

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

.REMK

IDENTIFICATION

PRODUCT CODE: AC-8484F-MC
PRODUCT NAME: CZDHKFO DH11 MODEM CONTROL MULTIPLEXER DIAGNOSTIC
DATE : 18-JUN-1985
MAINTAINER: NAC SOFTWARE ENGINEERING
AUTHOR: G. BAISLEY

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.

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

COPYRIGHT (C) 1974, 1977, 1984, 1985 BY DIGITAL EQUIPMENT CORPORATION

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

1.0 ABSTRACT

THIS PROGRAM IS A TEST OF THE MODEM CONTROL MULTIPLEXER USED WITH THE DH11 OPTION

THE PROGRAM IS DIVIDED INTO FUNCTIONAL TEST GROUPS AS FOLLOWS:

- GROUP 0: ALL LINE SCANNER AND LINE MULTIPLEXER FUNCTIONS ARE TESTED. NO TEST CONNECTOR IS NEEDED...
- GROUP 1: A SINGLE LINE IS TESTED USING THE MODEM CABLE AND A H315 TEST CONNECTOR
- GROUP 2: CONNECT-DISCONNECT TEST FOR 103A MODEMS
- GROUP 3: CONNECT DISCONNECT TEST FOR 202C MODEMS

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11 COMPUTER WITH AT LEAST 8K OF MEMORY WITH OR WITHOUT HARDWARE SWITCH REGISTER ASR-33 TELETYPE OR EQUIVALENT MODEM CONTROL MODULES

2.1.1 FOR 16 LINE SCANNER TEST

NO ADDITIONAL HARDWARE IS NEEDED. PROGRAM HAS BEEN MODIFIED TO RUN WITHOUT H861 TEST CONNECTOR.

2.1.2 FOR SINGLE LINE CABLE TEST

4 CABLES TO CONNECT TO THE DISTRIBUTION PANEL H315 TEST CONNECTOR

2.1.3 FOR ON LINE TESTS

4 CABLES TO CONNECT TO THE DISTRIBUTION PANEL
2 BELL 103A MODEMS (FOR 103A TEST)
2 BELL 202C MODEMS (FOR 202C TEST)

3.0 LOADING PROCEDURE

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

4.0 STARTING PROCEDURE

4.1 STARTING ADDRESS

THE STARTING ADDRESS FOR ALL TESTS IS 000200.

RESTART ADDRESS FOR ALL TESTS IS 000200

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

4.2 OPERATOR AND/OR PROGRAM ACTION

4.2.1 INITIAL PROGRAM START

NOTE

IF PROGRAM IS BEING RUN WITH THE "XOR" MODULE TESTER
LOCATION 1030(8) MUST BE MODIFIED TO CONTAIN A 240(8)
'NOP" TO ACTIVATE THAT CODE AFFECTING THE "XOR" TESTER.

NOTE

SOFTWARE SWITCH REGISTER IS DEFINED AS LOC. 176
(REFER TO SECTION 5.1.2 FOR DYNAMIC LOADING INSTRUCTIONS)

4.2.1.1 LOAD ADDRESS 000200
SET SW00 = 1
PRESS START
***SOFTWARE SWITCH REGISTER IS LOC. 176

4.2.1.2 PROGRAM WILL TYPE
"DH11-MODEM CONTROL DIAGNOSTIC "(ONCE ONLY)
***NOTE: IF USING SOFTWARE SWITCH REGISTER THE FOLLOWING
WILL BE TYPED BEFORE TITLE:
SWR=XXXXXX NEW= (REFER TO SECTION 5.1.2 FOR OPTIONS)

4.2.1.3 PROGRAM WILL TYPE (WITH SW00 = 1)
VECTOR ADDRESS-" AND WILL WAIT FOR AN INPUT
FROM THE TELETYPE KEYBOARD.

4.2.1.4 TYPE A THREE DIGIT NUMBER (OCTAL) WHICH IS THE
ADDRESS THAT THE MODEM CONTROL WILL INTERRUPT TO, FOLLOWED BY
<RETURN>. IF AN INCORRECT ADDRESS IS TYPED, THE PROGRAM WILL
TYPE "?" AND THEN REPEAT 4.2.1.3.

NOTE: IF THE ADDRESS ENTERED IS ACCEPTIBLE TO THE PROGRAM,
BUT IS NOT THE INTERRUPT VECTOR ADDRESS OF THE MODEM CONTROL
UNDER TEST, A HALT WILL OCCUR AT THAT ADDRESS+2, WHEN
THE MODEM CONTROL INTERRUPTS.

TO RECOVER, PERFORM 4.2.2.1.

4.2.1.5 THE PROGRAM WILL TYPE "CONTROL REGISTER ADDRESS-" AND WAIT FOR
AN INPUT FROM THE TELETYPE KEYBOARD.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

4.2.1.6 TYPE A 6 DIGIT (OCTAL NUMBER) WHICH IS THE ADDRESS OF THE MODEM CONTROL'S CONTROL REGISTER FOLLOWED BY <RETURN>. IF AN INCORRECT ADDRESS IS TYPED, THE PROGRAM WILL TYPE "?" AND THEN REPEAT 4.2.1.6.

NOTE: IF THE ADDRESS ENTERED IS ACCEPTIBLE TO THE PROGRAM BUT IS A NON-EXISTANT REGISTER, A BUS ERROR TRAP WILL OCCUR WHEN THE PROGRAM ADDRESSES THE REGISTER, AND THE PROGRAM WILL HALT AT LOCATION 6.

TO RECOVER, PERFORM 4.2.2.1.

4.2.1.7 THE PROGRAM WILL TYPE "LINE SELECTION PARAMETER-" AND WAIT FOR INPUT FROM THE TTY KEYBOARD.

4.2.1.8 TYPE AN OCTAL NUMBER TO SPECIFY THE LINES TO BE TESTED USING THE FOLLOWING ENCODING SCHEME:

| | |
|-----------|--------------------|
| BIT00 = 1 | TEST LINE 00 |
| BIT01 = 1 | TEST LINE 01 |
| BIT02 = 0 | DO NOT TEST LINE 2 |

"

| | |
|-----------|--------------|
| BIT15 = 1 | TEST LINE 15 |
|-----------|--------------|

EG: TYPING 377(8) SELECTS LINES 00 THRU 07
TYPING 17777(8) SELECTS ALL 16 LINES

IF THE NO. TYPED IS NOT ACCEPTABLE, THE PROGRAM TYPES A "?" AND ASKS FOR THE LINE SELECT PARAMETER AGAIN.

4.2.1.9 THE PROGRAM WILL TYPE "TEST-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD.

4.2.1.10 TYPE A THREE DIGIT OCTAL NUMBER CORRESPONDING TO THE NUMBER OF THE TEST TO BE RUN FOLLOWED BY <RETURN>. IF AN INCORRECT TEST NUMBER IS TYPED THE PROGRAM WILL TYPE "?" AND THEN REPEAT 4.2.1.7. THE AVAILABLE TESTS TOGETHER WITH THE NUMBER TO BE TYPED ARE GIVEN BELOW.

