000460	000462	.+2	:UNEXPECTED TRAP TO THIS LOCATION
719	000462	000000	HALT	:EXAMINE STACK TO FIND CAUSE
720	000464	000466	.+2	:UNEXPECTED TRAP TO THIS LOCATION
721	000466	000000	HALT	:EXAMINE STACK TO FIND CAUSE
722	000470	000472	.+2	:UNEXPECTED TRAP TO THIS LOCATION
723	000472	000000	HALT	:EXAMINE STACK TO FIND CAUSE
724	000474	000476	.+2	:UNEXPECTED TRAP TO THIS LOCATION
725	000476	000000	HALT	:EXAMINE STACK TO FIND CAUSE
726	000500	000502	.+2	:UNEXPECTED TRAP TO THIS LOCATION
727	000502	000000	HALT	:EXAMINE STACK TO FIND CAUSE
728	000504	000506	.+2	:UNEXPECTED TRAP TO THIS LOCATION
729	000506	000000	HALT	:EXAMINE STACK TO FIND CAUSE
730	000510	000512	.+2	:UNEXPECTED TRAP TO THIS LOCATION
731	000512	000000	HALT	:EXAMINE STACK TO FIND CAUSE

H02

02DHI MACY11 27:006: 29-SEP-76 14:44 PAGE 18
 02DHI.C.F11 29-SEP-76 14:42

SEQ 0017

732	000514	000516	.+2	: UNEXPECTED TRAP TO THIS LOCATION
733	000516	000000	HALT	: EXAMINE STACK TO FIND CAUSE
734	000520	000522	.+2	: UNEXPECTED TRAP TO THIS LOCATION
735	000522	000000	HALT	: EXAMINE STACK TO FIND CAUSE
736	000524	000526	.+2	: UNEXPECTED TRAP TO THIS LOCATION
737	000526	000000	HALT	: EXAMINE STACK TO FIND CAUSE
738	000530	000532	.+2	: UNEXPECTED TRAP TO THIS LOCATION
739	000532	000000	HALT	: EXAMINE STACK TO FIND CAUSE
740	000534	000536	.+2	: UNEXPECTED TRAP TO THIS LOCATION
741	000536	000000	HALT	: EXAMINE STACK TO FIND CAUSE
742	000540	000542	.+2	: UNEXPECTED TRAP TO THIS LOCATION
743	000542	000000	HALT	: EXAMINE STACK TO FIND CAUSE
744	000544	000546	.+2	: UNEXPECTED TRAP TO THIS LOCATION
745	000546	000000	HALT	: EXAMINE STACK TO FIND CAUSE
746	000550	000552	.+2	: UNEXPECTED TRAP TO THIS LOCATION
747	000552	000000	HALT	: EXAMINE STACK TO FIND CAUSE
748	000554	000556	.+2	: UNEXPECTED TRAP TO THIS LOCATION
749	000556	000000	HALT	: EXAMINE STACK TO FIND CAUSE
750	000560	000562	.+2	: UNEXPECTED TRAP TO THIS LOCATION
751	000562	000000	HALT	: EXAMINE STACK TO FIND CAUSE
752	000564	000566	.+2	: UNEXPECTED TRAP TO THIS LOCATION
753	000566	000000	HALT	: EXAMINE STACK TO FIND CAUSE
754	000570	000572	.+2	: UNEXPECTED TRAP TO THIS LOCATION
755	000572	000000	HALT	: EXAMINE STACK TO FIND CAUSE
756	000574	000576	.+2	: UNEXPECTED TRAP TO THIS LOCATION
757	000576	000000	HALT	: EXAMINE STACK TO FIND CAUSE
758	000600	000602	.+2	: UNEXPECTED TRAP TO THIS LOCATION
759	000602	000000	HALT	: EXAMINE STACK TO FIND CAUSE
760	000604	000606	.+2	: UNEXPECTED TRAP TO THIS LOCATION
761	000606	000000	HALT	: EXAMINE STACK TO FIND CAUSE
762	000610	000612	.+2	: UNEXPECTED TRAP TO THIS LOCATION
763	000612	000000	HALT	: EXAMINE STACK TO FIND CAUSE
764	000614	000616	.+2	: UNEXPECTED TRAP TO THIS LOCATION
765	000616	000000	HALT	: EXAMINE STACK TO FIND CAUSE
766	000620	000622	.+2	: UNEXPECTED TRAP TO THIS LOCATION
767	000622	000000	HALT	: EXAMINE STACK TO FIND CAUSE
768	000624	000626	.+2	: UNEXPECTED TRAP TO THIS LOCATION
769	000626	000000	HALT	: EXAMINE STACK TO FIND CAUSE
770	000630	000632	.+2	: UNEXPECTED TRAP TO THIS LOCATION
771	000632	000000	HALT	: EXAMINE STACK TO FIND CAUSE
772	000634	000636	.+2	: UNEXPECTED TRAP TO THIS LOCATION
773	000636	000000	HALT	: EXAMINE STACK TO FIND CAUSE
774	000640	000642	.+2	: UNEXPECTED TRAP TO THIS LOCATION
775	000642	000000	HALT	: EXAMINE STACK TO FIND CAUSE
776	000644	000646	.+2	: UNEXPECTED TRAP TO THIS LOCATION
777	000646	000000	HALT	: EXAMINE STACK TO FIND CAUSE
778	000650	000652	.+2	: UNEXPECTED TRAP TO THIS LOCATION
779	000652	000000	HALT	: EXAMINE STACK TO FIND CAUSE
780	000654	000656	.+2	: UNEXPECTED TRAP TO THIS LOCATION
781	000656	000000	HALT	: EXAMINE STACK TO FIND CAUSE
782	000660	000662	.+2	: UNEXPECTED TRAP TO THIS LOCATION
783	000662	000000	HALT	: EXAMINE STACK TO FIND CAUSE
784	000664	000666	.+2	: UNEXPECTED TRAP TO THIS LOCATION
785	000666	000000	HALT	: EXAMINE STACK TO FIND CAUSE
786	000670	000672	.+2	: UNEXPECTED TRAP TO THIS LOCATION
787	000672	000000	HALT	: EXAMINE STACK TO FIND CAUSE

DDH1 MACY11 27(1006) 29-SEP-76 14:44 PAGE 19
 DDH1C.P11 29-SEP-76 14:42

SEG 0018

788	000674	000676	.+2	: UNEXPECTED TRAP TO THIS LOCATION
789	000676	000000	HALT	: EXAMINE STACK TO FIND CAUSE
790	000700	000702	.+2	: UNEXPECTED TRAP TO THIS LOCATION
791	000702	000000	HALT	: EXAMINE STACK TO FIND CAUSE
792	000704	000706	.+2	: UNEXPECTED TRAP TO THIS LOCATION
793	000706	000000	HALT	: EXAMINE STACK TO FIND CAUSE
794	000710	000712	.+2	: UNEXPECTED TRAP TO THIS LOCATION
795	000712	000000	HALT	: EXAMINE STACK TO FIND CAUSE
796	000714	000716	.+2	: UNEXPECTED TRAP TO THIS LOCATION
797	000716	000000	HALT	: EXAMINE STACK TO FIND CAUSE
798	000720	000722	.+2	: UNEXPECTED TRAP TO THIS LOCATION
799	000722	000000	HALT	: EXAMINE STACK TO FIND CAUSE
800	000724	000726	.+2	: UNEXPECTED TRAP TO THIS LOCATION
801	000726	000000	HALT	: EXAMINE STACK TO FIND CAUSE
802	000730	000732	.+2	: UNEXPECTED TRAP TO THIS LOCATION
803	000732	000000	HALT	: EXAMINE STACK TO FIND CAUSE
804	000734	000736	.+2	: UNEXPECTED TRAP TO THIS LOCATION
805	000736	000000	HALT	: EXAMINE STACK TO FIND CAUSE
806	000740	000742	.+2	: UNEXPECTED TRAP TO THIS LOCATION
807	000742	000000	HALT	: EXAMINE STACK TO FIND CAUSE
808	000744	000746	.+2	: UNEXPECTED TRAP TO THIS LOCATION
809	000746	000000	HALT	: EXAMINE STACK TO FIND CAUSE
810	000750	000752	.+2	: UNEXPECTED TRAP TO THIS LOCATION
811	000752	000000	HALT	: EXAMINE STACK TO FIND CAUSE
812	000754	000756	.+2	: UNEXPECTED TRAP TO THIS LOCATION
813	000756	000000	HALT	: EXAMINE STACK TO FIND CAUSE
814	000760	000762	.+2	: UNEXPECTED TRAP TO THIS LOCATION
815	000762	000000	HALT	: EXAMINE STACK TO FIND CAUSE
816	000764	000766	.+2	: UNEXPECTED TRAP TO THIS LOCATION
817	000766	000000	HALT	: EXAMINE STACK TO FIND CAUSE
818	000770	000772	.+2	: UNEXPECTED TRAP TO THIS LOCATION
819	000772	000000	HALT	: EXAMINE STACK TO FIND CAUSE
820	000774	000776	.+2	: UNEXPECTED TRAP TO THIS LOCATION
821	000776	000000	HALT	: EXAMINE STACK TO FIND CAUSE

K02

DZDHI MACY11 27.1006: 29-SEP-76 14:44 PAGE 21
 DZDHI0.F11 29-SEP-76 14:42

SEQ 0020

```

854          001000          . = 1000
855
856          :PROGRAM INITIALIZATION
857          :LOCK OUT INTERRUPTS
858          :SET UP PROCESSOR STACK
859          :SET UP POWER FAIL VECTOR
860          :CLEAR PROGRAM FLAGS AND COUNTS
861          :TYPE TITLE MESSAGE
862
863 001000 012767 000340 176770 START: MOV      #340,PS          :LOCK OUT INTERRUPTS
864 001006 012706 013540          MOV      #STACK,SP          :SET UP PROCESSOR STACK
865 001012 012737 012456 000024 MOV      #PFAIL,2#24          :SET UP POWER FAIL TRAP
866 001020 005067 010670          CLR      STFLG              :CLEAR TEST START FLAG
867 001024 005067 010624          CLR      PASCNT             :CLEAR PASS COUNT
868 001030 005067 010622          CLR      ERRCNT             :CLEAR ERROR COUNT
869 001034 005067 010612          CLR      ERRFLG             :CLEAR ERROR FLAG
870 001040 005067 010606          CLR      ERRFLG             :CLEAR LAST ERROR PC
871 001044 104401 012622          TYPE     ,MTITLE            :TYPE TITLE MESSAGE
872 001050 005767 010636          TST      INIFLG              :CHECK INITIALIZATION FLAG
873 001054 001001          BNE      VEC1                        :IF NOT 0, CHECK SWITCHES
874                                     :FOR REINITIALIZATION
875 001056 000404          BR      VEC2
876 001060 032767 000001 176502 VEC1: BIT      #SW00,SWR          :IF SW00=1, GET NEW VECTOR
877 001066 001445          BEQ      BEGIN                      :AND CSR
878 001070 012701 000300          VEC2: MOV      #300,R1
879 001074 012702 000302          MOV      #302,R2
880 001100 012703 000004          MOV      #4,R3
881 001104 010211          IS:    MOV      R2,(R1)              :RESTORE TRAPCATCHER
882 001106 005012          CLR      (R2)                        :IN FLOATING VECTOR AREA
883 001110 060301          ADD      R3,R1
884 001112 060302          ADD      R3,R2
885 001114 020127 001000          CMP      R1,#1000
886 001120 001371          BNE      IS
887 001122 104403          INSTR                                :INPUT ADDRESS OF DEVICE VECTOR
888 001124 012670          MVECTOR                              :MESSAGE "VECTOR ADDRESS--"
889 001126 104405          PARAM                                :CONVERT STRING TO OCTAL
890 001130 000300          300                                  :LOW LIMIT
891 001132 000770          770                                  :HIGH LIMIT
892 001134 011642          DHRVEC                                :LOCATIONS TO BE FILLED
893 001136 003          .BYTE 3                                :NUMBER OF LOCATIONS
894 001137 004          .BYTE 4                                :LSB MASK
895 001140 104403          INSTR                                :INPUT ADDRESS OF DEVICE CSR
896 001142 012712          MREGAD                              :MESSAGE "CONTROL REGISTER ADDRESS--"
897 001144 104405          PARAM                                :CONVERT STRING TO OCTAL
898 001146 000000          0                                  :LOW LIMIT
899 001150 177776          177776                              :HIGH LIMIT
900 001152 011620          DHSCR                                :LOCATIONS TO BE FILLED
901 001154 007          .BYTE 7                                :NUMBER OF LOCATIONS
902 001155 010          .BYTE 10                               :LSB MASK
903 001156 016767 010454 010454 MOV      DHSSR,DHSLR          :SET UP ADDRESS OF SILO
904 001164 005267 010450          INC      DHSLR              :STATUS REGISTER HIGH BYTE
905 001170 005767 010516          TST      INIFLG              :IF INITIALIZATION FLAG
906 001174 001002          BNE      BEGIN                      :IS CLEARED
907 001176 005167 010510          COM      INIFLG              :SET IT
908
909          :PROGRAM START

```

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 22
 DZDHI.C.P11 29-SEP-76 14:42

SEQ 0021

```

910                                     ;CHECK FOR PROGRAM START AT SELECTED ADDRESS
911
912 001202 012767 000340 176566 BEGIN: MOV    #340,PS          ;LOCK OUT INTERRUPTS
913 001210 012706 013540          MOV    #STACK,SP          ;SET UP PROCESSOR STACK
914 001214 032767 000002 176346 BIT     #SW01,SWR          ;IF SW01=1
915 001222 001410          BEQ     1$                      ;GET PC FOR PROGRAM START
916 001224 104403          INSTR                          ;GET PC
917 001226 013056          MTSTPC                          ;MESSAGE "TEST PC"
918 001230 104405          PARAM                          ;CONVERT STRING TO OCTAL
919 001232 000000          0
920 001234 017500          17500
921 001236 000207          RETURN
922 001240 001          .BYTE 1
923 001241 001          .BYTE 1
924 001242 000410          BR      2$
925 001244 012767 001274 010406 1$: MOV    #T1,RETURN          ;NORMAL START, TEST 1
926 001252 005767 010436          TST    STFLG              ;IF LOOPING, BYPASS TYPEOUT
927 001256 001004          BNE     3$
928 001260 005167 010430          COM     STFLG
929 001264 104401 013052          2$: TYPE  MR
930 001270 000177 010354          3$: JMP     0RETURN          ;TYPE "R" TO INDICATE START
                                     ;START TESTING

```

M02

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 23
DZDHIC.P11 29-SEP-76 14:42