TEST GROUP 0:
OFF LINE TESTS -FIRST TEST=0
TEST GROUP 1:
OFF LINE TESTS USING DC11 TEST CONNECTOR AND MODEM CABLE-FIRST TEST=100
TEST GROUP 2:
CONNECT/DISCONNECT TEST FOR BELL 103A MODEMS-FIRST TEST=200
TEST GROUP 3:
CONNECT/DISCONNECT TEST FOR BELL 202C MODEMS-FIRST TEST=300

4.2.1.11 THE PROGRAM WILL ENTER THE SELECTED TEST GROUP.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52

4.2.2 PROGRAM RESTART

4.2.2.1 WITH SW00=1

LOAD ADDRESS 200
SET SW00=1 BEFORE PRESSING START.
SOFTWARE SWITCH REGISTER IS LOC 176
PRESS START

PROGRAM WILL PERFORM AS DESCRIBED IN 4.2.1.3 TO 4.2.1.10.

4.2.2.2 WITH SW00=0

LOAD ADDRESS 200
***SOFTWARE SWITCH REGISTER IS LOC. 176
PRESS START

PROGRAM WILL PERFORM AS DESCRIBED IN 4.2.1.7 TO 4.2.1.10

5.0 OPERATING PROCEDURE

5.1 TEST GROUP 0 16 LINE SCANNER TEST

5.1.1 TEST INITIALIZATION

NONE REQUIRED, PROGRAM TYPES "16 LINE SCANNER TEST"
AND BEGINS TEST EXECUTION.

5.1.2 OPERATIONAL SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G <+G>; THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE 'NEW=' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46

A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0 7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.

B) IF A CONTROL U <+U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

SW15=1, HALT ON ERROR
SW14=1, LOOP ON CURRENT TEST
SW13=1, SUPPRESS ERROR TYPEOUT
SW11=1, SUPPRESS ITERATIONS
SW10=1, ESCAPE TO NEXT TEST ON ERROR
SW09=1, FREEZE DATA

5.1.3 PROGRAM AND/OR OPERATOR ACTION

5.1.3.1 WITH ALL SWITCHES DOWN, THE PROGRAM WILL RUN ALL TESTS IN THE SELECTED GROUP, SEQUENTIALLY. EACH TEST IS REPEATED A FIXED NUMBER OF TIMES (SEE LISTING FOR DETAILS), EXCEPT FOR TO WHICH IS EXECUTED ONCE ONLY AFTER START OF TEST. WHEN ALL TESTS HAVE BEEN COMPLETED, THE PROGRAM WILL ISSUE A "RESET", RING THE TELETYPE BELL, AND RESTART AT THE FIRST TEST OF THE SELECTED GROUP.

IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE AND CONTINUE TESTING.

5.1.3.2 WITH SW15=1, PROGRAM ACTION WILL BE AS IN 5.1.3.1 EXCEPT THAT A HALT WILL OCCUR AFTER ERROR TYPEOUT.
NOTE: IF USING THE SOFTWARE SWITCH REGISTER AND AN EROR HALT OCCURS. THE SOFTWARE SWITCH REGISTER CAN BE CHANGED BY PRESSING CONTINUE THE PROGRAM WILL RESPOND WITH THE FOLLOWING:
SWR=XXXXXX NEW=

5.1.3.3 WITH SW13=1, PROGRAM ACTION WILL BE AS IN 5.1.3.1 EXCEPT THAT NO ERROR TYPEOUT WILL OCCUR. THE PC OF THE TEST THAT FAILED WILL BE DISPLAYED IN THE COMPUTER DATA LIGHTS.

5.1.3.4 THIS PROGRAM WILL NO LONGER TRACE TRAP WITH THIS RELEASE

5.1.3.5 WITH SW10=1, PROGRAM ACTION WILL BE AS IN 5.1.3.1 EXCEPT THAT AFTER AN ERROR HAS OCCURED, THE PROGRAM WILL IMMEDIATELY START THE NEXT TEST IN SEQUENCE.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

5.2 TEST GROUP 1 SINGLE LINE CABLE TEST

5.2.1 TEST INITIALIZATION

THE PROGRAM WILL TYPE "SINGLE LINE CABLE TEST
LINE NUMBER-" AND WILL WAIT FOR AN INPUT FROM
THE TELETYPE KEYBOARD.

TYPE A 2 DIGIT OCTAL NUMBER BETWEEN 0 AND 17, CORRESPONDING
TO THE NUMBER OF THE LINE TO BE TESTED, FOLLOWED BY
<RETURN>. THE PROGRAM WILL THEN BEGIN TEST EXECUTION.
IF THE TELETYPE INPUT IS INCORRECT, THE PROGRAM
WILL TYPE "?" AND REPEAT THE MESSAGE.

5.2.2 OPERATIONAL SWITCH SETTINGS

SAME AS 5.1.2

5.2.3 PROGRAM AND/OR OPERATOR ACTION

SAME AS 5.1.3

5.3 TEST GROUP 2 BELL 103A MODEM CONNECT-DISCONNECT TEST

5.3.1 TEST INITIALIZATION

THE PROGRAM WILL TYPE "103A CONNECT-DISCONNECT TEST
ORIGINATE LINE-" AND WAIT FOR AN INPUT FROM THE TELETYPE
KEYBOARD.

TYPE THE NUMBER OF THE LINE THAT WILL ORIGINATE THE
CALL (0-17 OCTAL) FOLLOWED BY RETURN.

THE PROGRAM WILL TYPE "ANSWER LINE-" AND WILL WAIT
FOR AN INPUT FROM THE TELETYPE KEYBOARD.

TYPE THE NUMBER OF THE LINE THAT WILL ANSWER THE CALL
(0-17 OCTAL) FOLLOWED BY <RETURN>.

THE PROGRAM WILL TYPE "DIAL ANSWERING DATA SET"
AND WILL WAIT FOR THE ORIGINATE AND ANSWERING MODEMS
TO GENERATE INTERRUPTS.

5.3.2 OPERATOR ACTION TO MAKE TELEPHONE CONNECTION

AFTER THE MESSAGE "DIAL ANSWERING DATA SET" IS TYPED
THE OPERATOR HAS APPROXIMATELY 5 MINUTES TO ESTABLISH
A CONNECTION BETWEEN THE 2 DATA SETS.

5.3.2.1 PLACE ANSWERING DATA SET IN "AUTO ANSWER" MODE

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

5.3.2.2 PLACE ORIGINATING DATA SET IN "TALK" MODE

5.3.2.3 DIAL DIAL ANSWERING DATA SET FROM ORIGINATING DATA SET

5.3.2.4 LISTEN FOR TONE IN HANDSET OF ORIGINATING DATA SET.

WHEN TONE IS HEARD, PRESS "DATA" BUTTON ON ORIGINATING DATA SET.

"DATA" LIGHT SHOULD ILLUMINATE

5.3.2.5 "DATA" LIGHT ON ANSWERING DATA SET SHOULD BE LIT.

5.3.2.6 THE PROGRAM WILL NOW WAIT FOR INTERRUPTS FROM THE MODEM CONTROL.

5.3.2.7 IF THE CONNECTION HAS BEEN PROPERLY ESTABLISHED, THE PROGRAM WILL TYPE "TYPE TTY KEY TO DISCONNECT".

WHEN TTY KEY IS STRUCK, THE PROGRAM WILL BEGIN THE DISCONNECT SEQUENCE.

5.3.2.8 WHEN THE DISCONNECT SEQUENCE HAS BEEN COMPLETED THE PROGRAM WILL TYPE "103A TEST COMPLETE", AND WILL REQUEST THE OPERATOR TO SELECT NEW LINES.