SEQ 0022

931

N02

DZDHI MACY:11 27(1006) 29-SEP-76 14:44 PAGE 24
 DZDHI.C.P11 29-SEP-76 14:42

SEQ 0023

```

932
933      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
934      ;ON LINE 0
935      ;SET BREAK BIT FOR LINE 0
936      ;TRANSMIT BINARY COUNT PATTERN ON LINE 0
937      ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
938      ;AND THAT IT IS A BREAK
939
940 001274 012767 000340 176474 11: MOV    #340,PS      ;DISABLE ALL INTERRUPTS
941 001302 012767 000020 010356      MOV    #20,ICOUNT  ;SET UP FOR 20 ITERATIONS
942 001310 012767 001524 010344      MOV    #4$,ESCAPE  ;SET UP TO ESCAPE TO NEXT TEST
943 001316 012777 004000 010274      MOV    #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
944 001324 004767 010376      JSR     PC,CLRALL  ;CLEAR ALL BUS ADDRESS AND
945                                     ;BLS ADDRESS MEMORY LOCATIONS
946 001330 012777 000000 010262      MOV    #0,ADHSCR   ;SELECT LINE 0
947 001336 012777 011722 010262      MOV    #NULL,ADHBA  ;SET UP TO TRANSMIT 0 CHARACTER
948 001344 012777 177776 010256      MOV    #-2,ADHBC    ;TWO 0S WILL BE TRANSMITTED
949 001352 012777 033503 010244      MOV    #33503,ADHLPR ;SET LINE SPEED=9600 BAUD
950                                     ;CHARACTER LENGTH =8 BITS
951 001360 012777 000001 010244      MOV    #1,ADHBA     ;SET BAR BIT FOR LINE 0
952 001366 122777 000002 010244 15: CMPB   #2,ADHSLR    ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
953 001374 001374      BNE     1$
954 001376 012777 004000 010214      MOV    #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
955 001404 012777 000000 010206      MOV    #0,ADHSCR   ;SELECT LINE 0
956 001412 012777 011754 010206      MOV    #TBUF,ADHBA  ;SET UP TO TRANSMIT 400
957 001420 012777 177400 010202      MOV    #-400,ADHBC  ;(OCTAL) CHARACTERS
958 001426 012777 033503 010170      MOV    #33503,ADHLPR ;LINE SPEED = 9600 BAUD
959 001434 012777 000001 010172      MOV    #1,ADHBC     ;SET BREAK BIT FOR LINE 0
960 001442 012777 000001 010162      MOV    #1,ADHBA     ;SET BAR BIT FOR LINE 0
961 001450 005777 010156 25: TST     ADHBA      ;WAIT FOR ALL CHARACTERS
962 001454 001375      BNE     2$          ;TO BE TRANSMITTED
963 001456 122777 000001 010154      CMPB   #1,ADHSLR    ;CHECK TO SEE THAT ONLY
964 001464 001407      BEQ     3$          ;1 CHARACTER WAS RECEIVED
965 001466 017704 010144      MOV    ADHSSR,R4    ;(R4)=ACTUAL RECEIVED DATA
966 001472 042704 000300      BIC     #300,R4     ;CLEAR UNWANTED BITS
967 001476 012705 000400      MOV    #400,R5     ;(R5)=EXPECTED SILO FILL LEVEL, 1
968 001502 104000      HLT     0              ;MORE THAN ONE CHARACTER RECEIVED, ERROR
969 001504 017704 010112 35: MOV     ADHNR0,R4    ;READ NEXT RECEIVED CHARACTER REGISTER
970 001510 026704 010702      CMP     RWRD0,R4    ;IS RECEIVED CHARACTER A BREAK
971 001514 001403      BEQ     4$
972 001516 016705 010674      MOV     RWRD0,R5    ;(R5)=EXPECTED RECEIVED CHARACTER
973 001522 104001      HLT     1              ;RECEIVED DATA ERROR
974 001524 104400 45: SCOPE      ;CHECK FOR ITERATIONS. LOOP
975
976      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
977      ;ON LINE 1
978      ;SET BREAK BIT FOR LINE 1
979      ;TRANSMIT BINARY COUNT PATTERN ON LINE 1
980      ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
981      ;AND THAT IT IS A BREAK
982
983 001526 012767 000340 176242 T2: MOV    #340,PS      ;DISABLE ALL INTERRUPTS
984 001534 012767 000020 010124      MOV    #20,ICOUNT  ;SET UP FOR 20 ITERATIONS
985 001542 012767 001756 010112      MOV    #4$,ESCAPE  ;SET UP TO ESCAPE TO NEXT TEST
986 001550 012777 004000 010042      MOV    #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
987 001556 004767 010144      JSR     PC,CLRALL  ;CLEAR ALL BUS ADDRESS AND

```

001600	012777	000001	010030	MOV	#1, 2DHSCR	: BUS ADDRESS MEMORY LOCATIONS
001601	012777	011722	010030	MOV	#NULL, 2DHBA	: SELECT LINE 1
001602	012777	177776	010024	MOV	#-2, 2DHBC	: SET UP TO TRANSMIT 0 CHARACTER
001603	012777	033503	010012	MOV	#33503, 2DHLPR	: TWO OS WILL BE TRANSMITTED
001612	012777	000002	010012	MOV	#2, 2DHBAR	: SET LINE SPEED=9600 BAUD
001620	122777	000002	010012	15: CMPB	#2, 2DHSLR	: CHARACTER LENGTH = 8 BITS
001626	001374			BNE	15	: SET BAR BIT FOR LINE 1
001630	012777	004000	007762	MOV	#BIT11, 2DHSCR	: WAIT FOR 2 CHARACTERS TO BE RECEIVED
001636	012777	000001	007754	MOV	#1, 2DHSCR	: MASTER CLEAR INTERFACE
001644	012777	011754	007754	MOV	#TBUF, 2DHBA	: SELECT LINE 1
001652	012777	177400	007750	MOV	#-400, 2DHBC	: SET UP TO TRANSMIT 400
001660	012777	033503	007735	MOV	#33503, 2DHLPR	: (OCTAL) CHARACTERS
001666	012777	000002	007740	MOV	#2, 2DHSCR	: LINE SPEED = 9600 BAUD
001674	012777	000002	007730	MOV	#2, 2DHBAR	: SET BREAK BIT FOR LINE 1
001702	005777	007724		25: TST	2DHBAR	: SET BAR BIT FOR LINE 1
001706	001375			BNE	25	: WAIT FOR ALL CHARACTERS
001710	122777	000001	007722	CMPB	#1, 2DHSLR	: TO BE TRANSMITTED
001716	001407			BEQ	35	: CHECK TO SEE THAT ONLY
001720	017704	007712		MOV	2DHSSR, R4	: 1 CHARACTER WAS RECEIVED
001724	042704	000300		BIC	#300, R4	: (R4)=ACTUAL RECEIVED DATA
001730	012705	000400		MOV	#400, R5	: CLEAR UNWANTED BITS
001734	104000			HLT	0	: (R5)=EXPECTED SILO FILL LEVEL, 1
001736	017704	007660		35: MOV	2DHNRC, R4	: MORE THAN ONE CHARACTER RECEIVED, ERROR
001742	026704	010452		CMP	RWORD1, R4	: READ NEXT RECEIVED CHARACTER REGISTER
001746	001403			BEQ	45	: IS RECEIVED CHARACTER A BREAK
001750	016705	010444		MOV	RWORD1, R5	: (R5)=EXPECTED RECEIVED CHARACTER
001754	104001			HLT	1	: RECEIVED DATA ERROR
001756	104400			45: SCOPE		: CHECK FOR ITERATIONS, LOOP
						: FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
						: ON LINE 2
						: SET BREAK BIT FOR LINE 2
						: TRANSMIT BINARY COUNT PATTERN ON LINE 2
						: VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
						: AND THAT IT IS A BREAK
001760	012767	000340	176010	T3: MOV	#340, PS	: DISABLE ALL INTERRUPTS
001766	012767	000020	007672	MOV	#20, ICOUNT	: SET UP FOR 20 ITERATIONS
001774	012767	002210	007660	MOV	#45, ESCAPE	: SET UP TO ESCAPE TO NEXT TEST
002002	012777	004000	007610	MOV	#BIT11, 2DHSCR	: MASTER CLEAR INTERFACE
002010	004767	007712		JSR	PC, CLRALL	: CLEAR ALL BUS ADDRESS AND
002014	012777	000002	007576	MOV	#2, 2DHSCR	: BUS ADDRESS MEMORY LOCATIONS
002022	012777	011722	007576	MOV	#NULL, 2DHBA	: SELECT LINE 2
002030	012777	177776	007572	MOV	#-2, 2DHBC	: SET UP TO TRANSMIT 0 CHARACTER
002036	012777	033503	007560	MOV	#33503, 2DHLPR	: TWO OS WILL BE TRANSMITTED
002044	012777	000004	007560	MOV	#4, 2DHBAR	: SET LINE SPEED=9600 BAUD
002052	122777	000002	007560	15: CMPB	#2, 2DHSLR	: CHARACTER LENGTH = 9 BITS
002060	001374			BNE	15	: SET BAR BIT FOR LINE 2
002062	012777	004000	007530	MOV	#BIT11, 2DHSCR	: WAIT FOR 2 CHARACTERS TO BE RECEIVED
002070	012777	000002	007522	MOV	#2, 2DHSCR	: MASTER CLEAR INTERFACE
002076	012777	011754	007522	MOV	#TBUF, 2DHBA	: SELECT LINE 2
002104	012777	177400	007516	MOV	#-400, 2DHBC	: SET UP TO TRANSMIT 400
						: (OCTAL) CHARACTERS

2204: 002122	002122	012777	003503	007504	MOV	#33503, 2DHLP	: LINE SPEED = 9600 BAUD
2204: 002123	002123	012777	000004	007506	MOV	#4, 2DHBCR	: SET BREAK BIT FOR LINE 2
2204: 002124	002124	012777	000004	007476	MOV	#4, 2DHBAR	: SET BAR BIT FOR LINE 2
2204: 002125	002125	005777	007472	25:	TST	2DHBAR	: WAIT FOR ALL CHARACTERS
2204: 002126	002126	001375			BNE	25	: TO BE TRANSMITTED
2204: 002127	002127	122777	000001	007470	CMPB	#1, 2DHSLR	: CHECK TO SEE THAT ONLY
2204: 002128	002128	001407			BEG	35	: 1 CHARACTER WAS RECEIVED
2204: 002129	002129	017704	007460		MOV	2DHSSR, R4	: (R4)=ACTUAL RECEIVED DATA
2204: 002130	002130	042704	000300		BIC	#300, R4	: CLEAR UNWANTED BITS
2204: 002131	002131	012705	000400		MOV	#400, R5	: (R5)=EXPECTED SILO FILL LEVEL, 1
2204: 002132	002132	104000			HLT	0	: MORE THAN ONE CHARACTER RECEIVED, ERROR
2204: 002133	002133	017704	007426	35:	MOV	2DHNRC, R4	: READ NEXT RECEIVED CHARACTER REGISTER
2204: 002134	002134	026704	010222		CMP	RWORD2, R4	: IS RECEIVED CHARACTER A BREAK
2204: 002135	002135	001403			BEG	45	
2204: 002136	002136	016705	010214		MOV	RWORD2, R5	: (R5)=EXPECTED RECEIVED CHARACTER
2204: 002137	002137	104001			HLT	1	: RECEIVED DATA ERROR
2204: 002138	002138	104400		45:	SCOPE		: CHECK FOR ITERATIONS, LOOP
2204: 002139	002139						
2204: 002140	002140						
2204: 002141	002141						
2204: 002142	002142						
2204: 002143	002143						
2204: 002144	002144						
2204: 002145	002145						
2204: 002146	002146						
2204: 002147	002147						
2204: 002148	002148						
2204: 002149	002149						
2204: 002150	002150						
2204: 002151	002151						
2204: 002152	002152						
2204: 002153	002153						
2204: 002154	002154						
2204: 002155	002155						
2204: 002156	002156						
2204: 002157	002157						
2204: 002158	002158						
2204: 002159	002159						
2204: 002160	002160						
2204: 002161	002161						
2204: 002162	002162						
2204: 002163	002163						
2204: 002164	002164						
2204: 002165	002165						
2204: 002166	002166						
2204: 002167	002167						
2204: 002168	002168						
2204: 002169	002169						
2204: 002170	002170						
2204: 002171	002171						
2204: 002172	002172						
2204: 002173	002173						
2204: 002174	002174						
2204: 002175	002175						
2204: 002176	002176						
2204: 002177	002177						
2204: 002178	002178						
2204: 002179	002179						
2204: 002180	002180						
2204: 002181	002181						
2204: 002182	002182						
2204: 002183	002183						
2204: 002184	002184						
2204: 002185	002185						
2204: 002186	002186						
2204: 002187	002187						
2204: 002188	002188						
2204: 002189	002189						
2204: 002190	002190						
2204: 002191	002191						
2204: 002192	002192						
2204: 002193	002193						
2204: 002194	002194						
2204: 002195	002195						
2204: 002196	002196						
2204: 002197	002197						
2204: 002198	002198						
2204: 002199	002199						

D03

DZDMT MACY11 27.1006 29-SEP-76 14:44 PAGE 27
 DZDMIC.P11 29-SEP-76 14:42

SEG 0026

```

1100 002432 001403 BEQ 4$
1101 002434 016705 MOV RWRD3,R5 ;R5:=EXPECTED RECEIVED CHARACTER
1102 002440 104001 HLT 1 ;RECEIVED DATA ERROR
1103 002442 104400 4$: SCOPE ;CHECK FOR ITERATIONS, LOOP
1104
1105 ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1106 ;ON LINE 4
1107 ;SET BREAK BIT FOR LINE 4
1108 ;TRANSMIT BINARY COUNT PATTERN ON LINE 4
1109 ;VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1110 ;AND THAT IT IS A BREAK
1111
1112 002444 012767 000340 175324 75: MOV #340,PS ;DISABLE ALL INTERRUPTS
1113 002452 012767 000020 007206 MOV #20,COUNT ;SET UP FOR 20 ITERATIONS
1114 002460 012767 002674 007174 MOV #4$,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST
1115 002466 012777 004000 007124 MOV #BIT11,JDHSCR ;MASTER CLEAR INTERFACE
1116 002474 004767 007226 JSR PC,CLRALL ;CLEAR ALL BUS ADDRESS AND
1117 ;BUS ADDRESS MEMORY LOCATIONS
1118 002500 012777 000004 007112 MOV #4,JDHSCR ;SELECT LINE 4
1119 002506 012777 011722 007112 MOV #NULL,JDHBA ;SET UP TO TRANSMIT 3 CHARACTER
1120 002514 012777 177776 007106 MOV #-2,JDHBC ;TWO 0S WILL BE TRANSMITTED
1121 002522 012777 033503 007074 MOV #33503,JDHLPR ;SET LINE SPEED=9600 BAUD
1122 ;CHARACTER LENGTH =8 BITS
1123 002530 012777 000020 007074 MOV #20,JDHBA ;SET BAR BIT FOR LINE 4
1124 002536 122777 000002 007074 15: CMPB #2,JDHSLR ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1125 002544 001374 BNE 15
1126 002546 012777 004000 007044 MOV #BIT11,JDHSCR ;MASTER CLEAR INTERFACE
1127 002554 012777 000004 007036 MOV #4,JDHSCR ;SELECT LINE 4
1128 002562 012777 011754 007036 MOV #TBUF,JDHBA ;SET UP TO TRANSMIT 400
1129 002570 012777 177400 007032 MOV #-400,JDHBC ;(OCTAL) CHARACTERS
1130 002576 012777 033503 007020 MOV #33503,JDHLPR ;LINE SPEED = 9600 BAUD
1131 002604 012777 000020 007022 MOV #20,JDHBCR ;SET BREAK BIT FOR LINE 4
1132 002612 012777 000020 007012 MOV #20,JDHBA ;SET BAR BIT FOR LINE 4
1133 002620 005777 007006 25: TST JDHBA ;WAIT FOR ALL CHARACTERS
1134 002624 001375 BNE 25 ;TO BE TRANSMITTED
1135 002626 122777 000001 007004 CMPB #1,JDHSLR ;CHECK TO SEE THAT ONLY
1136 002634 001407 BEQ 3$ ;1 CHARACTER WAS RECEIVED
1137 002636 017704 006774 MOV JDHSSR,R4 ;(R4)=ACTUAL RECEIVED DATA
1138 002642 042704 000300 BIC #300,R4 ;CLEAR UNWANTED BITS
1139 002646 012705 000400 MOV #400,R5 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1140 002652 104000 HLT 0 ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1141 002654 017704 006742 35: MOV JDHNR, R4 ;READ NEXT RECEIVED CHARACTER REGISTER
1142 002660 026704 007542 CMP RWRD4,R4 ;IS RECEIVED CHARACTER A BREAK
1143 002664 001403 BEQ 4$
1144 002666 016705 007534 MOV RWRD4,R5 ;(R5)=EXPECTED RECEIVED CHARACTER
1145 002672 104001 HLT 1 ;RECEIVED DATA ERROR
1146 002674 104400 4$: SCOPE ;CHECK FOR ITERATIONS, LOOP
1147
1148 ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1149 ;ON LINE 5
1150 ;SET BREAK BIT FOR LINE 5
1151 ;TRANSMIT BINARY COUNT PATTERN ON LINE 5
1152 ;VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1153 ;AND THAT IT IS A BREAK
1154
1155 002676 012767 000340 175072 76: MOV #340,PS ;DISABLE ALL INTERRUPTS

```

1156	002704	012767	000020	006754	MOV	#20,ICOUNT	:SET UP FOR 20 ITERATIONS
1157	002712	012767	003126	006742	MOV	#43,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1158	002720	012777	004000	006672	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1159	002726	004767	006754		JSR	PC,CLRAL	:CLEAR ALL BUS ADDRESS AND
1160							:BUS ADDRESS MEMORY LOCATIONS
1161	002732	012777	000005	006660	MOV	#5,ADHSCR	:SELECT LINE 5
1162	002740	012777	011722	006660	MOV	#NULL,ADHBA	:SET UPT TO TRANSMIT 0 CHARACTER
1163	002746	012777	177776	006554	MOV	#-2,ADHBC	:TWO 0S WILL BE TRANSMITTED
1164	002754	012777	033503	006642	MOV	#33503,ADHLPR	:SET LINE SPEED=9600 BAUD
1165							:CHARACTER LENGTH =8 BITS
1166	002762	012777	000040	006642	MOV	#40,ADHBAR	:SET BAR BIT FOR LINE 5
1167	002770	122777	000002	006642	15: CMPB	#2,ADHSLR	:WAIT FOR 2 CHARACTERS TO BE RECEIVED
1168	002776	001374			BNE	15	
1169	003000	012777	004000	006612	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1170	003006	012777	000005	006604	MOV	#5,ADHSCR	:SELECT LINE 5
1171	003014	012777	011754	006604	MOV	#TBJF,ADHBA	:SET UP TO TRANSMIT 400
1172	003022	012777	177400	006600	MOV	#-400,ADHBC	: (OCTAL) CHARACTERS
1173	003030	012777	033503	006566	MOV	#33503,ADHLPR	:LINE SPEED = 9600 BAUD
1174	003036	012777	000040	006570	MOV	#40,ADHBCR	:SET BREAK BIT FOR LINE 5
1175	003044	012777	000040	006560	MOV	#40,ADHBAR	:SET BAR BIT FOR LINE 5
1176	003052	005777	006554		25: TST	ADHBAR	:WAIT FOR ALL CHARACTERS
1177	003056	001375			BNE	25	:TO BE TRANSMITTED
1178	003060	122777	000001	006552	CMPB	#1,ADHSLR	:CHECK TO SEE THAT ONLY
1179	003066	001407			BEQ	35	:1 CHARACTER WAS RECEIVED
1180	003070	017704	006542		MOV	ADHSSR,R4	: (R4)=ACTUAL RECEIVED DATA
1181	003074	042704	000300		BIC	#300,R4	:CLEAR UNWANTED BITS
1182	003100	012705	000400		MOV	#400,R5	: (R5)=EXPECTED SILO FILL LEVEL, 1
1183	003104	104000			HLT	0	:MORE THAN ONE CHARACTER RECEIVED, ERROR
1184	003106	017704	006510		35: MOV	ADHNR0,R4	:READ NEXT RECEIVED CHARACTER REGISTER
1185	003112	026704	007312		CMP	RWRD5,R4	:IS RECEIVED CHARACTER A BREAK
1186	003116	001403			BEQ	45	
1187	003120	016705	007304		MOV	RWRD5,R5	: (R5)=EXPECTED RECEIVED CHARACTER
1188	003124	104001			HLT	1	:RECEIVED DATA ERROR
1189	003126	104400			45: SCOPE		:CHECK FOR ITERATIONS, LOOP
1190							
1191							
1192							:FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1193							:ON LINE 6
1194							:SET BREAK BIT FOR LINE 6
1195							:TRANSMIT BINARY COUNT PATTERN ON LINE 6
1196							:VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1197							:AND THAT IT IS A BREAK
1198	003130	012767	000340	174640	17: MOV	#340,PS	:DISABLE ALL INTERRUPTS
1199	003136	012767	000020	006522	MOV	#20,ICOUNT	:SET UP FOR 20 ITERATIONS
1200	003144	012767	003350	006510	MOV	#43,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1201	003152	012777	004000	006440	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1202	003160	004767	006542		JSR	PC,CLRAL	:CLEAR ALL BUS ADDRESS AND
1203							:BUS ADDRESS MEMORY LOCATIONS
1204	003164	012777	000006	006426	MOV	#6,ADHSCR	:SELECT LINE 6
1205	003172	012777	011722	006426	MOV	#NULL,ADHBA	:SET UPT TO TRANSMIT 0 CHARACTER
1206	003200	012777	177776	006422	MOV	#-2,ADHBC	:TWO 0S WILL BE TRANSMITTED
1207	003206	012777	033503	006410	MOV	#33503,ADHLPR	:SET LINE SPEED=9600 BAUD
1208							:CHARACTER LENGTH =8 BITS
1209	003214	012777	000100	006410	MOV	#100,ADHBAR	:SET BAR BIT FOR LINE 6
1210	003222	122777	000002	006410	15: CMPB	#2,ADHSLR	:WAIT FOR 2 CHARACTERS TO BE RECEIVED
1211	003230	001374			BNE	15	

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 29
 DZDHI.C.P11 29-SEP-76 14:42

SEQ 0028

1212	003232	012777	004000	006360		MOV	#BIT11, 2DHSCR	: MASTER CLEAR INTERFACE
1213	003240	012777	000006	006352		MOV	#6, 2DHSCR	: SELECT LINE 6
1214	003246	012777	011754	006352		MOV	#TBUF, 2DHBA	: SET UP TO TRANSMIT 400
1215	003254	012777	177400	006346		MOV	#-400, 2DHBC	: (OCTAL) CHARACTERS
1216	003262	012777	033503	006334		MOV	#33503, 2DHLPR	: LINE SPEED = 9600 BAUD
1217	003270	012777	000100	006336		MOV	#100, 2DHBCR	: SET BREAK BIT FOR LINE 6
1218	003276	012777	000100	006326		MOV	#100, 2DHBAR	: SET BAR BIT FOR LINE 6
1219	003304	005777	006322		25:	TST	2DHBAR	: WAIT FOR ALL CHARACTERS
1220	003310	001375				BNE	25	: TO BE TRANSMITTED
1221	003312	122777	000001	006320		CMPB	#1, 2DHSLR	: CHECK TO SEE THAT ONLY
1222	003320	001407				BEQ	35	: 1 CHARACTER WAS RECEIVED
1223	003322	017704	006310			MOV	2DHSSR, R4	: (R4)=ACTUAL RECEIVED DATA
1224	003326	042704	000300			BIC	#300, R4	: CLEAR UNWANTED BITS
1225	003332	012705	000400			MOV	#400, R5	: (R5)=EXPECTED SILO FILL LEVEL, 1
1226	003336	104000				HLT	0	: MORE THAN ONE CHARACTER RECEIVED, ERROR
1227	003340	017704	006256		35:	MOV	2DHNR, R4	: READ NEXT RECEIVED CHARACTER REGISTER
1228	003344	026704	007062			CMP	RWRD6, R4	: IS RECEIVED CHARACTER A BREAK
1229	003350	001403				BEQ	45	
1230	003352	016705	007054			MOV	RWRD6, R5	: (R5)=EXPECTED RECEIVED CHARACTER
1231	003356	104001				HLT	1	: RECEIVED DATA ERROR
1232	003360	104400			45:	SCOPE		: CHECK FOR ITERATIONS, LOOP
1233								
1234								: FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1235								: ON LINE 7
1236								: SET BREAK BIT FOR LINE 7
1237								: TRANSMIT BINARY COUNT PATTERN ON LINE 7
1238								: VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1239								: AND THAT IT IS A BREAK
1240								
1241	003362	012767	000340	174406	T10:	MOV	#340, PS	: DISABLE ALL INTERRUPTS
1242	003370	012767	000020	006270		MOV	#20, ICOUNT	: SET UP FOR 20 ITERATIONS
1243	003376	012767	003612	006256		MOV	#45, ESCAPE	: SET UP TO ESCAPE TO NEXT TEST
1244	003404	012777	004000	006206		MOV	#BIT11, 2DHSCR	: MASTER CLEAR INTERFACE
1245	003412	004767	006310			JSR	PC, CLRALL	: CLEAR ALL BUS ADDRESS AND
1246								: BUS ADDRESS MEMORY LOCATIONS
1247	003416	012777	000007	006174		MOV	#7, 2DHSCR	: SELECT LINE 7
1248	003424	012777	011722	006174		MOV	#NULL, 2DHBA	: SET UP TO TRANSMIT 0 CHARACTER
1249	003432	012777	177776	006170		MOV	#-2, 2DHBC	: TWO 0S WILL BE TRANSMITTED
1250	003440	012777	033503	006156		MOV	#33503, 2DHLPR	: SET LINE SPEED=9600 BAUD
1251								: CHARACTER LENGTH =8 BITS
1252	003446	012777	000200	006156		MOV	#200, 2DHBAR	: SET BAR BIT FOR LINE 7
1253	003454	122777	000002	006156	15:	CMPB	#2, 2DHSLR	: WAIT FOR 2 CHARACTERS TO BE RECEIVED
1254	003462	001374				BNE	15	
1255	003464	012777	004000	006126		MOV	#BIT11, 2DHSCR	: MASTER CLEAR INTERFACE
1256	003472	012777	000007	006120		MOV	#7, 2DHSCR	: SELECT LINE 7
1257	003500	012777	011754	006120		MOV	#TBUF, 2DHBA	: SET UP TO TRANSMIT 400
1258	003506	012777	177400	006114		MOV	#-400, 2DHBC	: (OCTAL) CHARACTERS
1259	003514	012777	033503	006102		MOV	#33503, 2DHLPR	: LINE SPEED = 9600 BAUD
1260	003522	012777	000200	006104		MOV	#200, 2DHBCR	: SET BREAK BIT FOR LINE 7
1261	003530	012777	000200	006074		MOV	#200, 2DHBAR	: SET BAR BIT FOR LINE 7
1262	003536	005777	006070		25:	TST	2DHBAR	: WAIT FOR ALL CHARACTERS
1263	003542	001375				BNE	25	: TO BE TRANSMITTED
1264	003544	122777	000001	006066		CMPB	#1, 2DHSLR	: CHECK TO SEE THAT ONLY
1265	003552	001407				BEQ	35	: 1 CHARACTER WAS RECEIVED
1266	003554	017704	006056			MOV	2DHSSR, R4	: (R4)=ACTUAL RECEIVED DATA
1267	003560	042704	000300			BIC	#300, R4	: CLEAR UNWANTED BITS

G03

DZDHI MACY11 27:006 29-SEP-76 14:44 PAGE 30
 DZDHI.C.P11 29-SEP-76 14:42

SEG 0029

```

1268 003564 012705 000400      MOV      #400,R5      ;(R5)=EXPECTED SILO FILL LEVEL, 1
1269 003570 104000      HLT      0      ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1270 003572 017704 006024      MOV      @DHNRC,R4      ;READ NEXT RECEIVED CHARACTER REGISTER
1271 003576 026704 006632      CMP      RWORD7,R4      ;IS RECEIVED CHARACTER A BREAK
1272 003602 001403      BEQ      4$
1273 003604 016705 006624      MOV      RWORD7,R5      ;(R5)=EXPECTED RECEIVED CHARACTER
1274 003610 104001      HLT      1      ;RECEIVED DATA ERROR
1275 003612 104400      4$: SCOPE      ;CHECK FOR ITERATIONS, LOOP

1276      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1277      ;ON LINE 10
1278      ;SET BREAK BIT FOR LINE 10
1279      ;TRANSMIT BINARY COUNT PATTERN ON LINE 10
1280      ;VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1281      ;AND THAT IT IS A BREAK

1284 003614 012767 000340 174154 T11: MOV      #340,PS      ;DISABLE ALL INTERRUPTS
1285 003622 012767 000020 006036      MOV      #20,COUNT      ;SET UP FOR 20 ITERATIONS
1286 003630 012767 004044 006024      MOV      #4$,ESCAPE      ;SET UP TO ESCAPE TO NEXT TEST
1287 003636 012777 004000 005754      MOV      #BIT11,@DHSCR      ;MASTER CLEAR INTERFACE
1288 003644 004767 006056      JSR      PC,CLRALL      ;CLEAR ALL BUS ADDRESS AND
1289      ;BUS ADDRESS MEMORY LOCATIONS
1290 003650 012777 000010 005742      MOV      #10,@DHSCR      ;SELECT LINE 10
1291 003656 012777 011722 005742      MOV      #NULL,@DHBA      ;SET UP TO TRANSMIT 0 CHARACTER
1292 003664 012777 177776 005736      MOV      #-2,@DHBC      ;TWO 0S WILL BE TRANSMITTED
1293 003672 012777 033503 005724      MOV      #33503,@DHLPR      ;SET LINE SPEED=9600 BAUD
1294      ;CHARACTER LENGTH=8 BITS
1295 003700 012777 000400 005724      MOV      #400,@DHBAR      ;SET BAR BIT FOR LINE 10
1296 003706 122777 000002 005724      1$: CMPB     #2,@DHSLR      ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1297 003714 001374      BNE      1$
1298 003716 012777 004000 005674      MOV      #BIT11,@DHSCR      ;MASTER CLEAR INTERFACE
1299 003724 012777 000010 005666      MOV      #10,@DHSCR      ;SELECT LINE 10
1300 003732 012777 011754 005666      MOV      #TBUF,@DHBA      ;SET UP TO TRANSMIT 400
1301 003740 012777 177400 005662      MOV      #-400,@DHBC      ;(OCTAL) CHARACTERS
1302 003746 012777 033503 005650      MOV      #33503,@DHLPR      ;LINE SPEED = 9600 BAUD
1303 003754 012777 000400 005652      MOV      #400,@DHBCR      ;SET BREAK BIT FOR LINE 10
1304 003762 012777 000400 005642      MOV      #400,@DHBAR      ;SET BAR BIT FOR LINE 10
1305 003770 005777 005636      2$: TST      @DHBAR      ;WAIT FOR ALL CHARACTERS
1306 003774 001375      BNE      2$      ;TO BE TRANSMITTED
1307 003776 122777 000001 005634      CMPB     #1,@DHSLR      ;CHECK TO SEE THAT ONLY
1308 004004 001407      BEQ      3$      ;1 CHARACTER WAS RECEIVED
1309 004006 017704 005624      MOV      @DHSSR,R4      ;(R4)=ACTUAL RECEIVED DATA
1310 004012 042704 000300      BIC      #300,R4      ;CLEAR UNWANTED BITS
1311 004016 012705 000400      MOV      #400,R5      ;(R5)=EXPECTED SILO FILL LEVEL, 1
1312 004022 104000      HLT      0      ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1313 004024 017704 005572      3$: MOV      @DHNRC,R4      ;READ NEXT RECEIVED CHARACTER REGISTER
1314 004030 026704 006402      CMP      RWORD10,R4      ;IS RECEIVED CHARACTER A BREAK
1315 004034 001403      BEQ      4$
1316 004036 016705 006374      MOV      RWORD10,R5      ;(R5)=EXPECTED RECEIVED CHARACTER
1317 004042 104001      HLT      1      ;RECEIVED DATA ERROR
1318 004044 104400      4$: SCOPE      ;CHECK FOR ITERATIONS, LOOP

1319      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1320      ;ON LINE 11
1321      ;SET BREAK BIT FOR LINE 11
1322      ;TRANSMIT BINARY COUNT PATTERN ON LINE 11
1323

```

H03

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 31
 DZDHI.C.P11 29-SEP-76 14:42