5.3.3 PROGRAM ACTION IN CASE OF ERROR

5.3.3.1 RING ON INCORRECT LINE

IF THE PROGRAM DETECTS A RING SIGNAL ON AN INCORRECT LINE, OR IF ANY OTHER TRANSITION BESIDES RING IS DETECTED BEFORE RING, THE PROGRAM WILL TYPE A FATAL ERROR MESSAGE AND REQUEST THE OPERATOR TO RESELECT LINES AND REDIAL.

5.3.3.2 OTHER ERRORS

IF ANY ERRORS OCCUR AFTER THE FIRST RING HAS BEEN DETECTED, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE AND CONTINUE TESTING TO COMPLETION.

THE ONLY EXCEPTION TO THIS IS IF AN INTERRUPT OCCURS ON A LINE NOT SELECTED, IN WHICH CASE A FATAL ERROR WILL BE REPORTED, AND THE PROGRAM WILL PROCEED AS DESCRIBED IN 5.3.3.1

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
5.3.4 OPERATION SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G <+G>; THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE ''NEW='' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
 - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
 - B) IF A CONTROL U <+U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

SW15=1, HALT ON ERROR
SW13=1, SUPPRESS ERROR TYPEOUT

5.3.5 DATA SET MODE SWITCHING

AFTER THE PROGRAM HAS TYPED THE MESSAGE DESCRIBED IN 5.3.2.7, BUT BEFORE TTY KEY IS STRUCK, THE OPERATOR MAY SWITCH EITHER DATA SET FROM THE MODE THAT IT IS IN TO ANOTHER MODE. ALL TRANSITIONS DETECTED AT THIS TIME WILL BE REPORTED.

NOTE: THE ORIGINATE DATA SET MUST BE RETURNED TO "TALK" MODE AND THE ANSWERING DATA SET TO "AUTO ANSWER" BEFORE DISCONNECT IS STARTED TO PREVENT ERRORS FROM BEING DETECTED THAT ARE CAUSED BY THE FACT THAT THE MODEM IS IN THE INCORRECT STATE.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53

- 5.4 TEST GROUP 3 BELL 202C MODEM CONNECT DISCONNECT TEST
- 5.4.1 TEST INITIALIZATION
SAME AS 5.3.1 EXCEPT PROGRAM WILL TYPE "202C CONNECT DISCONNECT TEST".
- 5.4.2 OPERATOR ACTION TO MAKE TELEPHONE CONNECTION
SAME AS 5.3.2 EXCEPT AT END OF TEST, PROGRAM WILL TYPE "202C TEST COMPLETE".
- 5.4.3 PROGRAM ACTION IN CASE OF ERRORS
SAME AS 5.3.3
- 5.4.4 OPERATIONAL SWITCH SETTINGS
SAME AS 5.3.4
- 5.4.5 DATA SET MODE SWITCHING
SAME AS 5.3.5
- 5.5 TEST RESELECTION
TO ESCAPE FROM THE TEST IN PROGRESS, AND SELECT A NEW TEST, TYPE <CONTROL C>.
THE PROGRAM WILL STOP EXECUTION OF THE TEST IN PROGRESS AND THEN TYPE "TEST-" AND WAIT FOR AN INPUT FROM THE TELETYPE KEYBOARD.
PROCEED AS DESCRIBED IN 4.2.1.8
- 5.5 ADDRESS CHANGE
TO CHANGE THE VECTOR AND REGISTER ADDRESS OF THE MODEM CONTROL UNDER TEST, TYPE <CONTROL V>. THE PROGRAM WILL STOP EXECUTION OF THE TEST IN PROGRESS AND PROCEED AS DESCRIBED IN SECTION 4.2.1, EXCEPT THAT "MODEM CONTROL DIAGNOSTIC" WILL NOT BE TYPED.
- 5.6 LINE NUMBER CHANGE
TO CHANGE THE LINE NUMBER(S) UNDER TEST, TYPE <CONTROL L>. THE PROGRAM WILL SUSPEND THE TEST IN PROGRESS AND RETURN TO THE INITIALIZATION STAGE OF THE SELECTED TEST.
WHEN THE LINE NUMBER(S) HAS BEEN CHANGED, THE PROGRAM WILL RESTART THE SELECTED TEST USING THE NEW LINE NUMBER(S).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42

5.7 POWER FAILURE

IF A POWER FAIL TRAP OCCURS DURING TEST EXECUTION THE PROGRAM WILL SAVE THE GENERAL REGISTERS OF THE PROCESSOR AND HALT.

WHEN POWER UP OCCURS, THE PROGRAM WILL TYPE "POWER FAILURE-CURRENT TEST WILL BE RESTARTED".

THE PROGRAM WILL THEN RUSUME TEST EXECUTION.

NOTE: IF A TEST IS NOT IN PROGRESS, I.E., IF THE PROGRAM IS WAITING FOR AN INPUT FROM THE TELETYPE KEYBOARD, THE ERROR MESSAGE WILL BE "POWER FAILURE". THE PROGRAM WILL THEN REQUEST THE OPERATOR TO SELECT A TEST.

NOTE: IF MACHINE HAS A SOLID-STATE SWITCH REGISTER, THEN THE CONTENTS WILL BE LOST ON A POWER FAIL AND THEREFORE WILL HAVE TO BE RELOADED.

6.0 ERRORS

6.1 NORMAL OPERATION

IF AN ERROR OCCURS WITH ALL SWITCHES DOWN, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE AND THEN RESUME TESTING.

THERE ARE SEVERAL ERROR MESSAGE FORMATS, AND THE PARTICULAR MESSAGE TYPED DEPENDS UPON THE TEST IN PROGRESS.

6.1.1 ERROR MESSAGES

6.1.1.1 UNIQUE ERROR

ONLY PC OF FAILING TEST IS OUTPUT TO TELEPRINTER

AN EXAMPLE OF THIS TYPE OF ERROR IS:

1. AN INTERRUPT OCCURED AT THE WRONG PRIORITY
2. A REGISTER BIT WAS NOT CLEARED BY RESET

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
6.1.1.2 TRANSITION DETECTION ERROR

THIS ERROR WILL OCCUR IN ONE OF THE ON-LINE TESTS
IF AN EXPECTED INTERRUPT DOES NOT OCCUR, OR IF
AN UNEXPECTED INTERRUPT DOES OCCUR, ON THE LINES
UNDER TEST.

FORMAT FOR ERROR TYPEOUT IS

XXXXXX TRANSITION ERROR
EXP REC LINE
AA BB CC

WHERE XXXXX=PC+2 OF CALL TO ERROR ROUTINE
AA=EXPECTED INTERRUPT FLAGS (CORRESPONDS TO 4 MSB OF CONTROL REGISTER
BB=RECEIVED INTERRUPT FLAGS (AS ABOVE)
CC=LINE ON WHICH ERROR OCCURED

6.1.1.3 SINGLE LINE STATUS ERROR

THIS ERROR WILL OCCUR IN ANY TEST, OFF LINE OR ON-LINE
WHEN THE EXPECTED AND RECEIVED LINE STATUS ARE NOT
THE SAME.

FORMAT FOR SINGLE LINE STATUS ERROR IS

XXXX LINE ERROR
EXP REC LINE
AAA BBB CC

WHERE XXXXX=PC+2 OF CALL TO ERROR ROUTINE
AAA=EXPECTED LINE STATUS AT TIME OF ERROR
BBB=RECEIVED LINE STATUS AT TIME OF ERROR
CC=LINE ON WHICH ERROR OCCURED

6.1.1.4 FATAL TRANSITION ERROR

THIS ERROR WILL OCCUR IN AN ON-LINE TEST IF AN INTERRUPT
OCCURS ON A LINE NOT SELECTED FOR TESTING.

FORMAT FOR FATAL ERROR TYPEOUT IS

XXXXXX FATAL ERROR
CSTAT LSTAT
AAAAAA BBB

WHERE XXXXX=PC+2 OF CALL TO ERROR ROUTINE
AAAAAA=RECEIVED CONTROL STATUS ON LINE THAT INTERRUPTED
BBB=RECEIVED LINE STATUS ON LINE THAT INTERRUPTED

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36

6.1.1.4 CONTROL STATUS ERROR

THIS ERROR WILL OCCUR IN A TEST THAT PRIMARILY INVOLVES THE LINE SCANNER

FORMAT FOR CONTROL STATUS ERROR IS

XXXXXX STATUS ERROR
EXP REC
AAAAA BBBB

WHERE XXXXXX=PC+2 OF CALL TO ERROR ROUTINE
AAAAA=EXPECTED CONTROL STATUS AT TIME OF ERROR
BBBBB=RECEIVED(ACTUAL) CONTROL STATUS AT TIME OF ERROR

6.1.1.5 LINE STATUS ERROR

THIS ERROR WILL OCCUR IN THOSE OFF LINE TESTS THAT SET ONE LINE TO A PARTICULAR STATE, AND THEN CHECK ALL OTHER LINES

FORMAT FOR LINE STATUS ERROR IS

XXXX LINE ERROR
EXP REC LINE SEL
AAA DDD CC DD

WHERE XXXXXX=PC+2 OF CALL TO ERROR ROUTINE
AAA=EXPECTED LINE STATUS AT TIME OF ERROR
BBB=RECEIVED LINE STATUS AT TIME OF ERROR
CC=LINE ON WHICH ERROR OCCURED
DD=THE LINE ON WHICH THE PROGRAM WAS OPERATING

37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

6.1.1.6 TIME OUT ERROR

THIS ERROR WILL OCCUR IF THE LINE UNDER TEST DOES NOT INTERRUPT WITHIN A GIVEN TIME FRAME.

FORMAT FOR THIS ERROR IS

XXXXXX TIME OUT WAITING FOR INTERRUPT
LN CSR LSR
AAA BBBBBB CCCCC

WHERE XXXXXX=PC+2 OF ERROR CALL
AAA=FAILING LINE NUMBER
BBBBBB=CONTROL STATUS REGISTER
CCCCC=LINE STATUS REGISTER

6.1.2 REPEATED ERRORS

IF THE SAME ERROR OCCURS REPEATEDLY IN A GIVEN TEST ONLY THE DATA RELATING TO THAT ERROR WILL BE TYPED IF THE ERROR OCCURS IN THE SAME TEST ON THE SAME PASS

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55

6.2 SCOPE LOOPS

NOTE: SCOPE LOOPING APPLIES ONLY TO TEST GROUPS 0 AND 1

6.2.1 AFTER ERROR HALT

TO LOOP ON A GIVEN TEST AFTER AN ERROR HALT,
SET SW15=0 TO RUN WITHOUT STOPPING
SET SW14=1 TO LOOP ON CURRENT TEST
SET SW13=1 TO SUPPRESS ERROR TYPEOUT
SET SW10=0 (IF IT IS 1)
SET SW09=1 TO LOOP ON SAME DATA (IF REQUIRED)

***IF USING SOFTWARE SWITCH REGISTER AND YOU WANT TO CHANGE
THE SWITCH SETTING TYPE A <+G> BEFORE CONTINUING.
PRESS CONTINUE

THE PROGRAM WILL LOOP ON THE SAME TEST.

6.2.2 FROM PROGRAM START

6.2.2.1 PROCEED AS DESCRIBED IN 4.2.1.1 TO 4.2.1.4

6.2.2.2 WHEN THE PROGRAM TYPES "TEST-", SET SW14=1 TO LOOP
ON THE TEST THAT WILL BE SELECTED.

6.2.2.3 TYPE IN THE NUMBER OF THE TEST THAT IS TO BE LOOPED
ON (SEE LISTING FOR TEST NUMBER REFERENCE DESIGNATIONS)

6.2.2.4 THE PROGRAM WILL LOOP ON THE SELECTED TEST UNTIL
SW14=0.

6.2.3 AFTER <CONTROL>

SAME AS 6.2.2.2 TO 6.2.2.4

7.0 RESTRICTIONS

7.1 STARTING

7.1.1 FOR 16 LINE SCANNER TEST

NO TEST CONNECTOR IS NEEDED TO RUN THIS TEST....

7.1.2 FOR SINGLE LINE CABLE TEST

H315 TEST CONNECTOR MUST BE INSTALLED ON MODEM CABLE

7.1.3 FOR ON LINE TESTS

NONE

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

7.2 OPERATING

NONE.

7.3 WHEN ON ACT 11 OR 'XOR'
PROGRAM WILL DEFAULT TO 16 LINE SCANNER TEST

7.4 DEFAULT PARAMETERS (INCLUDING ACT-11 & "XOR")

VECTORS

DHMVEC: 300 (AUTOMATICALLY GENERATED
DHMLVL: 302 BY PROGRAM WHEN UNDER ACT-11 OR "XOR")
ADDRESSES

DHMCSR: 170500
DHMLSR: 170502

NOTE: SWOOC(RESELECT ADDRESSES AND VECTORS BECOMES
INOPERATIVE UNDER ACT-11 OR 'XOR'.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52

8.0 EXECUTION TIME

8.1 16 LINE SCANNER TEST

THE TIME FOR 2 PASSES OF THE 16 LINE SCANNER TEST IS APPROXIMATELY 1.5 MINUTES.

8.2 SINGLE LINE CABLE TEST

THE TIME FOR 12 PASSES OF THE SINGLE LINE CABLE TEST IS APPROXIMATELY 1 MINUTE.

8.3 103A MODEM CONNECT-DISCONNECT TEST

APPROXIMATELY 30 SECONDS WILL ELAPSE BETWEEN THE TIME THAT THE ANSWERING DATA SET FIRST DETECTS A RING SIGNAL TO THE TIME THAT THE PROGRAM TYPES "SET SW01=1 TO DISCONNECT".

APPROXIMATELY 30 SECONDS WILL ELAPSE BETWEEN THE TIME THAT THE PROGRAM TYPES THE ABOVE MESSAGE UNTIL THE TIME THAT THE PROGRAM TYPES "103A TEST COMPLETE".

8.4 202C MODEM CONNECT-DISCONNECT TEST

APPROXIMATELY 1.5 MINUTES WILL ELAPSE BETWEEN THE TIME THAT THE ANSWERING DATA SET DETECTS THE FIRST RING SIGNAL TO THE TIME THAT THE PROGRAM TYPES "SET SW01=1 TO DISCONNECT".

APPROXIMATELY 30 SECONDS WILL ELAPSE BETWEEN THE TIME THAT THE PROGRAM TYPES THE ABOVE MESSAGE UNTIL THE PROGRAM TYPES "202C TEST COMPLETE".