SEG 0030

```

1324                                     :VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1325                                     :AND THAT IT IS A BREAK
1326
1327 004046 012767 000340 173722 T12: MOV    #340,PS           :DISABLE ALL INTERRUPTS
1328 004054 012767 000020 005604      MOV    #20,ICOUNT        :SET UP FOR 20 ITERATIONS
1329 004062 012767 004276 005572      MOV    #45,ESCAPE        :SET UP TO ESCAPE TO NEXT TEST
1330 004070 012777 004000 005522      MOV    #BIT11,JDHSCR      :MASTER CLEAR INTERFACE
1331 004076 004767 005624              JSR     PC,CLRALL        :CLEAR ALL BUS ADDRESS AND
1332                                     :BUS ADDRESS MEMORY LOCATIONS
1333 004102 012777 000011 005510      MOV    #11,JDHSCR        :SELECT LINE 11
1334 004110 012777 011722 005510      MOV    #NULL,JDHBA       :SET UP TO TRANSMIT 0 CHARACTER
1335 004116 012777 177776 005504      MOV    #-2,JDHBC         :TWO 0S WILL BE TRANSMITTED
1336 004124 012777 033503 005472      MOV    #33503,JDHLPR      :SET LINE SPEED=9600 BAUD
1337                                     :CHARACTER LENGTH =8 BITS
1338 004132 012777 001000 005472      MOV    #1000,JDHBA       :SET BAR BIT FOR LINE 11
1339 004140 122777 000002 005472 1S:  CMPB   #2,JDHSLR        :WAIT FOR 2 CHARACTERS TO BE RECEIVED
1340 004146 001374              BNE     1S
1341 004150 012777 004000 005442      MOV    #BIT11,JDHSCR      :MASTER CLEAR INTERFACE
1342 004156 012777 000011 005434      MOV    #11,JDHSCR        :SELECT LINE 11
1343 004164 012777 011754 005434      MOV    #TBUF,JDHBA       :SET UP TO TRANSMIT 400
1344 004172 012777 177400 005430      MOV    #-400,JDHBC        :OCTAL) CHARACTERS
1345 004200 012777 033503 005416      MOV    #33503,JDHLPR      :LINE SPEED = 9600 BAUD
1346 004206 012777 001000 005420      MOV    #1000,JDHBCR      :SET BREAK BIT FOR LINE 11
1347 004214 012777 001000 005410      MOV    #1000,JDHBA       :SET BAR BIT FOR LINE 11
1348 004222 005777 005404 2S:  TST     JDHBA          :WAIT FOR ALL CHARACTERS
1349 004226 001375              BNE     2S          :TO BE TRANSMITTED
1350 004230 122777 000001 005402      CMPB   #1,JDHSLR        :CHECK TO SEE THAT ONLY
1351 004236 001407              BEQ     3S          :1 CHARACTER WAS RECEIVED
1352 004240 017704 005372      MOV    JDHSSR,R4          : (R4)=ACTUAL RECEIVED DATA
1353 004244 042704 000300      BIC     #300,R4          :CLEAR UNWANTED BITS
1354 004250 012705 000400      MOV    #400,R5          : (R5)=EXPECTED SILO FILL LEVEL, 1
1355 004254 104000              HLT     0              :MORE THAN ONE CHARACTER RECEIVED, ERROR
1356 004256 017704 005340 3S:  MOV    JDHNR,R4          :READ NEXT RECEIVED CHARACTER REGISTER
1357 004262 026704 006152      CMP     RWORD11,R4        :IS RECEIVED CHARACTER A BREAK
1358 004266 001403              BEQ     4S
1359 004270 016705 006144      MOV    RWORD11,R5          : (R5)=EXPECTED RECEIVED CHARACTER
1360 004274 104001              HLT     1              :RECEIVED DATA ERROR
1361 004276 104400 4S:  SCOPE          :CHECK FOR ITERATIONS, LOOP
1362
1363                                     :FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1364                                     :ON LINE 12
1365                                     :SET BREAK BIT FOR LINE 12
1366                                     :TRANSMIT BINARY COUNT PATTERN ON LINE 12
1367                                     :VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1368                                     :AND THAT IT IS A BREAK
1369
1370 004300 012767 000340 173470 T13: MOV    #340,PS           :DISABLE ALL INTERRUPTS
1371 004306 012767 000020 005352      MOV    #20,ICOUNT        :SET UP FOR 20 ITERATIONS
1372 004314 012767 004530 005340      MOV    #45,ESCAPE        :SET UP TO ESCAPE TO NEXT TEST
1373 004322 012777 004000 005270      MOV    #BIT11,JDHSCR      :MASTER CLEAR INTERFACE
1374 004330 004767 005372              JSR     PC,CLRALL        :CLEAR ALL BUS ADDRESS AND
1375                                     :BUS ADDRESS MEMORY LOCATIONS
1376 004334 012777 000012 005256      MOV    #12,JDHSCR        :SELECT LINE 12
1377 004342 012777 011722 005256      MOV    #NULL,JDHBA       :SET UP TO TRANSMIT 0 CHARACTER
1378 004350 012777 177776 005252      MOV    #-2,JDHBC         :TWO 0S WILL BE TRANSMITTED
1379 004356 012777 033503 005240      MOV    #33503,JDHLPR      :SET LINE SPEED=9600 BAUD

```

```

1380
1381 004364 012777 002000 005240      MOV      #2000,2DHBAR      ;CHARACTER LENGTH =8 BITS
1382 004372 122777 003002 005240 1$: CMPB      #2,2DHSR      ;SET BAR BIT FOR LINE 12
1383 004400 001374      BNE      1$      ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1384 004402 012777 004000 005210      MOV      #BIT11,2DHSCR      ;MASTER CLEAR INTERFACE
1385 004410 012777 000012 005202      MOV      #12,2DHSCR      ;SELECT LINE 12
1386 004416 012777 011754 005202      MOV      #TBUF,2DHBA      ;SET UP TO TRANSMIT 400
1387 004424 012777 177400 005176      MOV      #-400,2DHBC      ;(OCTAL) CHARACTERS
1388 004432 012777 033503 005164      MOV      #33503,2DHLPR      ;LINE SPEED = 9600 BAUD
1389 004440 012777 002000 005166      MOV      #2000,2DHBCR      ;SET BREAK BIT FOR LINE 12
1390 004446 012777 002000 005156      MOV      #2000,2DHBR      ;SET BAR BIT FOR LINE 12
1391 004454 005777 005152 2$: TST      2DHBR      ;WAIT FOR ALL CHARACTERS
1392 004460 001375      BNE      2$      ;TO BE TRANSMITTED
1393 004462 122777 000001 005150      CMPB      #1,2DHSR      ;CHECK TO SEE THAT ONLY
1394 004470 001407      BEQ      3$      ;1 CHARACTER WAS RECEIVED
1395 004472 017704 005140      MOV      2DHSSR,R4      ;(R4)=ACTUAL RECEIVED DATA
1396 004476 042704 000300      BIC      #300,R4      ;CLEAR UNWANTED BITS
1397 004502 012705 000400      MOV      #400,R5      ;(R5)=EXPECTED SILO FILL LEVEL, 1
1398 004506 104000      HLT      0      ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1399 004510 017704 005106 3$: MOV      2DHNR,R4      ;READ NEXT RECEIVED CHARACTER REGISTER
1400 004514 026704 005722      CMP      RWRD12,R4      ;IS RECEIVED CHARACTER A BREAK
1401 004520 001403      BEQ      4$      ;(R5)=EXPECTED RECEIVED CHARACTER
1402 004522 016705 005714      MOV      RWRD12,R5      ;RECEIVED DATA ERROR
1403 004526 104001      HLT      1      ;CHECK FOR ITERATIONS, LOOP
1404 004530 104400 4$: SCOPE
1405
1406      ;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1407      ;ON LINE 13
1408      ;SET BREAK BIT FOR LINE 13
1409      ;TRANSMIT BINARY COUNT PATTERN ON LINE 13
1410      ;VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1411      ;AND THAT IT IS A BREAK
1412
1413 004532 012767 000340 173236 T14: MOV      #340,P5      ;DISABLE ALL INTERRUPTS
1414 004540 012767 000020 005120      MOV      #20,ICOUNT      ;SET UP FOR 20 ITERATIONS
1415 004546 012767 004762 005106      MOV      #4$,ESCAPE      ;SET UP TO ESCAPE TO NEXT TEST
1416 004554 012777 004000 005036      MOV      #BIT11,2DHSCR      ;MASTER CLEAR INTERFACE
1417 004562 004767 005140      JSR      PC,CLRALL      ;CLEAR ALL BUS ADDRESS AND
1418      ;BUS ADDRESS MEMORY LOCATIONS
1419 004566 012777 000013 005024      MOV      #13,2DHSCR      ;SELECT LINE 13
1420 004574 012777 011722 005024      MOV      #NULL,2DHBA      ;SET UP TO TRANSMIT 0 CHARACTER
1421 004602 012777 177776 005020      MOV      #-2,2DHBC      ;TWO 0S WILL BE TRANSMITTED
1422 004610 012777 033503 005006      MOV      #33503,2DHLPR      ;SET LINE SPEED=9600 BAUD
1423      ;CHARACTER LENGTH =8 BITS
1424 004616 012777 004000 005006      MOV      #4000,2DHBR      ;SET BAR BIT FOR LINE 13
1425 004624 122777 000002 005006 1$: CMPB      #2,2DHSR      ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1426 004632 001374      BNE      1$
1427 004634 012777 004000 004756      MOV      #BIT11,2DHSCR      ;MASTER CLEAR INTERFACE
1428 004642 012777 000013 004750      MOV      #13,2DHSCR      ;SELECT LINE 13
1429 004650 012777 011754 004750      MOV      #TBUF,2DHBA      ;SET UP TO TRANSMIT 400
1430 004656 012777 177400 004744      MOV      #-400,2DHBC      ;(OCTAL) CHARACTERS
1431 004664 012777 033503 004732      MOV      #33503,2DHLPR      ;LINE SPEED = 9600 BAUD
1432 004672 012777 004000 004734      MOV      #4000,2DHBCR      ;SET BREAK BIT FOR LINE 13
1433 004700 012777 004000 004724      MOV      #4000,2DHBR      ;SET BAR BIT FOR LINE 13
1434 004706 005777 004720 2$: TST      2DHBR      ;WAIT FOR ALL CHARACTERS
1435 004712 001375      BNE      2$      ;TO BE TRANSMITTED

```

1436	004714	122777	000001	004716		CMPB	#1,ADHSLR	:CHECK TO SEE THAT ONLY
1437	004722	001407				BEQ	3\$	:1 CHARACTER WAS RECEIVED
1438	004724	017704	004706			MOV	ADHSSR,R4	:(R4)=ACTUAL RECEIVED DATA
1439	004730	042704	000300			BIC	#300,R4	:CLEAR UNWANTED BITS
1440	004734	012705	000400			MOV	#400,R5	:(R5)=EXPECTED SILO FILL LEVEL, 1
1441	004740	104000				HLT	0	:MORE THAN ONE CHARACTER RECEIVED, ERROR
1442	004742	017704	004654	3\$:		MOV	ADHNRC,R4	:READ NEXT RECEIVED CHARACTER REGISTER
1443	004746	026704	005472			CMP	RWRD13,R4	:IS RECEIVED CHARACTER A BREAK
1444	004752	001403				BEQ	4\$	
1445	004754	016705	005464			MOV	RWRD13,R5	:(R5)=EXPECTED RECEIVED CHARACTER
1446	004760	104001				HLT	1	:RECEIVED DATA ERROR
1447	004762	104400		4\$:		SCOPE		:CHECK FOR ITERATIONS, LOOP
1448								
1449								
1450								
1451								
1452								
1453								
1454								
1455								
1456	004764	012767	000340	173004	T15:	MOV	#340,PS	:DISABLE ALL INTERRUPTS
1457	004772	012767	000020	004666		MOV	#20,ICOUNT	:SET UP FOR 20 ITERATIONS
1458	005000	012767	005214	004654		MOV	#4\$ ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1459	005006	012777	004000	004604		MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1460	005014	004767	004706			JSR	PC,CLRALL	:CLEAR ALL BUS ADDRESS AND
1461								:BUS ADDRESS MEMORY LOCATIONS
1462	005020	012777	000014	004572		MOV	#14,ADHSCR	:SELECT LINE 14
1463	005026	012777	011722	004572		MOV	#NULL,ADHBA	:SET UP TO TRANSMIT 0 CHARACTER
1464	005034	012777	177776	004566		MOV	#-2,ADHBC	:TWO OS WILL BE TRANSMITTED
1465	005042	012777	033503	004554		MOV	#33503,ADHLPR	:SET LINE SPEED=9600 BAUD
1466								:CHARACTER LENGTH =8 BITS
1467	005050	012777	010000	004554		MOV	#10000,ADHBAR	:SET BAR BIT FOR LINE 14
1468	005056	122777	000002	004554	1\$:	CMPB	#2,ADHSLR	:WAIT FOR 2 CHARACTERS TO BE RECEIVED
1469	005064	001374				BNE	1\$	
1470	005066	012777	004000	004524		MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1471	005074	012777	000014	004516		MOV	#14,ADHSCR	:SELECT LINE 14
1472	005102	012777	011754	004516		MOV	#TBUF,ADHBA	:SET UP TO TRANSMIT 400
1473	005110	012777	177400	004512		MOV	#-400,ADHBC	: (OCTAL) CHARACTERS
1474	005116	012777	033503	004500		MOV	#33503,ADHLPR	:LINE SPEED = 9600 BAUD
1475	005124	012777	010000	004502		MOV	#10000,ADHBCR	:SET BREAK BIT FOR LINE 14
1476	005132	012777	010000	004472		MOV	#10000,ADHBAR	:SET BAR BIT FOR LINE 14
1477	005140	005777	004466		2\$:	TST	ADHBAR	:WAIT FOR ALL CHARACTERS
1478	005144	001375				BNE	2\$	:TO BE TRANSMITTED
1479	005146	122777	000001	004464		CMPB	#1,ADHSLR	:CHECK TO SEE THAT ONLY
1480	005154	001407				BEQ	3\$	:1 CHARACTER WAS RECEIVED
1481	005156	017704	004454			MOV	ADHSSR,R4	:(R4)=ACTUAL RECEIVED DATA
1482	005162	042704	000300			BIC	#300,R4	:CLEAR UNWANTED BITS
1483	005166	012705	000400			MOV	#400,R5	:(R5)=EXPECTED SILO FILL LEVEL, 1
1484	005172	104000				HLT	0	:MORE THAN ONE CHARACTER RECEIVED, ERROR
1485	005174	017704	004422	3\$:		MOV	ADHNRC,R4	:READ NEXT RECEIVED CHARACTER REGISTER
1486	005200	026704	005242			CMP	RWRD14,R4	:IS RECEIVED CHARACTER A BREAK
1487	005204	001403				BEQ	4\$	
1488	005206	016705	005234			MOV	RWRD14,R5	:(R5)=EXPECTED RECEIVED CHARACTER
1489	005212	104001				HLT	1	:RECEIVED DATA ERROR
1490	005214	104400		4\$:		SCOPE		:CHECK FOR ITERATIONS, LOOP
1491								

;FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
 ;ON LINE 14
 ;SET BREAK BIT FOR LINE 14
 ;TRANSMIT BINARY COUNT PATTERN ON LINE 14
 ;VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
 ;AND THAT IT IS A BREAK

K03

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 34
 DZDHI.C.P11 29-SEP-76 14:42

SEG 0033

```

1492 ;FLUSH JART BY TRANSMITTING 2 NULL CHARACTERS
1493 ;ON LINE 15
1494 ;SET BREAK BIT FOR LINE 15
1495 ;TRANSMIT BINARY COUNT PATTERN ON LINE 15
1496 ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1497 ;AND THAT IT IS A BREAK
1498
1499 005216 012767 000340 172552 T16: MOV #340,PS ;DISABLE ALL INTERRUPTS
1500 005224 012767 000020 004434 MOV #20,ICOUNT ;SET UP FOR 20 ITERATIONS
1501 005232 012767 005446 004422 MOV #45,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST
1502 005240 012777 004000 004352 MOV #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
1503 005246 004767 004454 JSR PC,CLRALL ;CLEAR ALL BUS ADDRESS AND
1504 ;BUS ADDRESS MEMORY LOCATIONS
1505 005252 012777 000015 004340 MOV #15,ADHSCR ;SELECT LINE 15
1506 005260 012777 011722 004340 MOV #NULL,ADHBA ;SET UPT TO TRANSMIT 0 CHARACTER
1507 005266 012777 177776 004334 MOV #-2,ADHBC ;TWO OS WILL BE TRANSMITTED
1508 005274 012777 033503 004322 MOV #33503,ADHLPR ;SET LINE SPEED=9600 BAUD
1509 ;CHARACTER LENGTH =8 BITS
1510 005302 012777 020000 004322 MOV #20000,ADHBAR ;SET BAR BIT FOR LINE 15
1511 005310 122777 000002 004322 1$: CMPB #2,ADHSLR ;WAIT FOR 2 CHARACTERS TO BE RECEIVED
1512 005316 001374 BNE 1$
1513 005320 012777 004000 004272 MOV #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
1514 005326 012777 000015 004264 MOV #15,ADHSCR ;SELECT LINE 15
1515 005334 012777 011754 004264 MOV #TBUF,ADHBA ;SET UP TO TRANSMIT 400
1516 005342 012777 177400 004260 MOV #-400,ADHBC ;(OCTAL) CHARACTERS
1517 005350 012777 033503 004246 MOV #33503,ADHLPR ;LINE SPEED = 9600 BAUD
1518 005356 012777 020000 004250 MOV #20000,ADHBCR ;SET BREAK BIT FOR LINE 15
1519 005364 012777 020000 004240 MOV #20000,ADHBAR ;SET BAR BIT FOR LINE 15
1520 005372 005777 004234 2$: TST ADHBAR ;WAIT FOR ALL CHARACTERS
1521 005376 001375 BNE 2$ ;TO BE TRANSMITTED
1522 005400 122777 000001 004232 CMPB #1,ADHSLR ;CHECK TO SEE THAT ONLY
1523 005406 001407 BEQ 3$ ;1 CHARACTER WAS RECEIVED
1524 005410 017704 004222 MOV ADHSSR,R4 ;(R4)=ACTUAL RECEIVED DATA
1525 005414 042704 000300 BIC #300,R4 ;CLEAR UNWANTED BITS
1526 005420 012705 000400 MOV #400,R5 ;(R5)=EXPECTED SILO FILL LEVEL, 1
1527 005424 104000 HLT 0 ;MORE THAN ONE CHARACTER RECEIVED, ERROR
1528 005426 017704 004170 3$: MOV ADHNR0,R4 ;READ NEXT RECEIVED CHARACTER REGISTER
1529 005432 026704 005012 CMP RWRD15,R4 ;IS RECEIVED CHARACTER A BREAK
1530 005436 001403 BEQ 4$
1531 005440 016705 005004 MOV RWRD15,R5 ;(R5)=EXPECTED RECEIVED CHARACTER
1532 005444 104001 HLT 1 ;RECEIVED DATA ERROR
1533 005446 104400 4$: SCOPE ;CHECK FOR ITERATIONS, LOOP
1534
1535 ;FLUSH JART BY TRANSMITTING 2 NULL CHARACTERS
1536 ;ON LINE 16
1537 ;SET BREAK BIT FOR LINE 16
1538 ;TRANSMIT BINARY COUNT PATTERN ON LINE 16
1539 ;VERIFY HTAT ONLY 1 CHARACTER IS RECEIVED
1540 ;AND THAT IT IS A BREAK
1541
1542 005450 012767 000340 172320 T17: MOV #340,PS ;DISABLE ALL INTERRUPTS
1543 005456 012767 000020 004202 MOV #20,ICOUNT ;SET UP FOR 20 ITERATIONS
1544 005464 012767 005700 004170 MOV #45,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST
1545 005472 012777 004000 004120 MOV #BIT11,ADHSCR ;MASTER CLEAR INTERFACE
1546 005500 004767 004222 JSR PC,CLRALL ;CLEAR ALL BUS ADDRESS AND
1547 ;BUS ADDRESS MEMORY LOCATIONS