9. PROGRAM DESCRIPTION

THIS PROGRAM CONSISTS OF A SERIES OF TEST GROUPS LINKED BY A SET OF COMMON SERVICE ROUTINES AND A KEYBOARD MONITOR.

WHEN INITIALLY LOADED AND STARTED ...SW00 MUST BE SET =1, THE PROGRAM WILL BEGIN A DIALOG WITH THE OPERATOR TO INPUT THE PARAMETERS REQUIRED BY THE PROGRAM.

WHEN ALL INFORMATION HAS BEEN INPUTTED, THE PROGRAM WILL REQUEST THE OPERATOR TO SELECT A TEST BY TYPING THE NUMBER OF THE TEST TO BE RUN. WHEN A CORRECT TEST NUMBER IS RECEIVED, THE PROGRAM WILL BEGIN EXECUTION OF THE SELECTED TEST.

AT ANY TIME DURING TEST EXECUTION, THE OPERATOR MAY CHANGE A TEST PARAMETER BY ENTERING THE APPROPRIATE COMMAND VIA THE TELETYPE KEYBOARD.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

9. CONT'D

IF AN OFF LINE TEST HAS BEEN SELECTED, THAT TEST WILL BE REPEATED UNTIL THE OPERATOR INTERVENES.

IF AN ON LINE TEST HAS BEEN SELECTED, THE OPERATOR IS REQUIRED TO TAKE ACTION EACH TIME THE TEST IS COMPLETED.

AT THE END OF EVERY OFF LINE TEST PASS, THE PROGRAM WILL RING THE TELETYPE BELL.

AT THE END OF AN ON LINE TEST, A TEST COMPLETE MESSAGE WILL BE TYPED.

10. LISTING

11. MODIFICATION HISTORY:

10-JULY-84 KEN RAUHALA
ADDED DELAY FOR PDP-11/44 WITH CACHE ON.
THE "MUXS2" MACRO WAS MODIFIED.

✱

```

1
2 000000
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

```

```

.TITLE CZDHK-F
.ENABLE ABS,AMA
;MODEM CONTROL DIAGNOSTIC
;THIS PROGRAM CONTAINS TEST OF THE MODEM CONTROL IN
;THE OFF LINE MODE OF OPERATION ONLY
;MODIFIED BY ED CROWLEY APRIL, 1976
;MODIFIED BY S. CARPENTER JULY, 1976 TO SUPPORT THE SOFTWARE SWITCH
;REGISTER.
;ALSO, SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER.
;MODIFIED APRIL 77 TO RUN SCANNER TEST W/O H861 CONNETTOR
;
;MODIFIED BY S. SKONETSKI MAY 1985
;REMOVE .EQUIV WHICH IS NOT RECOGNISED BY PDP11 MACRO, AND TO
;DYNAMICALLY SET VECTOR SPACE WHICH IS CLOBBED BY THE NEW XXDP

;SWITCH REGISTER OPTIONS

;SW15=1, HALT ON ERROR
;SW14=1, LOOP ON CURRENT TEST
;SW13=1, SUPPRESS ERROR TYPEOUT
;SW12=1, SUPPRESS TRACE TRAPPING(THIS IS INOPERATIVE IN THIS RELEASE)
;SW11=1, SUPPRESS ITERATIONS
;SW10=1, ESCAPE TO NEXT TEST ON ERROR
;SW09=1, FREEZE DATA
;SW01=1, START DISCONNECT SEQUENCE
;SW00=1, RESELECT VECTOR AND CONTROL REGISTER ADDRESS
;AFTER PROGRAM RESTART

;STARTING ADDRESS FOR ALL TESTS IS 000200
;RESTART ADDRESS=000200

;TESTS AVAILABLE

;TEST GROUP 0-
;OFF LINE TESTS USING NO TEST CONNECTOR-FIRST TEST=0
;TEST GROUP 1-
;OFF LINE TESTS USING DC11 TEST CONNECTOR AND MODEM CABLE-FIRST TEST=100
;TEST GROUP 2-
;CONNECT/DISCONNECT TEST FOR BELL 103A MODEMS-FIRST TEST=200
;TEST GROUP 3-
;CONNECT/DISCONNECT TEST FOR BELL 202C MODEMS-FIRST TEST=300

;SYMBOL DEFINITIONS

100000 SW15=100000
040000 SW14=40000
020000 SW13=20000
010000 SW12=10000
004000 SW11=4000
002000 SW10=2000
001000 SW09=1000
000400 SW08=400
000100 SW06=100

.NLIST MC,MD,CND
.LIST ME

```

```
60
61      .MACRO COMMENT
62
63      .NLIST
64      ;CODITIONAL ASSEMBLY PARAMETERS
65
66      SINGLE=0      ;IF 0, ASSEMBLE FOR MULTIPLE LINES
67                  ;IF 1, ASSEMBLE FOR SINGLE LINE
68      .LIST
69      .ENDM
70 000000          COMMENT
                  ;CODITIONAL ASSEMBLY PARAMETERS
                  SINGLE=0      ;IF 0, ASSEMBLE FOR MULTIPLE LINES
                              ;IF 1, ASSEMBLE FOR SINGLE LINE
```

000000

```

1
2
3           ;REGISTER DEFINITIONS
4           000000      R0=#0           ;GENERAL REGISTER
5           000001      R1=#1           ;GENERAL REGISTER
6           000002      R2=#2           ;GENERAL REGISTER
7           000003      R3=#3           ;GENERAL REGISTER
8           000004      R4=#4           ;GENERAL REGISTER
9           000005      R5=#5           ;GENERAL REGISTER
10          000006      SP=#6           ;PROCESSOR STACK POINTER
11          000007      PC=#7           ;PROGRAM COUNTER
12
13           ;LOCATION EQUIVALENCIES
14
15          177776      PS =177776
16          177776      PSW=177776      ;PROCESSOR STATUS WORD
17
18          .MACRO PS
19          PSW
20          .ENDM PS
21
22          014436      RADIX=DIVIS      ;CONVERSION FACTOR FOR DECIMAL OUTPUT
23          014432      BINWRD=DIVIDL   ;WORD TO BE CONVERTED TO OCTAL ASCII
24          014434      DIGIT=DIVIDH    ;ASCII OCTAL DIGIT
25
26           ;CONTROL STATUS REGISTER BIT FUNCTIONS
27
28          000020      BUSY=20          ;LINE SCANNER RUNNING
29          000040      SCNENA=40        ;LINE SCANNER ENABLE
30          000100      INTENA=100       ;INTERRUPT ENABLE
31          000200      DONE=200        ;SCANNER DONE
32          000400      STEP=400         ;CAUSES LINE COUNTER TO BE INCREMENTED BY 1 COUNT
33          001000      MAINT=1000       ;FORCES 1S TO INPUT OF SCRATCH PAD MEMORY
34          002000      CLRMUX=2000     ;CLEAR MULTIPLEXER FUNCTION FLIPFLOPS
35          004000      CLRSCN=4000     ;CLEARS SCANNER SCRATCHPAD MEMORY
36          010000      SECRCF=10000    ;SECONDARY RECEIVE TRANSITION WAS DETECTED BY SCANNER
37          020000      CSF=20000      ;CLEAR TO SEND TRANSITION WAS DETECTED BY SCANNER
38          040000      COF=40000      ;CARRIER TRANSITION WAS DETECTED BY SCANNER
39          100000      RINGF=100000    ;RING SIGNAL WAS DETECTED BY SCANNER
40
41           ;LINE REGISTER BIT FUNCTIONS
42
43          000001      LINENA=1         ;=1, RECOGNIZE TRANSITIONS ON THIS LINE
44          000002      TRMRDY=2        ;=1, SEND TERMINAL READY TO MODEM
45          000004      RS=4            ;=1, SEND REQUEST TO SEND TO MODEM
46          000010      SECTX=10        ;=1, SEND SECONDARY TRANSMIT TO MODEM
47          000020      SECRCX=20       ;=1, SECONDARY RECEIVE TURNED ON BY MODEM
48          000040      CS=40           ;=1, CLEAR TO SEND TURNED ON BY MODEM
49          000100      CO=100          ;=1, CARRIER TURNED ON BY MODEM
50          000200      RING=200        ;=1, RING TURNED ON BY MODEM
51
52           ;SOFTWARE TRANSITION FLAGS
53
54          000004      XCO=4            ;CARRIER TRANSITION WAS DETECTED
55          000002      XCS=2           ;CLEAR TO SEND TRANSITION WAS DETECTED
56          000001      XSCRX=1         ;SECONDARY RECEIVE TRANSITION WAS DETECTED

```

```

1
2
3      ;INSTRUCTION DEFINITIONS
4      005746      PUSH1SP=5746      ;DECREMENT PROCESSOR STACK 1 WORD
5      005726      POP1SP=5726      ;INCREMENT PROCESSOR STACK 1 WORD
6      010046      PUSHRO=10046     ;SAVE R0 ON STACK
7      012600      POPRO=12600     ;RESTORE R0 FROM STACK
8      024646      PUSH2SP=24646   ;DECREMENT STACK TWICE
9      022626      POP2SP=22626    ;INCREMENT STACK TWICE
10
11      ;EMT DEFINITION TABLE
12
13 000000      EMTDEF  ERRORC,+/CONTROL STATUS ERROR SERVICE/
                104000      ERRORC=EMT+X      ;CONTROL STATUS ERROR SERVICE
                000001      X=X+1
14 000000      EMTDEF  ERRORL,+/LINE STATUS ERROR SERVICE/
                104001      ERRORL=EMT+X     ;LINE STATUS ERROR SERVICE
                000002      X=X+1
15 000000      EMTDEF  SCOPE,+/SCOPE LOOP AND ITERATION SERVICE/
                104002      SCOPE=EMT+X     ;SCOPE LOOP AND ITERATION SERVICE
                000003      X=X+1
16 000000      EMTDEF  SCOPEF,+/DATA FREEZE SERVICE/
                104003      SCOPEF=EMT+X    ;DATA FREEZE SERVICE
                000004      X=X+1
17 000000      EMTDEF  TYPE,+/TELETYPE OUTPUT/
                104004      TYPE=EMT+X     ;TELETYPE OUTPUT
                000005      X=X+1
18 000000      EMTDEF  SAVOSP,+/SAVE R0-R5, PC+2 OF CALL/
                104005      SAVOSP=EMT+X   ;SAVE R0-R5, PC+2 OF CALL
                000006      X=X+1
19 000000      EMTDEF  OCTASC,+/CONVERT DATA TO ASCII AND TYPE/
                104006      OCTASC=EMT+X   ;CONVERT DATA TO ASCII AND TYPE
                000007      X=X+1
20 000000      EMTDEF  RESO5,+/RESTORE R0-R5/
                104007      RESO5=EMT+X   ;RESTORE R0-R5
                000010      X=X+1
21 000000      EMTDEF  CONVERT,+/ASCII CONVERSION ROUTINE/
                104010      CONVERT=EMT+X ;ASCII CONVERSION ROUTINE
                000011      X=X+1
22 000000      EMTDEF  EXTRACT,+/DIGIT EXTRACTION ROUTINE/
                104011      EXTRACT=EMT+X ;DIGIT EXTRACTION ROUTINE
                000012      X=X+1
23 000000      EMTDEF  ERROR,+/TYPE PC OF FAILING TESTS ONLY/
                104012      ERROR=EMT+X   ;TYPE PC OF FAILING TESTS ONLY
                000013      X=X+1
24 000000      EMTDEF  INSTRG,+/INPUT OCTAL DATA STRING/
                104013      INSTRG=EMT+X  ;INPUT OCTAL DATA STRING
                000014      X=X+1
25 000000      EMTDEF  ERRORT,+/TRANSITION ERROR/
                104014      ERRORT=EMT+X  ;TRANSITION ERROR
                000015      X=X+1
26 000000      EMTDEF  ERRORS,+/ON LINE STATUS ERROR/
                104015      ERRORS=EMT+X  ;ON LINE STATUS ERROR
                000016      X=X+1
27 000000      EMTDEF  ERRORN,+/FATAL TRANSITION/
                104016      ERRORN=EMT+X  ;FATAL TRANSITION
                000017      X=X+1

```

```
28 000000      104017      EMTDEF GETLNS,+/INPUT LINE NUMBERS/
                000020      GETLNS=EMT+X          ;INPUT LINE NUMBERS
                                X=X+1
29 000000      104020      EMTDEF SETUP,+/SET UP FOR ON LINE TEST/
                000021      SETUP=EMT+X          ;SET UP FOR ON LINE TEST
                                X=X+1
30 000000      104021      EMTDEF CKRING,+/CHECK FOR RING ON CORRERT LINE/
                000022      CKRING=EMT+X        ;CHECK FOR RING ON CORRERT LINE
                                X=X+1
31 000000      104022      EMTDEF WAITRN,+/WAIT FOR TRANSITIONS/
                000023      WAITRN=EMT+X        ;WAIT FOR TRANSITIONS
                                X=X+1
32 000000      104023      EMTDEF CKTRAN,+/CHECK TRANSITIONS/
                000024      CKTRAN=EMT+X        ;CHECK TRANSITIONS
                                X=X+1
33 000000      104024      EMTDEF WAITS,+/DELAY FOR TRANSIENTS/
                000025      WAITS=EMT+X        ;DELAY FOR TRANSIENTS
                                X=X+1
34 000000      104025      EMTDEF CNTLUU,+/CHANGE SWREG ROUTINE/
                000026      CNTLUU=EMT+X       ;CHANGE SWREG ROUTINE
                                X=X+1
35 000000      104026      EMTDEF CKINTT,+/CHECK FOR INTERRUPTS-FLAG STYLE/
                000027      CKINTT=EMT+X       ;CHECK FOR INTERRUPTS-FLAG STYLE
                                X=X+1
36 000000      104027      EMTDEF KBDIN,+/FAKE INTERRUPT ENTRY POINT/
                000030      KBDIN=EMT+X       ;FAKE INTERRUPT ENTRY POINT
                                X=X+1
37 000000      104030      EMTDEF ERRINT,+/TIME OUT ERROR FOR INTERRUPTS/
                000031      ERRINT=EMT+X      ;TIME OUT ERROR FOR INTERRUPTS
                                X=X+1
```



```

1
2
4
;TRAPCATCAER FOR ILLEGAL INTERRUPTS
.LIST MC,MD
.NLIST ME
7      000000
8      000200
9
10
11
.ENDR
000000 000002
000002 000000
000004 000006
000006 000000
000010 000012
000012 000000
000014 000016
000016 000000
000020 000022
000022 000000
000024 000026
000026 000000
000030 000032
000032 000000
000034 000036
000036 000000
000040 000042
000042 000000
000044 000046
000046 000000
000050 000052
000052 000000
000054 000056
000056 000000
000060 000062
000062 000000
000064 000066
000066 000000
000070 000072
000072 000000
000074 000076
000076 000000
000100 000102
000102 000000
000104 000106
000106 000000
000110 000112
000112 000000
000114 000116
000116 000000
000120 000122
000122 000000
000124 000126
000126 000000
000130 000132
000132 000000
000134 000136
000136 000000