```

1548	005504	012777	000016	004106	MOV	#16,ADHSCR	:SELECT LINE 16
1549	005512	012777	011722	004106	MOV	#NJLL,ADHBA	:SET UP TO TRANSMIT 0 CHARACTER
1550	005520	012777	177776	004102	MOV	#-2,ADHBC	:TWO OS WILL BE TRANSMITTED
1551	005526	012777	033503	004070	MOV	#33503,ADHLPR	:SET LINE SPEED=9600 BAUD
1552							:CHARACTER LENGTH =8 BITS
1553	005534	012777	040000	004070	MOV	#40000,ADHBAR	:SET BAR BIT FOR LINE 16
1554	005542	122777	000002	004070	15: CMPB	#2,ADHSLR	:WAIT FOR 2 CHARACTERS TO BE RECEIVED
1555	005550	001374			BNE	15	
1556	005552	012777	004000	004040	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1557	005560	012777	000016	004032	MOV	#16,ADHSCR	:SELECT LINE 16
1558	005566	012777	011754	004032	MOV	#TBUF,ADHBA	:SET UP TO TRANSMIT 400
1559	005574	012777	177400	004026	MOV	#-400,ADHBC	: (OCTAL) CHARACTERS
1560	005602	012777	033503	004014	MOV	#33503,ADHLPR	:LINE SPEED = 9600 BAUD
1561	005610	012777	040000	004016	MOV	#40000,ADHBCR	:SET BREAK BIT FOR LINE 16
1562	005616	012777	040000	004006	MOV	#40000,ADHBAR	:SET BAR BIT FOR LINE 16
1563	005624	005777	004002		25: TST	ADHBAR	:WAIT FOR ALL CHARACTERS
1564	005630	001375			BNE	25	:TO BE TRANSMITTED
1565	005632	122777	000001	004000	CMPB	#1,ADHSLR	:CHECK TO SEE THAT ONLY
1566	005640	001407			BEQ	35	:1 CHARACTER WAS RECEIVED
1567	005642	017704	003770		MOV	ADHSSR,R4	: (R4)=ACTUAL RECEIVED DATA
1568	005646	042704	000300		BIC	#300,R4	:CLEAR UNWANTED BITS
1569	005652	012705	000400		MOV	#400,R5	: (R5)=EXPECTED SILO FILL LEVEL, 1
1570	005656	104000			HLT	0	:MORE THAN ONE CHARACTER RECEIVED, ERROR
1571	005660	017704	003736		35: MOV	ADHNRC,R4	:READ NEXT RECEIVED CHARACTER REGISTER
1572	005664	026704	004562		CMP	RWRD16,R4	:IS RECEIVED CHARACTER A BREAK
1573	005670	001403			BEQ	45	
1574	005672	016705	004554		MOV	RWRD16,R5	: (R5)=EXPECTED RECEIVED CHARACTER
1575	005676	104001			HLT	1	:RECEIVED DATA ERROR
1576	005700	104400			45: SCOPE		:CHECK FOR ITERATIONS, LOOP
1577							
1578							:FLUSH UART BY TRANSMITTING 2 NULL CHARACTERS
1579							:ON LINE 17
1580							:SET BREAK BIT FOR LINE 17
1581							:TRANSMIT BINARY COUNT PATTERN ON LINE 17
1582							:VERIFY THAT ONLY 1 CHARACTER IS RECEIVED
1583							:AND THAT IT IS A BREAK
1584							
1585	005702	012767	000340	172066	T20: MOV	#340,PS	:DISABLE ALL INTERRUPTS
1586	005710	012767	000020	003750	MOV	#20,COUNT	:SET UP FOR 20 ITERATIONS
1587	005716	012767	006132	003736	MOV	#45,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1588	005724	012777	004000	003666	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1589	005732	004767	003770		JSR	PC,CLALL	:CLEAR ALL BUS ADDRESS AND
1590							:BUS ADDRESS MEMORY LOCATIONS
1591	005736	012777	000017	003654	MOV	#17,ADHSCR	:SELECT LINE 17
1592	005744	012777	011722	003654	MOV	#NULL,ADHBA	:SET UP TO TRANSMIT 0 CHARACTER
1593	005752	012777	177776	003650	MOV	#-2,ADHBC	:TWO OS WILL BE TRANSMITTED
1594	005760	012777	033503	003636	MOV	#33503,ADHLPR	:SET LINE SPEED=9600 BAUD
1595							:CHARACTER LENGTH =8 BITS
1596	005766	012777	100000	003636	MOV	#100000,ADHBAR	:SET BAR BIT FOR LINE 17
1597	005774	122777	000002	003636	15: CMPB	#2,ADHSLR	:WAIT FOR 2 CHARACTERS TO BE RECEIVED
1598	006002	001374			BNE	15	
1599	006004	012777	004000	003606	MOV	#BIT11,ADHSCR	:MASTER CLEAR INTERFACE
1600	006012	012777	000017	003600	MOV	#17,ADHSCR	:SELECT LINE 17
1601	006020	012777	011754	003600	MOV	#TBUF,ADHBA	:SET UP TO TRANSMIT 400
1602	006026	012777	177400	003574	MOV	#-400,ADHBC	: (OCTAL) CHARACTERS
1603	006034	012777	033503	003562	MOV	#33503,ADHLPR	:LINE SPEED = 9600 BAUD

1604	006042	012777	100000	003564		MOV	#100000,ADHBCR	;SET BREAK BIT FOR LINE 17
1605	006050	012777	100000	003554		MOV	#100000,ADHBAR	;SET BAR BIT FOR LINE 17
1606	006056	005777	003550		2\$:	TST	ADHBAR	;WAIT FOR ALL CHARACTERS
1607	006062	001375				BNE	2\$	;TO BE TRANSMITTED
1608	006064	122777	000001	003546		CMPB	#1,ADHSLR	;CHECK TO SEE THAT ONLY
1609	006072	001407				BEQ	3\$	;1 CHARACTER WAS RECEIVED
1610	006074	017704	003536			MOV	ADHSSR,R4	; (R4)=ACTUAL RECEIVED DATA
1611	006100	042704	000300			BIC	#300,R4	;CLEAR UNWANTED BITS
1612	006104	012705	000400			MOV	#400,R5	; (R5)=EXPECTED SILO FILL LEVEL, 1
1613	006110	104000				HLT	0	;MORE THAN ONE CHARACTER RECEIVED, ERROR
1614	006112	017704	003504		3\$:	MOV	ADHNRG,R4	;READ NEXT RECEIVED CHARACTER REGISTER
1615	006116	026704	004332			CMP	RWRD17,R4	;IS RECEIVED CHARACTER A BREAK
1616	006122	001403				BEQ	4\$	
1617	006124	016705	004324			MOV	RWRD17,R5	; (R5)=EXPECTED RECEIVED CHARACTER
1618	006130	104001				HLT	1	;RECEIVED DATA ERROR
1619	006132	104400			4\$:	SCOPE		;CHECK FOR ITERATIONS, LOOP
1620								
1621								
1622								
1623								
1624								
1625	006134	012767	000340	171634	T21:	MOV	#340,PS	;DISABLE ALL INTERRUPTS
1626	006142	012767	000020	003516		MOV	#20,ICOUNT	;SET UP FOR 20 ITERATIONS
1627	006150	012767	006244	003504		MOV	#2\$,ESCAPE	;SET UP TO ESCAPE TO NEXT TEST
1628	006156	012777	004000	003434		MOV	#BIT11,ADHSCR	;MASTER CLEAR INTERFACE
1629	006164	004767	003536			JSR	PC,CLRALL	;CLEAR ALL BYTE COUNT AND
1630								;AND BUS ADDRESS MEMORY LOCATIONS
1631	006170	012777	000000	003422		MOV	#0,ADHSCR	;SELECT LINE 0
1632	006176	012777	011754	003422		MOV	#TBUF,ADHBA	;SET UP TO TRANSMIT
1633	006204	012777	177400	003416		MOV	#-400,ADHBC	;400 (OCTAL) CHARACTERS
1634	006212	012777	073503	003404		MOV	#73503,ADHLPR	;SET RECEIVER BLIND
1635								;LINE SPEED =9600 BAUD
1636								;CHARACTER LENGTH = 8 BITS
1637	006220	012777	000001	003404		MOV	#1,ADHBAR	;SET BAR BIT FOR LINE 0
1638	006226	005777	003400		1\$:	TST	ADHBAR	;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1639	006232	001375				BNE	1\$	
1640	006234	105777	003360			TSTB	ADHSCR	;WERE ANY CHARACTERS RECEIVED
1641	006240	100001				BPL	2\$	
1642	006242	104002				HLT	2	;RECEIVER NOT BLINDED, ERROR
1643	006244	104400			2\$:	SCOPE		;CHECK FOR ITERATIONS, LOOP
1644								
1645								
1646								
1647								
1648								
1649	006246	012767	000340	171522	T22:	MOV	#340,PS	;DISABLE ALL INTERRUPTS
1650	006254	012767	000020	003404		MOV	#20,ICOUNT	;SET UP FOR 20 ITERATIONS
1651	006262	012767	006356	003372		MOV	#2\$,ESCAPE	;SET UP TO ESCAPE TO NEXT TEST
1652	006270	012777	004000	003322		MOV	#BIT11,ADHSCR	;MASTER CLEAR INTERFACE
1653	006276	004767	003424			JSR	PC,CLRALL	;CLEAR ALL BYTE COUNT AND
1654								;AND BUS ADDRESS MEMORY LOCATIONS
1655	006302	012777	000001	003310		MOV	#1,ADHSCR	;SELECT LINE 1
1656	006310	012777	011754	003310		MOV	#TBUF,ADHBA	;SET UP TO TRANSMIT
1657	006316	012777	177400	003304		MOV	#-400,ADHBC	;400 (OCTAL) CHARACTERS
1658	006324	012777	073503	003272		MOV	#73503,ADHLPR	;SET RECEIVER BLIND
1659								;LINE SPEED =9600 BAUD


```

1716
1717
1718
1719
1720
1721 006604 012767 000340 171164 T25: MOV #340,PS ;DISABLE ALL INTERRUPTS
1722 006612 012767 000020 003046 MOV #20,ICOUNT ;SET UP FOR 20 ITERATIONS
1723 006620 012767 006714 003034 MOV #25,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST
1724 006626 012777 004000 002764 MOV #BIT11,2DHSCR ;MASTER CLEAR INTERFACE
1725 006634 004767 003066 JSR PC,CLRALL ;CLEAR ALL BYTE COUNT AND
1726 ;AND BUS ADDRESS MEMORY LOCATIONS
1727 006640 012777 000004 002752 MOV #4,2DHSCR ;SELECT LINE 4
1728 006646 012777 011754 002752 MOV #TBUF,2DHBA ;SET UP TO TRANSMIT
1729 006654 012777 177400 002746 MOV #400,2DHBC ;400 (OCTAL) CHARACTERS
1730 006662 012777 073503 002734 MOV #73503,2DHLPR ;SET RECEIVER BLIND
1731 ;LINE SPEED = 9600 BAUD
1732 ;CHARACTER LENGTH = 8 BITS
1733 006670 012777 000020 002734 MOV #20,2DHBAR ;SET BAR BIT FOR LINE 4
1734 006676 005777 002730 15: TST 2DHBAR ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1735 006702 001375 BNE 15 ;WERE ANY CHARACTERS RECEIVED
1736 006704 105777 002710 TSTB 2DHSCR
1737 006710 100001 BPL 25
1738 006712 104002 HLT 2 ;RECEIVER NOT BLINDED. ERROR
1739 006714 104400 25: SCOPE ;CHECK FOR ITERATIONS. LOOP
1740
1741
1742
1743
1744
1745 006716 012767 000340 171052 T26: MOV #340,PS ;DISABLE ALL INTERRUPTS
1746 006724 012767 000020 002734 MOV #20,ICOUNT ;SET UP FOR 20 ITERATIONS
1747 006732 012767 007026 002722 MOV #25,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST
1748 006740 012777 004000 002652 MOV #BIT11,2DHSCR ;MASTER CLEAR INTERFACE
1749 006746 004767 002754 JSR PC,CLRALL ;CLEAR ALL BYTE COUNT AND
1750 ;AND BUS ADDRESS MEMORY LOCATIONS
1751 006752 012777 000005 002640 MOV #5,2DHSCR ;SELECT LINE 5
1752 006760 012777 011754 002640 MOV #TBUF,2DHBA ;SET UP TO TRANSMIT
1753 006766 012777 177400 002634 MOV #400,2DHBC ;400 (OCTAL) CHARACTERS
1754 006774 012777 073503 002622 MOV #73503,2DHLPR ;SET RECEIVER BLIND
1755 ;LINE SPEED = 9600 BAUD
1756 ;CHARACTER LENGTH = 8 BITS
1757 007002 012777 000040 002622 MOV #40,2DHBAR ;SET BAR BIT FOR LINE 5
1758 007010 005777 002616 15: TST 2DHBAR ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1759 007014 001375 BNE 15 ;WERE ANY CHARACTERS RECEIVED
1760 007016 105777 002576 TSTB 2DHSCR
1761 007022 100001 BPL 25
1762 007024 104002 HLT 2 ;RECEIVER NOT BLINDED. ERROR
1763 007026 104400 25: SCOPE ;CHECK FOR ITERATIONS. LOOP
1764
1765
1766
1767
1768
1769 007030 012767 000340 170740 T27: MOV #340,PS ;DISABLE ALL INTERRUPTS
1770 007036 012767 000020 002622 MOV #20,ICOUNT ;SET UP FOR 20 ITERATIONS
1771 007044 012767 007140 002612 MOV #25,ESCAPE ;SET UP TO ESCAPE TO NEXT TEST