```

| | | |
|--------|--------|------|
| 000140 | 000142 | .+2 |
| 000142 | 000000 | HALT |
| 000144 | 000146 | .+2 |
| 000146 | 000000 | HALT |
| 000150 | 000152 | .+2 |
| 000152 | 000000 | HALT |
| 000154 | 000156 | .+2 |
| 000156 | 000000 | HALT |
| 000160 | 000162 | .+2 |
| 000162 | 000000 | HALT |
| 000164 | 000166 | .+2 |
| 000166 | 000000 | HALT |
| 000170 | 000172 | .+2 |
| 000172 | 000000 | HALT |
| 000174 | 000176 | .+2 |
| 000176 | 000000 | HALT |
| 000200 | 000202 | .+2 |
| 000202 | 000000 | HALT |
| 000204 | 000206 | .+2 |
| 000206 | 000000 | HALT |
| 000210 | 000212 | .+2 |
| 000212 | 000000 | HALT |
| 000214 | 000216 | .+2 |
| 000216 | 000000 | HALT |
| 000220 | 000222 | .+2 |
| 000222 | 000000 | HALT |
| 000224 | 000226 | .+2 |
| 000226 | 000000 | HALT |
| 000230 | 000232 | .+2 |
| 000232 | 000000 | HALT |
| 000234 | 000236 | .+2 |
| 000236 | 000000 | HALT |
| 000240 | 000242 | .+2 |
| 000242 | 000000 | HALT |
| 000244 | 000246 | .+2 |
| 000246 | 000000 | HALT |
| 000250 | 000252 | .+2 |
| 000252 | 000000 | HALT |
| 000254 | 000256 | .+2 |
| 000256 | 000000 | HALT |
| 000260 | 000262 | .+2 |
| 000262 | 000000 | HALT |
| 000264 | 000266 | .+2 |
| 000266 | 000000 | HALT |
| 000270 | 000272 | .+2 |
| 000272 | 000000 | HALT |
| 000274 | 000276 | .+2 |
| 000276 | 000000 | HALT |
| 000300 | 000302 | .+2 |
| 000302 | 000000 | HALT |
| 000304 | 000306 | .+2 |
| 000306 | 000000 | HALT |
| 000310 | 000312 | .+2 |
| 000312 | 000000 | HALT |
| 000314 | 000316 | .+2 |
| 000316 | 000000 | HALT |
| 000320 | 000322 | .+2 |

| | | |
|--------|--------|------|
| 000322 | 000000 | HALT |
| 000324 | 000326 | .+2 |
| 000326 | 000000 | HALT |
| 000330 | 000332 | .+2 |
| 000332 | 000000 | HALT |
| 000334 | 000336 | .+2 |
| 000336 | 000000 | HALT |
| 000340 | 000342 | .+2 |
| 000342 | 000000 | HALT |
| 000344 | 000346 | .+2 |
| 000346 | 000000 | HALT |
| 000350 | 000352 | .+2 |
| 000352 | 000000 | HALT |
| 000354 | 000356 | .+2 |
| 000356 | 000000 | HALT |
| 000360 | 000362 | .+2 |
| 000362 | 000000 | HALT |
| 000364 | 000366 | .+2 |
| 000366 | 000000 | HALT |
| 000370 | 000372 | .+2 |
| 000372 | 000000 | HALT |
| 000374 | 000376 | .+2 |
| 000376 | 000000 | HALT |
| 000400 | 000402 | .+2 |
| 000402 | 000000 | HALT |
| 000404 | 000406 | .+2 |
| 000406 | 000000 | HALT |
| 000410 | 000412 | .+2 |
| 000412 | 000000 | HALT |
| 000414 | 000416 | .+2 |
| 000416 | 000000 | HALT |
| 000420 | 000422 | .+2 |
| 000422 | 000000 | HALT |
| 000424 | 000426 | .+2 |
| 000426 | 000000 | HALT |
| 000430 | 000432 | .+2 |
| 000432 | 000000 | HALT |
| 000434 | 000436 | .+2 |
| 000436 | 000000 | HALT |
| 000440 | 000442 | .+2 |
| 000442 | 000000 | HALT |
| 000444 | 000446 | .+2 |
| 000446 | 000000 | HALT |
| 000450 | 000452 | .+2 |
| 000452 | 000000 | HALT |
| 000454 | 000456 | .+2 |
| 000456 | 000000 | HALT |
| 000460 | 000462 | .+2 |
| 000462 | 000000 | HALT |
| 000464 | 000466 | .+2 |
| 000466 | 000000 | HALT |
| 000470 | 000472 | .+2 |
| 000472 | 000000 | HALT |
| 000474 | 000476 | .+2 |
| 000476 | 000000 | HALT |
| 000500 | 000502 | .+2 |
| 000502 | 000000 | HALT |

| | | |
|--------|--------|------|
| 000504 | 000506 | .+2 |
| 000506 | 000000 | HALT |
| 000510 | 000512 | .+2 |
| 000512 | 000000 | HALT |
| 000514 | 000516 | .+2 |
| 000516 | 000000 | HALT |
| 000520 | 000522 | .+2 |
| 000522 | 000000 | HALT |
| 000524 | 000526 | .+2 |
| 000526 | 000000 | HALT |
| 000530 | 000532 | .+2 |
| 000532 | 000000 | HALT |
| 000534 | 000536 | .+2 |
| 000536 | 000000 | HALT |
| 000540 | 000542 | .+2 |
| 000542 | 000000 | HALT |
| 000544 | 000546 | .+2 |
| 000546 | 000000 | HALT |
| 000550 | 000552 | .+2 |
| 000552 | 000000 | HALT |
| 000554 | 000556 | .+2 |
| 000556 | 000000 | HALT |
| 000560 | 000562 | .+2 |
| 000562 | 000000 | HALT |
| 000564 | 000566 | .+2 |
| 000566 | 000000 | HALT |
| 000570 | 000572 | .+2 |
| 000572 | 000000 | HALT |
| 000574 | 000576 | .+2 |
| 000576 | 000000 | HALT |
| 000600 | 000602 | .+2 |
| 000602 | 000000 | HALT |
| 000604 | 000606 | .+2 |
| 000606 | 000000 | HALT |
| 000610 | 000612 | .+2 |
| 000612 | 000000 | HALT |
| 000614 | 000616 | .+2 |
| 000616 | 000000 | HALT |
| 000620 | 000622 | .+2 |
| 000622 | 000000 | HALT |
| 000624 | 000626 | .+2 |
| 000626 | 000000 | HALT |
| 000630 | 000632 | .+2 |
| 000632 | 000000 | HALT |
| 000634 | 000636 | .+2 |
| 000636 | 000000 | HALT |
| 000640 | 000642 | .+2 |
| 000642 | 000000 | HALT |
| 000644 | 000646 | .+2 |
| 000646 | 000000 | HALT |
| 000650 | 000652 | .+2 |
| 000652 | 000000 | HALT |
| 000654 | 000656 | .+2 |
| 000656 | 000000 | HALT |
| 000660 | 000662 | .+2 |
| 000662 | 000000 | HALT |
| 000664 | 000666 | .+2 |