```

Address	Offset	Hex	Dec	Label	Instruction	Comment
1772	007052	012777	004000	002540	MOV	#BIT11, 2DHSCR
1773	007060	004767	002642		JSR	PC, CLRALL
1774						:MASTER CLEAR INTERFACE
1775	007064	012777	000006	002526	MOV	#6, 2DHSCR
1776	007072	012777	011754	002526	MOV	#TBUF, 2DHBA
1777	007100	012777	177400	002522	MOV	#-400, 2DHBC
1778	007106	012777	073503	002510	MOV	#73503, 2DHLPR
1779						:SET RECEIVER BLIND
1780						:LINE SPEED = 9600 BAUD
1781	007114	012777	000100	002510	MOV	#100, 2DHBA
1782	007122	005777	002504	1S:	TST	2DHBA
1783	007126	001375			BNE	1S
1784	007130	005777	002464		TSTB	2DHSCR
1785	007134	100001			BPL	2S
1786	007136	104002			HLT	2
1787	007140	104400		2S:	SCOPE	
1788						:RECEIVER NOT BLINDED, ERROR
1789						:CHECK FOR ITERATIONS, LOOP
1790						:SET HALF DUPLEX ON LINE 7
1791						:TRANSMIT A BINARY COUNT PATTERN
1792						:VERIFY THAT NO CHARACTERS ARE RECEIVED
1793	007142	012767	000340	170626	T30: MOV	#340, PS
1794	007150	012767	000020	002510	MOV	#20, ICOUNT
1795	007156	012767	007252	002476	MOV	#25, ESCAPE
1796	007164	012777	004000	002426	MOV	#BIT11, 2DHSCR
1797	007172	004767	002530		JSR	PC, CLRALL
1798						:DISABLE ALL INTERRUPTS
1799	007176	012777	000007	002414	MOV	#7, 2DHSCR
1800	007204	012777	011754	002414	MOV	#TBUF, 2DHBA
1801	007212	012777	177400	002410	MOV	#-400, 2DHBC
1802	007220	012777	073503	002376	MOV	#73503, 2DHLPR
1803						:SET RECEIVER BLIND
1804						:LINE SPEED = 9600 BAUD
1805	007226	012777	000200	002376	MOV	#200, 2DHBA
1806	007234	005777	002372	1S:	TST	2DHBA
1807	007240	001375			BNE	1S
1808	007242	005777	002352		TSTB	2DHSCR
1809	007246	100001			BPL	2S
1810	007250	104002			HLT	2
1811	007252	104400		2S:	SCOPE	
1812						:RECEIVER NOT BLINDED, ERROR
1813						:CHECK FOR ITERATIONS, LOOP
1814						:SET HALF DUPLEX ON LINE 10
1815						:TRANSMIT A BINARY COUNT PATTERN
1816						:VERIFY THAT NO CHARACTERS ARE RECEIVED
1817	007254	012767	000340	170514	T31: MOV	#340, PS
1818	007262	012767	000020	002376	MOV	#20, ICOUNT
1819	007270	012767	007364	002364	MOV	#25, ESCAPE
1820	007276	012777	004000	002314	MOV	#BIT11, 2DHSCR
1821	007304	004767	002416		JSR	PC, CLRALL
1822						:DISABLE ALL INTERRUPTS
1823	007310	012777	000010	002302	MOV	#10, 2DHSCR
1824	007316	012777	011754	002302	MOV	#TBUF, 2DHBA
1825	007324	012777	177400	002276	MOV	#-400, 2DHBC
1826	007332	012777	073503	002264	MOV	#73503, 2DHLPR
1827						:SET RECEIVER BLIND

1829	007340	012777	000400	002264		MOV	#400,2DHBR	:CHARACTER LENGTH = 8 BITS
1830	007346	005777	002260		1\$:	TST	2DHBR	:SET BAR BIT FOR LINE 11
1831	007352	001375				BNE	1\$	:WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1832	007354	105777	002240			TSTB	2DHSCR	:WERE ANY CHARACTERS RECEIVED
1833	007360	100001				BPL	2\$	
1834	007362	104002				HLT	2	:RECEIVER NOT BLINDED, ERROR
1835	007364	104400			2\$:	SCOPE		:CHECK FOR ITERATIONS, LOOP
1836								
1837								:SET HALF DUPLEX ON LINE 11
1838								:TRANSMIT A BINARY COUNT PATTERN
1839								:VERIFY THAT NO CHARACTERS ARE RECEIVED
1840								
1841	007366	012767	000340	170402	T32:	MOV	#340,PS	:DISABLE ALL INTERRUPTS
1842	007374	012767	000020	002264		MOV	#20,COUNT	:SET UP FOR 20 ITERATIONS
1843	007402	012767	007476	002252		MOV	#2\$,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1844	007410	012777	004000	002202		MOV	#BIT11,2DHSCR	:MASTER CLEAR INTERFACE
1845	007416	004767	002304			JSR	PC,CLRALL	:CLEAR ALL BYTE COUNT AND
1846								:AND BUS ADDRESS MEMORY LOCATIONS
1847	007422	012777	000011	002170		MOV	#11,2DHSCR	:SELECT LINE 11
1848	007430	012777	011754	002170		MOV	#TBUF,2DHBA	:SET UP TO TRANSMIT
1849	007436	012777	177400	002164		MOV	#-400,2DHBC	:400 (OCTAL) CHARACTERS
1850	007444	012777	073503	002152		MOV	#73503,2DHLPR	:SET RECEIVER BLIND
1851								:LINE SPEED =9600 BAUD
1852								:CHARACTER LENGTH = 8 BITS
1853	007452	012777	001000	002152		MOV	#1000,2DHBR	:SET BAR BIT FOR LINE 11
1854	007460	005777	002146		1\$:	TST	2DHBR	:WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1855	007464	001375				BNE	1\$	
1856	007466	105777	002126			TSTB	2DHSCR	:WERE ANY CHARACTERS RECEIVED
1857	007472	100001				BPL	2\$	
1858	007474	104002				HLT	2	:RECEIVER NOT BLINDED, ERROR
1859	007476	104400			2\$:	SCOPE		:CHECK FOR ITERATIONS, LOOP
1860								
1861								:SET HALF DUPLEX ON LINE 12
1862								:TRANSMIT A BINARY COUNT PATTERN
1863								:VERIFY THAT NO CHARACTERS ARE RECEIVED
1864								
1865	007500	012767	000340	170270	T33:	MOV	#340,PS	:DISABLE ALL INTERRUPTS
1866	007506	012767	000020	002152		MOV	#20,COUNT	:SET UP FOR 20 ITERATIONS
1867	007514	012767	007610	002140		MOV	#2\$,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1868	007522	012777	004000	002070		MOV	#BIT11,2DHSCR	:MASTER CLEAR INTERFACE
1869	007530	004767	002172			JSR	PC,CLRALL	:CLEAR ALL BYTE COUNT AND
1870								:AND BUS ADDRESS MEMORY LOCATIONS
1871	007534	012777	000012	002056		MOV	#12,2DHSCR	:SELECT LINE 12
1872	007542	012777	011754	002056		MOV	#TBUF,2DHBA	:SET UP TO TRANSMIT
1873	007550	012777	177400	002052		MOV	#-400,2DHBC	:400 (OCTAL) CHARACTERS
1874	007556	012777	073503	002040		MOV	#73503,2DHLPR	:SET RECEIVER BLIND
1875								:LINE SPEED =9600 BAUD
1876								:CHARACTER LENGTH = 8 BITS
1877	007564	012777	002000	002040		MOV	#2000,2DHBR	:SET BAR BIT FOR LINE 12
1878	007572	005777	002034		1\$:	TST	2DHBR	:WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1879	007576	001375				BNE	1\$	
1880	007600	105777	002014			TSTB	2DHSCR	:WERE ANY CHARACTERS RECEIVED
1881	007604	100001				BPL	2\$	
1882	007606	104002				HLT	2	:RECEIVER NOT BLINDED, ERROR
1883	007610	104400			2\$:	SCOPE		:CHECK FOR ITERATIONS, LOOP

```

1884
1885
1886      :SET HALF DUPLEX ON LINE 13
1887      :TRANSMIT A BINARY COUNT PATTERN
1888      :VERIFY THAT NO CHARACTERS ARE RECEIVED
1889
1890      007612 012767 000340 170156 T34:  MOV      #340,PS           ;DISABLE ALL INTERRUPTS
1891      007620 012767 000020 002040      MOV      #20,ICOUNT        ;SET UP FOR 20 ITERATIONS
1892      007626 012767 007722 002026      MOV      #25,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1893      007634 012777 004000 001756      MOV      #BIT11,JDHSCR      ;MASTER CLEAR INTERFACE
1894      007642 004767 002060              JSR      PC,CLRALL        ;CLEAR ALL BYTE COUNT AND
1895      007646 012777 000013 001744      MOV      #13,JDHSCR        ;AND BUS ADDRESS MEMORY LOCATIONS
1896      007654 012777 011754 001744      MOV      #TBUF,JDHBA       ;SELECT LINE 13
1897      007662 012777 177400 001740      MOV      #-400,JDHBC      ;SET UP TO TRANSMIT
1898      007670 012777 073503 001726      MOV      #73503,JDHLPR    ;400 (OCTAL) CHARACTERS
1899
1900
1901      007676 012777 004000 001726      MOV      #4000,JDHBAR      ;SET RECEIVER BLIND
1902      007704 005777 001722 1S:      TST      JDHBAR          ;LINE SPEED = 9600 BAUD
1903      007710 001375              BNE      1S              ;CHARACTER LENGTH = 8 BITS
1904      007712 105777 001702      TSTB     JDHSCR          ;SET BAR BIT FOR LINE 13
1905      007716 100001              BPL      2S              ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1906      007720 104002              HLT      2                  ;WERE ANY CHARACTERS RECEIVED
1907      007722 104400      2S:      SCOPE                ;RECEIVER NOT BLINDED, ERROR
1908
1909
1910      :SET HALF DUPLEX ON LINE 14
1911      :TRANSMIT A BINARY COUNT PATTERN
1912      :VERIFY THAT NO CHARACTERS ARE RECEIVED
1913
1914      007724 012767 000340 170044 T35:  MOV      #340,PS           ;DISABLE ALL INTERRUPTS
1915      007732 012767 000020 001726      MOV      #20,ICOUNT        ;SET UP FOR 20 ITERATIONS
1916      007740 012767 010034 001714      MOV      #25,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST
1917      007746 012777 004000 001644      MOV      #BIT11,JDHSCR      ;MASTER CLEAR INTERFACE
1918      007754 004767 001746              JSR      PC,CLRALL        ;CLEAR ALL BYTE COUNT AND
1919
1920      007760 012777 000014 001632      MOV      #14,JDHSCR        ;AND BUS ADDRESS MEMORY LOCATIONS
1921      007766 012777 011754 001632      MOV      #TBUF,JDHBA       ;SELECT LINE 14
1922      007774 012777 177400 001626      MOV      #-400,JDHBC      ;SET UP TO TRANSMIT
1923      010002 012777 073503 001614      MOV      #73503,JDHLPR    ;400 (OCTAL) CHARACTERS
1924
1925      010010 012777 010000 001614      MOV      #10000,JDHBAR     ;SET RECEIVER BLIND
1926      010016 005777 001610 1S:      TST      JDHBAR          ;LINE SPEED = 9600 BAUD
1927      010022 001375              BNE      1S              ;CHARACTER LENGTH = 8 BITS
1928      010024 105777 001570      TSTB     JDHSCR          ;SET BAR BIT FOR LINE 14
1929      010030 100001              BPL      2S              ;WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1930      010032 104002              HLT      2                  ;WERE ANY CHARACTERS RECEIVED
1931      010034 104400      2S:      SCOPE                ;RECEIVER NOT BLINDED, ERROR
1932
1933
1934      :SET HALF DUPLEX ON LINE 15
1935      :TRANSMIT A BINARY COUNT PATTERN
1936      :VERIFY THAT NO CHARACTERS ARE RECEIVED
1937
1938      010036 012767 000340 167732 T36:  MOV      #340,PS           ;DISABLE ALL INTERRUPTS
1939      010044 012767 000020 001614      MOV      #20,ICOUNT        ;SET UP FOR 20 ITERATIONS
1940      010052 012767 010146 001602      MOV      #25,ESCAPE        ;SET UP TO ESCAPE TO NEXT TEST

```

F04

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 42
 DZDHI.C.P11 29-SEP-76 14:42

SEG 004:

1940	010060	012777	004000	001532		MOV	#BIT11,2DHSCR	:MASTER CLEAR INTERFACE
1941	010066	004767	001634			JSR	PC,CLRALL	:CLEAR ALL BYTE COUNT AND
1942								:AND BUS ADDRESS MEMORY LOCATIONS
1943	010072	012777	000015	001520		MOV	#15,2DHSCR	:SELECT LINE 15
1944	010100	012777	011754	001520		MOV	#TBUF,2DHBA	:SET UP TO TRANSMIT
1945	010106	012777	177400	001514		MOV	#-400,2DHBC	:400 (OCTAL) CHARACTERS
1946	010114	012777	073503	001502		MOV	#73503,2DHLPR	:SET RECEIVER BLIND
1947								:LINE SPEED =9600 BAUD
1948								:CHARACTER LENGTH = 8 BITS
1949	010122	012777	020000	001502		MOV	#20000,2DHBAR	:SET BAR BIT FOR LINE 15
1950	010130	005777	001476		1S:	TST	2DHBAR	:WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1951	010134	001375				BNE	1S	
1952	010136	105777	001456			TSTB	2DHSCR	:WERE ANY CHARACTERS RECEIVED
1953	010142	100001				BPL	2S	
1954	010144	104002				HLT	2	:RECEIVER NOT BLINDED, ERROR
1955	010146	104400			2S:	SCOPE		:CHECK FOR ITERATIONS, LOOP
1956								
1957								:SET HALF DUPLEX ON LINE 16
1958								:TRANSMIT A BINARY COUNT PATTERN
1959								:VERIFY THAT NO CHARACTERS ARE RECEIVED
1960								
1961	010150	012767	000340	167620	T37:	MOV	#340,PS	:DISABLE ALL INTERRUPTS
1962	010156	012767	000020	001502		MOV	#20,ICOUNT	:SET UP FOR 20 ITERATIONS
1963	010164	012767	010260	001470		MOV	#25,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1964	010172	012777	004000	001420		MOV	#BIT11,2DHSCR	:MASTER CLEAR INTERFACE
1965	010200	004767	001522			JSR	PC,CLRALL	:CLEAR ALL BYTE COUNT AND
1966								:AND BUS ADDRESS MEMORY LOCATIONS
1967	010204	012777	000016	001406		MOV	#16,2DHSCR	:SELECT LINE 16
1968	010212	012777	011754	001406		MOV	#TBUF,2DHBA	:SET UP TO TRANSMIT
1969	010220	012777	177400	001402		MOV	#-400,2DHBC	:400 (OCTAL) CHARACTERS
1970	010226	012777	073503	001370		MOV	#73503,2DHLPR	:SET RECEIVER BLIND
1971								:LINE SPEED =9600 BAUD
1972								:CHARACTER LENGTH = 8 BITS
1973	010234	012777	040000	001370		MOV	#40000,2DHBAR	:SET BAR BIT FOR LINE 16
1974	010242	005777	001354		1S:	TST	2C-BAR	:WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
1975	010246	001375				BNE	1S	
1976	010250	105777	001344			TSTB	2DHSCR	:WERE ANY CHARACTERS RECEIVED
1977	010254	100001				BPL	2S	
1978	010256	104002				HLT	2	:RECEIVER NOT BLINDED, ERROR
1979	010260	104400			2S:	SCOPE		:CHECK FOR ITERATIONS, LOOP
1980								
1981								:SET HALF DUPLEX ON LINE 17
1982								:TRANSMIT A BINARY COUNT PATTERN
1983								:VERIFY THAT NO CHARACTERS ARE RECEIVED
1984								
1985	010262	012767	000340	167506	T40:	MOV	#340,PS	:DISABLE ALL INTERRUPTS
1986	010270	012767	000020	001370		MOV	#20,ICOUNT	:SET UP FOR 20 ITERATIONS
1987	010276	012767	010372	001356		MOV	#25,ESCAPE	:SET UP TO ESCAPE TO NEXT TEST
1988	010304	012777	004000	001306		MOV	#BIT11,2DHSCR	:MASTER CLEAR INTERFACE
1989	010312	004767	001410			JSR	PC,CLRALL	:CLEAR ALL BYTE COUNT AND
1990								:AND BUS ADDRESS MEMORY LOCATIONS
1991	010316	012777	000017	001274		MOV	#17,2DHSCR	:SELECT LINE 17
1992	010324	012777	011754	001274		MOV	#TBUF,2DHBA	:SET UP TO TRANSMIT
1993	010332	012777	177400	001270		MOV	#-400,2DHBC	:400 (OCTAL) CHARACTERS
1994	010340	012777	073503	001256		MOV	#73503,2DHLPR	:SET RECEIVER BLIND
1995								:LINE SPEED =9600 BAUD

G04

DZDH1 MACY11 27.1006 29-SEP-76 14:44 PAGE 43
 DZDHIC.F11 29-SEP-76 14:42

SEG 0042

1996						
1997	010346	012777	100000	001256		MOV
1998	010354	005777	001252		15:	TST
1999	010360	001375				BNE
2000	010362	105	001232			TSTB
2001	010366	100001				BPL
2002	010370	104002				HLT
2003	010372	104003			25:	SCOPE

```

: CHARACTER LENGTH = 8 BITS
: SET BAR BIT FOR LINE 17
: WAIT FOR ALL CHARACTERS TO BE TRANSMITTED
: WERE ANY CHARACTERS RECEIVED
: RECEIVER NOT BLINDED, ERROR
: CHECK FOR ITERATIONS, LOOP
  
```

CZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 44
CZDHI.C.P11 29-SEP-76 14:42

CZDHIC.P11 29-SEP-76 14:42

Address	Hex	Label	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	
---------	-----	-------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--

DZDHI MACY11 27.1006: 29-SEP-76 14:44 PAGE 45
 DZDHI.C.P11 29-SEP-76 14:42

SEG 0044

```

2055
2056                                     :ERROR HANDLER
2057
2058 010564 032767 020000 166776 ERRORS: BIT      #SW13,SWR
2059 010572 001051          BNE      HALTS
2060 010574 021667 001116          CMP      (SP),LAST
2061 010600 001404          BEQ      1$
2062 010602 011667 001110          MOV      (SP),LAST
2063 010606 005067 001040          CLR      ERRFLG
2064 010612 104406          1$: SAVOSP
2065 010614 011605          MOV      (SP),R5
2066 010616 162705 000002          SUB      #2,R5
2067 010622 011504          MOV      (R5),R4
2068 010624 006304          ASL      R4
2069 010626 006304          ASL      R4
2070 010630 042704 177001          BIC      #177001,R4
2071 010634 062704 013152          ADD      #ERRTAB,R4
2072 010640 012467 000034          MOV      (R4)+,ERRMSG
2073 010644 011467 000042          MOV      (R4),DATABP
2074 010650 005767 000776          TST      ERRFLG
2075 010654 001403          BEQ      TYPMSG
2076 010656 005767 000030          TST      DATABP
2077 010662 001007          BNE      TYPDAT
2078 010664 104402          TYPMSG: OCTASC
2079 010666 010760          ERTABO
2080 010670 012767 000001 000754          MOV      #1,ERRFLG
2081 010676 104401          TYPE
2082 010700 000000          ERRMSG: 0
2083 010702 005767 000004          TYPDAT: TST      DATABP
2084 010706 001402          BEQ      RESREG
2085 010710 104402          OCTASC
2086 010712 000000          DATABP: 0
2087 010714 104407          RESREG: RESOS
2088 010716 005767 166646          HALTS: TST      SWR
2089 010722 100005          BPL      EXITER
2090 010724 010046          PUSHRO
2091 010726 016600 000002          MOV      2(SP),R0
2092 010732 000000          HALT
2093 010734 012600          POPRO
2094 010736 005267 000714          EXITER: INC      ERRCNT
2095 010742 032767 002000 166620          BIT      #SW10,SWR
2096 010750 001402          BEQ      1$
2097 010752 016716 000704          MOV      ESCAPE,(SP)
2098 010756 000002          1$: RTI
2099 010760 000001          ERTABO: 1
2100 010762 006          .BYTE 6,2
2101 010764 011710          SAVPC

```

J04

CZDH1 MACY11 27.10061 29-SEP-76 14:44 PAGE 46
 CZDH1C.P11 29-SEP-76 14:42

SEG 0045

```

2102      ;TRAP DISPATCH SERVICE
2103      ;ARGUMENT OF TRAP IS EXTRACTED
2104      ;AND USED AS OFFSET TO OBTAIN POINTER
2105      ;TO SELECTED SUBROUTINE
2106
2107      010766 011646      TRPSRV: MOV      (SP), -(SP)      ;GET PC OF RETURN
2108      010770 162716 000002      SUB      #2, (SP)      ;PC OF TRAP
2109      010774 017616 000000      MOV      2(SP), (SP)      ;GET TRP
2110      011000 006316      TRPOK: ASL      (SP)      ;MULTIPLY TRAP ARG BY 2
2111      011002 042716 177001      BIC      #177001, (SP)      ;CLEAR UNWANTED BITS
2112      011006 062716 013072      ADD      #TRPTAB, (SP)      ;POINTER TO SUBROUTINE ADDRESS
2113      011012 017616 000000      MOV      2(SP), (SP)      ;SUBROUTINE ADDRESS
2114      011016 000136      JMP      2(SP)+      ;GO TO SUBROUTINE
2115
2116      ;SAVE PC OF TEST THAT FAILED AND RD-R5
2117
2118      011020 016667 000004 000662 SVOSP: MOV      4(SP), SAVPC
2119
2120      ;SAVE RD-R5
2121
2122      011026 010567 000652      SVOS:  MOV      R5, SAVR5
2123      011032 010467 000644      MOV      R4, SAVR4
2124      011036 010367 000636      MOV      R3, SAVR3
2125      011042 010267 000630      MOV      R2, SAVR2
2126      011046 010167 000622      MOV      R1, SAVR1
2127      011052 010067 000614      MOV      R0, SAVR0
2128      011056 000002      RTI
2129      ;RESTORE RD-R5
2130
2131      011060 016700 000606      RSOS:  MOV      SAVR0, R0
2132      011064 016701 000604      MOV      SAVR1, R1
2133      011070 016702 000602      MOV      SAVR2, R2
2134      011074 016703 000600      MOV      SAVR3, R3
2135      011100 016704 000576      MOV      SAVR4, R4
2136      011104 016705 000574      MOV      SAVR5, R5
2137      011110 000002      RTI

```

K04

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 47
 DZDHI.C.P11 29-SEP-76 14:42

SEQ 0046

```

2138
2139
2140
2141 011112 017605 000000
2142 011116 062716 000002
2143 011122 105777 000466
2144 011126 100375
2145 011130 105715
2146 011132 001001
2147 011134 000002
2148 011136 112577 000454
2149 011142 000767
2150
2151
2152
2153
2154 011144 017667 000000 000006
2155 011152 062716 000002
2156 011156 104401
2157 011160 000000
2158 011162 012704 013114
2159 011166 012703 000007
2160 011172 105777 000412
2161 011176 100375
2162 011200 117714 000406
2163 011204 142714 000200
2164 011210 122427 000015
2165 011214 001413
2166 011216 117777 000370 000372
2167 011224 105777 000364
2168 011230 100375
2169 011232 005303
2170 011234 001356
2171 011236 104401
2172 011240 012746
2173 011242 000745
2174 011244 000002

```

TELETYPE OUTPUT ROUTINE

```

TYPED: MOV 2(SP),R5
ADD #2,(SP)
15: TSTB 2TPCSR
BPL 15
TSTB (R5)
BNE 25
RTI
25: MOVB (R5)+,2TPDBR
BR 15

```

ASCII STRING INPUT ROUTINE

```

INSTRG: MOV 2(SP),MSG
ADD #2,(SP)
INSTR1: TYPE
MSG: 0
MOV #INBUF,R4
MOV #7,R3
15: TSTB 2TKCSR
BPL 15
MOVB 2TKDBR,(R4)
BICB #200,(R4)
CMPB (R4)+,#15
BEQ INSTR2
MOVB 2TKDBR,2TPDBR
25: TSTB 2TPCSR
BPL 25
DEC R3
BNE 15
INSTR2: TYPE
MQM
BR INSTR1
INSTR2: RTI

```

```

2174
2175                                     ;CONVERT ASCII STRING TO OCTAL
2176
2177 011246 011605
2178 011250 012567 000146
2179 011254 012567 000144
2180 011260 012567 000142
2181 011264 112567 000140
2182 011270 112567 000135
2183 011274 010516
2184 011276 005005
2185 011300 012704 013114
2186 011304 122714 000015
2187 011310 001420
2188 011312 121427 000060
2189 011316 002415
2190 011320 121427 000067
2191 011324 003012
2192 011326 142714 000060
2193 011332 152405
2194 011334 122714 000015
2195 011340 001406
2196 011342 006305
2197 011344 006305
2198 011346 006305
2199 011350 000760
2200 011352 104404
2201 011354 000750
2202
2203                                     ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
2204
2205 011356 020567 000042
2206 011362 101373
2207 011364 020567 000032
2208 011370 103770
2209 011372 136705 000032
2210 011376 001365
2211
2212                                     ;STORE NUMBER AT SPECIFIED ADDRESS
2213
2214 011400 016704 000022
2215 011404 010524
2216 011406 062705 000002
2217 011412 105367 000013
2218 011416 001372
2219 011420 000002
2220 011422 000000
2221 011424 000000
2222 011426 000000
2223 011430 000000
2224 011431

```

```

PARPMS: MOV (SP),R5
        MOV (R5)+,LOLIM
        MOV (R5)+,HILIM
        MOV (R5)+,DEVADR
        MOV (R5)+,LOBITS
        MOV (R5)+,ADRCNT
        MOV R5,(SP)
PARAM1: CLR R5
        MOV #INBUF,R4
        CMPB #15,(R4)
        BEQ PARERR
1$:      CMPB (R4),#60
        BLT PARERR
        CMPB (R4),#67
        BGT PARERR
        BICB #60,(R4)
        BISB (R4)+,R5
        CMPB #15,(R4)
        BEQ LIMITS
        ASL R5
        ASL R5
        ASL R5
        BR 1$
PARERR: INSTER
        BR PARAM1

```

```

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

```

```

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

```

M04

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 49
 DZDHI.C.P11 29-SEP-76 14:42

SEG 0048

```

2225
2226                                     ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
2227
2228 011432 104401 OCTASN: TYPE
2229 011434 012752 MCRLF
2230 011436 017601 000000 MOV 2(SP),R1
2231 011442 062716 000002 ADD #2,(SP)
2232 011446 012167 000130 MOV (R1)+,WRDCNT
2233 011452 112167 000126 1$: MOV (R1)+,CHRCNT
2234 011456 112167 000123 MOV (R1)+,SPACNT
2235 011462 013167 000120 MOV 2(R1)+,BINWRD
2236 011466 016704 000114 2$: MOV BINWRD,R4
2237 011472 116705 000106 MOV CHRCNT,R5
2238 011476 012700 013126 MOV #TEMP,R0
2239 011502 010403 3$: MOV R4,R3
2240 011504 042703 177770 BIC #177770,R3
2241 011510 062703 000260 ADD #260,R3
2242 011514 110320 MOV R3,(R0)+
2243 011516 006204 ASR R4
2244 011520 006204 ASR R4
2245 011522 006204 ASR R4
2246 011524 005305 DEC R5
2247 011526 001365 BNE 3$
2248 011530 012703 013140 MOV #MDATA,R3
2249 011534 114023 4$: MOV (R0),(R3)+
2250 011536 105367 000042 DECB CHRCNT
2251 011542 001374 BNE 4$
2252 011544 105767 000035 TSTB SPACNT
2253 011550 001405 BEQ 6$
2254 011552 112723 000240 5$: MOV #240,(R3)+
2255 011556 105367 000023 DECB SPACNT
2256 011562 001373 BNE 5$
2257 011564 105013 6$: CLRB (R3)
2258 011566 104401 TYPE
2259 011570 013140 MDATA
2260 011572 005367 000004 DEC WRDCNT
2261 011576 001325 BNE 1$
2262 011600 000002 RTI
2263 011602 000000 WRDCNT: 0
2264 011604 000000 CHRCNT: 0
2265 011605 011605 SPACNT=CHRCNT+1
2266 011606 000000 BINWRD: 0

```

N04

DZDHI MACY11 27:006 29-SEP-76 14:44 PAGE 50
DZDHI.C.P11 29-SEP-76 14:42

SEQ 0049

```

2267                                     :INDIRECT POINTERS
2268
2269 011610 177560 TKCSR: 177560
2270 011612 177562 TKDBR: 177562
2271 011614 177564 TPCSR: 177564
2272 011616 177566 TPDBR: 177566
2273 011620 000000 DHSCR: 0
2274 011622 000000 DHNRC: 0
2275 011624 000000 DHLPR: 0
2276 011626 000000 DHBA: 0
2277 011630 000000 DHBC: 0
2278 011632 000000 DHBAR: 0
2279 011634 000000 DHBCR: 0
2280 011636 000000 DHSSR: 0
2281 011640 000000 DHSLR: 0
2282 011642 000000 DHRVEC: 0
2283 011644 000000 DHRLVL: 0
2284 011646 000000 DHTVEC: 0
2285 011650 000000 DHTLVL: 0
2286                                     :PROGRAM VARIABLES
2287
2288 011652 000000 ERRFLG: 0 ;ERROR FLAG
2289 011654 000000 PASCNT: 0 ;PASS COUNT
2290 011656 000000 ERRCNT: 0 ;ERROR COUNT
2291 011660 000000 RETURN: 0 ;SCOPE RETURN ADDRESS FOR TEST LOOPING
2292 011662 000000 ESCAPE: 0 ;ADDRESS FOR ERROR ESCAPE
2293 011664 000000 FREEZ1: 0 ;DATA LOOPING RETURN ADDRESS
2294 011666 000000 ICOUNT: 0 ;ITERATION COUNT FOR TEST IN PROGRESS
2295 011670 000000 LPCNT: 0 ;NUMBER OF ITERATIONS THIS TEST
2296 011672 000000 SAVR0: 0 ;R0 SAVE AREA
2297 011674 000000 SAVR1: 0 ;R1 SAVE AREA
2298 011676 000000 SAVR2: 0 ;R2 SAVE AREA
2299 011700 000000 SAVR3: 0 ;R3 SAVE ARE
2300 011702 000000 SAVR4: 0 ;R4 SAVE AREA
2301 011704 000000 SAVR5: 0 ;R5 SAVE AREA
2302 011706 000000 SAVSP: 0 ;STACK POINTER SAVE AREA
2303 011710 000000 SAVPC: 0 ;CALLING ROUTINE SAVT AREA
2304 011712 000000 INIFLG: 0 ;PROGRAM INITIALIZATION FLAG
2305 011714 000000 STFLG: 0 ;PROGRAM START FLAG
2306 011716 000000 LAST: 0 ;LAST ERROR PC
2307 011720 000000 ENDFLG: 0
2308 011722 000000 NULL: 0
2309 011724 000000 LINACT: 0

```

```

CLRALL:  MOV      R20,R0
15:      CLR      R20H
        CLR      R20H
        INC      R20H
        DEC      R20
        BNE     R20,R0
        RTS
TRUE:    .BYTE    0

```

[illegible]

DDMI MACY11 2711006 29-SEP-76 14:44 PAGE 54
DDMIC.P11 29-SEP-76 14:42

SEC 0052

012212	.BYTE	TCAT
012213	.BYTE	TCAT
012214	.BYTE	TCAT
012215	.BYTE	TCAT
012216	.BYTE	TCAT
012217	.BYTE	TCAT
012220	.BYTE	TCAT
012221	.BYTE	TCAT
012222	.BYTE	TCAT
012223	.BYTE	TCAT
012224	.BYTE	TCAT
012225	.BYTE	TCAT
012226	.BYTE	TCAT
012227	.BYTE	TCAT
012228	.BYTE	TCAT
012229	.BYTE	TCAT
012230	.BYTE	TCAT
012231	.BYTE	TCAT
012232	.BYTE	TCAT
012233	.BYTE	TCAT
012234	.BYTE	TCAT
012235	.BYTE	TCAT
012236	.BYTE	TCAT
012237	.BYTE	TCAT
012240	.BYTE	TCAT
012241	.BYTE	TCAT
012242	.BYTE	TCAT
012243	.BYTE	TCAT
012244	.BYTE	TCAT
012245	.BYTE	TCAT
012246	.BYTE	TCAT
012247	.BYTE	TCAT
012248	.BYTE	TCAT
012249	.BYTE	TCAT
012250	.BYTE	TCAT
012251	.BYTE	TCAT
012252	.BYTE	TCAT
012253	.BYTE	TCAT
012254	.BYTE	TCAT
012255	.BYTE	TCAT
012256	.BYTE	TCAT
012257	.BYTE	TCAT
012260	.BYTE	TCAT
012261	.BYTE	TCAT
012262	.BYTE	TCAT
012263	.BYTE	TCAT
012264	.BYTE	TCAT
012265	.BYTE	TCAT
012266	.BYTE	TCAT
012267	.BYTE	TCAT
012270	.BYTE	TCAT
012271	.BYTE	TCAT
012272	.BYTE	TCAT
012273	.BYTE	TCAT
012274	.BYTE	TCAT
012275	.BYTE	TCAT
012276	.BYTE	TCAT
012277	.BYTE	TCAT
012300	.BYTE	TCAT
012301	.BYTE	TCAT

H05

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 57
 DZDHI0.P11 29-SEP-76 14:42

SEG 0056

```

2595                                     :ENTER HERE ON POWER FAILURE
2596
2597
2598 012456 010046          PFAIL: MOV      R0,-(SP)          :SAVE R0-R5 ON PROCESSOR STACK
2599 012460 010146          MOV      R1,-(SP)
2600 012462 010246          MOV      R2,-(SP)
2601 012464 010346          MOV      R3,-(SP)
2602 012466 010446          MOV      R4,-(SP)
2603 012470 010546          MOV      R5,-(SP)
2604 012472 016746 165326  MOV      24,-(SP)
2605 012476 010667 177204  MOV      SP,SAVSP          :SAVE STACK POINTER
2606 012502 012767 012514 165314  MOV      *RESTART,24          :SET UP FOR POWER UP TRAP
2607 012510 000000          HALT                          :HALT ON POWER DOWN NORMAL
2608 012512 000777          BR
2609
2610                                     :PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
2611
2612
2613 012514 016706 177166  RESTAR: MOV      SAVSP,SP          :RESTORE STACK POINTER
2614 012520 012605          MOV      (SP)+,R5          :RESTORE R0-R5
2615 012522 012604          MOV      (SP)+,R4
2616 012524 012603          MOV      (SP)+,R3
2617 012526 012602          MOV      (SP)+,R2
2618 012530 012601          MOV      (SP)+,R1
2619 012532 012600          MOV      (SP)+,R0
2620 012534 012767 012456 165262  MOV      *PFAIL,24          :SET UP FOR POWER FAILURE
2621 012542 012767 000340 165226  MOV      *340,PS
2622 012550 012706 013540          MOV      *STACK,SP
2623 012554 005067 000346          CLR      TEMP
2624 012560 005267 000342          INC      TEMP
2625 012564 001375          BNE      .-4
2626 012566 104402          OCTASC
2627 012570 012612          PFTAB
2628 012572 104401          TYPE
2629 012574 012755          MPFAIL
2630 012576 005067 177050          CLR      ERRFLG
2631 012602 005067 177110          CLR      LAST
2632 012606 000177 177046          JMP      *RETURN
2633 012612 000001          PFTAB: 1
2634 012614 000006 000002          6,2
2635 012620 000207          RETURN

```

02041 MACY11 27:006 29-SEP-76 14:44 PAGE 58
 020410.P11 29-SEP-76 14:42

SEQ 0057

```

2635 012622 005015 042012 030510 MTITLE: .ASCIZ <15><12><12>/DH11 HALF-DUPLEX AND BREAK TEST /<15><12>/
2636 012630 020061 040510 043114
2637 012636 042055 050125 042514
2638 012644 020130 047101 020104
2639 012652 051102 040505 020113
2640 012660 042524 052123 006440
2641 012666 000012
2642 012670 005015 042526 052103 MVECT0: .ASCIZ <15><12>/VECTOR ADDRESS-/
2643 012676 051117 040440 042104
2644 012704 042522 051523 000055
2645 012712 005015 047503 052116 MREGAD: .ASCIZ <15><12>/CONTROL REGISTER ADDRESS-/
2646 012720 047522 020114 042522
2647 012726 044507 052123 051105
2648 012734 040440 042104 042522
2649 012742 051523 000055
2650 012746 020040 000077
2651 012752 005015 000 MQM: .ASCIZ / ?/
2652 012755 040 050040 053517 MCRLF: .ASCIZ <15><12>/
2653 012762 051105 043040 044501 MPFAIL: .ASCIZ / POWER FAILURE, PROGRAM RESTART AT TEST IN PROGRESS/
2654 012770 052514 042522 020054
2655 012776 051120 043517 040522
2656 013004 020115 042522 052123
2657 013012 051101 020124 052101
2658 013020 052040 051505 020124
2659 013026 047111 050040 047522
2660 013034 051107 051505 000123
2661 013042 005015 055104 044104 MEPASS: .ASCIZ <15><12>/DZDHI/
2662 013050 000111
2663 013052 005015 000122 MR: .ASCIZ <15><12>/R/
2664 013056 005015 042524 052123 MTSTPC: .ASCIZ <15><12>/TEST PC-/
2665 013064 050040 026503 000
2666 013072 .EVEN
2667
2668 ;TABLE OF POINTERS FOR TRAP DECODING
2669
2670 TRPTAB: SCOPER
2671 013072 010446 TYPER
2672 013074 011112 OCTASN
2673 013076 011432 INSTRG
2674 013100 011144 INSTRG
2675 013102 011236 INSTRG
2676 013104 011246 INSTRG
2677 013106 011020 INSTRG
2678 013110 011060 INSTRG
2679 013112 010546 INSTRG
2680
2681 ;BUFFERS FOR INPUT-OUTPUT
2682
2683 INBUF: 0
2684 013114 000000 .=.+10
2685 013126 000000 TEMP: 0
2686 013126 000000 .=.+10
2687 013140 000000 MDATA: 0
2688 013152 .=.+10
2689
2690 ;TABLE OF POINTERS TO ERROR MESSAGES AND DATA

```

J05

CZDHI MACY11 27.1006 29-SEP-76 14:44 PAGE 59
CZDHI.C.P11 29-SEP-76 14:42

SEQ 0058

2691	C13152				ERRTAB:			
2692	013152	013166				EM1		
2693	013154	013326				DT1		
2694	013156	013242				EM2		
2695	013160	013326				DT1		
2696	013162	013300				EM3		
2697	013164	000000				0		
2698	013166	047515	042522	052040	EM1:	.ASCIZ	/MORE THAN 1 CHARACTER RECEIVED/	<15><12>/EXP REC/
2699	013174	040510	020116	020061				
2700	013202	044103	051101	041501				
2701	013210	042524	020122	042522				
2702	013216	042503	053111	042105				
2703	013224	005015	054105	020120				
2704	013232	020040	020040	042522				
2705	013240	000103						
2706	013242	051102	040505	020113	EM2:	.ASCIZ	/BREAK DATA ERROR/	<15><12>/EXP REC/
2707	013250	040504	040524	042440				
2708	013256	051122	051117	005015				
2709	013264	054105	020120	020040				
2710	013272	020040	042522	000103				
2711	013300	042522	042503	053111	EM3:	.ASCIZ	/RECEIVER NOT BLINDED/	
2712	013306	051105	047040	052117				
2713	013314	041040	044514	042116				
2714	013322	042105	000					
2715		013326				.EVEN		
2716	013326	000002			DT1:	2		
2717	013330	006	002		.BYTE	6,2		
2718	013332	011704				SAVR5		
2719	013334	006	002		.BYTE	6,2		
2720	013336	011702				SAVR4		
2721	013340	000000			ENDCOD:	0		
2722		000001			.END			

K05

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 61
 DZDHI.C.P11 29-SEP-76 14:42

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0059

ADRCNT = 011431	2182*	2217*	2224*											
BEGIN = 001202	877	906	912*	2025										
BITWARD = 011606	2235*	2236	2266*											
BITX = 000000	932*	975*	1018*	1061*	1104*	1147*	1190*	1233*	1276*	1319*	1362*	1405*	1448*	
	1491*	1534*	1577*	1620*	1644*	1668*	1692*	1716*	1740*	1764*	1788*	1812*	1836*	
	1860*	1884*	1908*	1932*	1956*	1980*	2004*							
BIT00 = 000001	563*													
BIT01 = 000002	562*													
BIT02 = 000004	561*													
BIT03 = 000010	560*													
BIT04 = 000020	559*													
BIT05 = 000040	558*													
BIT06 = 000100	557*													
BIT07 = 000200	556*													
BIT08 = 000400	555*													
BIT09 = 001000	554*													
BIT10 = 002000	553*													
BIT11 = 004000	552*	943	954	986	997	1029	1040	1072	1083	1115	1126	1158	1169	
	1201	1212	1244	1255	1287	1298	1330	1341	1373	1384	1416	1427	1459	
	1470	1502	1513	1545	1556	1588	1599	1628	1652	1676	1700	1724	1748	
	1772	1796	1820	1844	1868	1892	1916	1940	1964	1988				
BIT12 = 010000	551*													
BIT13 = 020000	550*													
BIT14 = 040000	549*													
BIT15 = 100000	548*													
CHARCNT = 011604	2233*	2237	2250*	2264*	2265									
CHARALL = 011726	944	987	1030	1073	1116	1159	1202	1245	1298	1321	1374	1417	1460	
	1503	1546	1589	1629	1653	1677	1701	1725	1749	1773	1797	1821	1845	
	1869	1893	1917	1941	1965	1989	2313*							
DATA = 127400	2579*													
DATABP = 010712	2073*	2076	2083	2086*										
DATAX = 127400	2579*	2580*	2581*	2582*	2583*	2584*	2585*	2586*	2587*	2588*	2589*	2590*	2591*	
	2592*	2593*	2594*											
DEVADR = 011426	2180*	2214	2222*											
DMBA = 011626	947*	956*	990*	999*	1033*	1042*	1076*	1085*	1119*	1128*	1162*	1171*	1205*	
	1214*	1248*	1257*	1291*	1300*	1334*	1343*	1377*	1386*	1420*	1429*	1463*	1472*	
	1506*	1515*	1549*	1558*	1592*	1601*	1632*	1656*	1680*	1704*	1728*	1752*	1776*	
	1800*	1824*	1848*	1872*	1896*	1920*	1944*	1968*	1992*	2276*	2314*			
DMBAR = 011632	951*	960*	961	994*	1003*	1004	1037*	1046*	1047	1080*	1089*	1090	1123*	
	1132*	1133	1166*	1175*	1176	1209*	1218*	1219	1252*	1261*	1262	1295*	1304*	
	1305	1338*	1347*	1348	1381*	1390*	1391	1424*	1433*	1434	1467*	1476*	1477	
	1510*	1519*	1520	1553*	1562*	1563	1596*	1605*	1606	1637*	1638	1661*	1662	
	1685*	1686	1709*	1710	1733*	1734	1757*	1758	1781*	1782	1805*	1806	1829*	
	1830	1853*	1854	1877*	1878	1901*	1902	1925*	1926	1949*	1950	1973*	1974	
	1997*	1998	2278*											
DMBC = 011630	948*	957*	991*	1000*	1034*	1043*	1077*	1086*	1120*	1129*	1163*	1172*	1206*	
	1215*	1249*	1258*	1292*	1301*	1335*	1344*	1378*	1387*	1421*	1430*	1464*	1473*	
	1507*	1516*	1550*	1559*	1593*	1602*	1633*	1657*	1681*	1705*	1729*	1753*	1777*	
	1801*	1825*	1849*	1873*	1897*	1921*	1945*	1969*	1993*	2277*	2315*			
DMBCR = 011634	959*	1002*	1045*	1088*	1131*	1174*	1217*	1260*	1303*	1346*	1389*	1432*	1475*	
	1518*	1561*	1604*	2279*										
DMBCR = 011624	949*	958*	992*	1001*	1035*	1044*	1078*	1087*	1121*	1130*	1164*	1173*	1207*	
	1216*	1250*	1259*	1293*	1302*	1336*	1345*	1379*	1388*	1422*	1431*	1465*	1474*	
	1508*	1517*	1551*	1560*	1594*	1603*	1634*	1658*	1682*	1706*	1730*	1754*	1778*	
	1802*	1826*	1850*	1874*	1898*	1922*	1946*	1970*	1994*	2275*				
DMBCR = 011622	969	1012	1055	1098	1141	1184	1227	1270	1313	1356	1399	1442	1485	

DZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 63
DZDHIC.P11 29-SEP-76 14:42 CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

N05

0ZDHI MACY11 27(1006) 29-SEP-76 14:44 PAGE 64

SEQ 0052

OZDHIC.P11 29-SEP-76 14:42 CROSS REFERENCE TABLE -- USER SYMBOLS

RWRD15	012450	1529	1531	2592#															
RWRD16	012452	1572	1574	2593#															
RWRD17	012454	1615	1617	2594#															
RWRD2	012422	1056	1058	2581#															
RWRD3	012424	1099	1101	2582#															
RWRD4	012426	1142	1144	2583#															
RWRD5	012430	1185	1187	2584#															
RWRD6	012432	1228	1230	2585#															
RWRD7	012434	1271	1273	2586#															
SAVPC	011710	2101	2118*	2303#															
SAVR0	011672	2127*	2131	2296#															
SAVR1	011674	2126*	2132	2297#															
SAVR2	011676	2125*	2133	2298#															
SAVR3	011700	2124*	2134	2299#															
SAVR4	011702	2123*	2135	2300#	2720														
SAVR5	011704	2122*	2136	2301#	2718														
SAVSP	011706	2302#	2605*	2612															
SAVOSP=	104406	847#	2064																
SCOPE =	104400	841#	974	1017	1060	1103	1146	1189	1232	1275	1318	1361	1404	1447					
		1490	1533	1576	1619	1643	1667	1691	1715	1739	1763	1787	1811	1835					
		1859	1883	1907	1931	1955	1979	2003											
SCOPE1	010446	2030#	2670																
SCOPE1=	104410	849#																	
SCOPE1R	010546	2051#	2678																
SPACNT=	011605	2234*	2252	2255*	2265#														
STACK =	013540	535#	864	913	2621														
START	001000	833	863#																
STFLG	011714	866*	926	928*	2305#														
SV05	011026	2122#																	
SV05P	011020	2118#	2676																
SWR =	177570	532#	876	914	2030	2032	2034	2051	2058	2088	2095								
SW00 =	000001	515#	876																
SW01 =	000002	514#	914																
SW02 =	000004	513#																	
SW03 =	000010	512#																	
SW04 =	000020	511#																	
SW05 =	000040	510#																	

C06

03DH: MACY: 27 1006 29-SEP-76 14:44 PAGE 66

29-SEP-76 14:42

CROSS REFERENCE TABLE -- USER SYMBOLS

SEC 3054

[illegible]

D06

DZDHIC.P11 29-SEP-76 14:44 PAGE 68

SEQ 0055

DZDHIC.P11 29-SEP-76 14:42 CROSS REFERENCE TABLE -- MACRO NAMES

BALNO:	9218	1620	1644	1668	1692	1716	1740	1764	1788	1812	1836	1860	1884	1908	1932
BREAK:	9318	9320	975	1018	1061	1104	1147	1190	1233	1276	1319	1362	1405	1448	1491
CCRAL	9328	2311													
CODEM1	8548	9033													
4	5458	968	973	1011	1016	1054	1059	1097	1102	1140	1145	1183	1188	1226	1231
	1269	1274	1312	1317	1355	1360	1398	1403	1441	1446	1484	1489	1527	1532	1570
	1575	1613	1618	1642	1666	1690	1714	1738	1762	1786	1810	1834	1858	1882	1906
	1930	1954	1978	2002											
USE	9318														
APDET		841	842	843	844	845	846	847	848	849					
5		940	983	1026	1069	1112	1155	1198	1241	1284	1327	1370	1413	1456	1499
	154	1585	1628	1649	1673	1697	1721	1745	1769	1793	1817	1841	1865	1889	1913
	1937	1961	1985												
WORDS	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591	2592	2593
	2594														
BUFFE		2679													
CHANCE		564													
ENMOCC		2721													
ERROR		2102													
ERRORS		2155													
ERRORS		2688													
ERRORS		479													
ERRORS		2153													
ERRORS		2635													
ERRORS		2225													
ERRORS		2174													
ERRORS		2295													
ERRORS		2267													
ERRORS		2129													
ERRORS		2115													
ERRORS		2126													
ERRORS		2140													
ERRORS		822													
ERRORS		854													
ERRORS		498													
ERRORS		836													
ERRORS		2102													
ERRORS		2667													
ERRORS		2138													
ERRORS		2286													

. ABS. 013342 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DZDHIC.BIN,DZDHIC.SEG/CRF/SOL/DOC=UTIL2.P11,DZDHIC.P11
RUN-TIME: 14 24 2 SECONDS
RUN-TIME RATIO: 221/41=5.3
CORE USED: 11K (21 PAGES)

DOCUMENT PAGES: 65

E06

EDCHI MACY11 27.1006 29-SEP-76 14:44 PAGE 69
EDCHIC.F11 29-SEP-76 14:42 CROSS REFERENCE TABLE -- MACRO NAMES

SEP 3066

Spooler runtime 11 Seconds, 49 KCS, 259 disk reads, 4 disk writes, 66 pages

[illegible]