| | | |
|--------|--------|------|
| 000666 | 000000 | HALT |
| 000670 | 000672 | .+2 |
| 000672 | 000000 | HALT |
| 000674 | 000676 | .+2 |
| 000676 | 000000 | HALT |
| 000700 | 000702 | .+2 |
| 000702 | 000000 | HALT |
| 000704 | 000706 | .+2 |
| 000706 | 000000 | HALT |
| 000710 | 000712 | .+2 |
| 000712 | 000000 | HALT |
| 000714 | 000716 | .+2 |
| 000716 | 000000 | HALT |
| 000720 | 000722 | .+2 |
| 000722 | 000000 | HALT |
| 000724 | 000726 | .+2 |
| 000726 | 000000 | HALT |
| 000730 | 000732 | .+2 |
| 000732 | 000000 | HALT |
| 000734 | 000736 | .+2 |
| 000736 | 000000 | HALT |
| 000740 | 000742 | .+2 |
| 000742 | 000000 | HALT |
| 000744 | 000746 | .+2 |
| 000746 | 000000 | HALT |
| 000750 | 000752 | .+2 |
| 000752 | 000000 | HALT |
| 000754 | 000756 | .+2 |
| 000756 | 000000 | HALT |
| 000760 | 000762 | .+2 |
| 000762 | 000000 | HALT |
| 000764 | 000766 | .+2 |
| 000766 | 000000 | HALT |
| 000770 | 000772 | .+2 |
| 000772 | 000000 | HALT |
| 000774 | 000776 | .+2 |
| 000776 | 000000 | HALT |

13
14

.LIST ME
.NLIST MC,MD,CND

```

1
2
3           ;STANDARD INTERRUPT VECTORS
4           . =24
5 000024 015204      PFAIL           ;POWER FAIL HANDLER
6 000026 000340      340           ;SERVICE AT LEVEL 7
7 000030 013044      EMTSRV        ;EMT DISPATCH SERVICE
8 000032 000340      340           ;SERVICE AT LEVEL 7
9
10          . =46
11 000046 013026      LOGICAL        ;ACT11?
12
13          . =60
14 000060 002020      KBDINT        ;KEYBOARD MONITOR
15 000062 000340      340           ;SERVICE AT LEVEL 7
16          . =174
17 000174 000000      DISPREG:      0
18 000176 000000      SWREG: 0
19
20          . =200
21 000200 000167 000674  JMP      START      ;GO TO START OF PROGRAM
22
23
24          .MACRO TS      XN,X
25          .NLIST
26          ;REFERANCE NUMBER DEFINITION
27          .LIST
28          T'XN':           ;REFERENCE DESIGNATION
29          .NLIST
30          N'X' =N'X'+1
31          .LIST
32          .ENDM
33          .MACRO TSS      XNN
34          T'XNN':
35          .NLIST
36          NN=NN+1
37          .LIST
38          .ENDM
39
40          .MACRO COMMENT
41          .NLIST
42          ;EMT GENERATOR
43          X=0
44          .LIST
45          .ENDM
46 000204          COMMENT
47          000000          ;EMT GENERATOR
48          .MACRO EMTDEF FNCT,COMMNT
49          .NLIST          FNCT=EMT+X      ;'COMMNT'
50          X=X+1
51          .LIST
52          .ENDM
53          .MACRO COMMENT
54          .NLIST
55          ;TEST TABLE GENERATOR

```

```
56 .LIST
57 .ENDM
58 000204 COMMENT
;TEST TABLE GENERATOR
59
60 .MACRO TM XM,Y
61 .NLIST
62 TIMES=4000
63 .IIF GT MO-21,TIMES=400
64 .IIF GT MO-24,TIMES=200
65 .IIF GT MO-34,TIMES=4000
66 .LIST
67 T'XM'
68 TIMES
69 .NLIST
70 M'Y'=M'Y'+1
71 .LIST
72 .ENDM
```

```

1
2      001100      .-1100
3      001100      STACK:
4      001100 012702 000024      START:  MOV    #24, R2          ; SET UP VECTOR AREA
5      001104 012722 015204      MOV    #PFAIL, (R2)+      ; POWER FAIL HANDLER
6      001110 012722 000340      MOV    #340, (R2)+      ; SERVICE AT LEVEL 7
7      001114 012722 013044      MOV    #EMTSRV, (R2)+    ; EMT DISPATCH SERVICE
8      001120 012722 000340      MOV    #340, (R2)+      ; SERVICE AT LEVEL 7
9      001124 012737 013026 000046  MOV    #LOGICAL, @#46
10     001132 012737 002020 000060  MOV    #KBDINT, @#60
11     001140 012737 000340 000062  MOV    #340, @#62        ; SERVICE AT LEVEL 7
12     001146 005067 000644      CLR    TIPFLG           ; CLEAR TEST IN PROGRESS FLAG
13     001152 005077 014440      CLR    @TKCSR
14     001156 012706 001100      MOV    #STACK, SP      ; SET UP STACK POINTER
15
16     001162 013746 000006      SUSWR: MOV    @#6, -(SP)    ; SAVE VECTORS
17     001166 013746 000004      MOV    @#4, -(SP)
18     001172 012737 001212 000004  MOV    #64$, @#4        ; SET UP FOR TIMEOUT
19     001200 022777 177777 014420  CMP    #-1, @SWR       ; REFERENCE HARDWARE SWITCH REGISTER
20     001206 001402      BEQ    65$
21     001210 000407      BR     66$
22     001212 022626      64$:  CMP    (SP)+, (SP)+    ; ADJUST STACK
23     001214 012767 000176 014404  65$:  MOV    #SWREG, SWR     ; POINT TO SOFTWARE SWITCH REG
24     001222 012767 000174 014400  MOV    #DISPREG, DISPLAY ; POINT TO SOFT DISPLAY REG
25     001230 012637 000004      66$:  MOV    (SP)+, @#4      ; RESTORE VECTORS
26     001234 012637 000006      MOV    (SP)+, @#6
27     001240 012777 000100 014350  MOV    #INTENA, @TKCSR  ; ENABLE TELETYPE INTERRUPTS
28     001246 005067 000040      CLR    XFLAG           ; XOR = NO
29
30     ;*****
31     ;REPLACE THE FOLLOWING BRANCH WITH A "NOP" (240) TO ACTIVATE "XOR" CODE
32     ;*****
33     001252 000423      BR     STARTO          ; SKIP XOR STUFF
34     001254 016746 176524      MOV    4, -(SP) ;SAVE 4
35     001260 012767 001314 176516  MOV    #XORSVC, 4      ; SET UP SVC ROUTINE
36     001266 005767 175566      TST    177060          ; GOT AN XOR TESTER OUT THERE ?
37     001272 012667 176506      MOV    (SP)+, 4        ; YES
38     001276 005167 000010      COM    XFLAG           ; XOR = YES
39     001302 004767 014020      JSR    PC, XOR         ; AUTO VECTOR
40     001306 000167 000010      JMP    STARTO          ; RESTORE TRAPCATCHER
41     001312 000000      XFLAG: 0              ; XOR FLAG
42     001314 022626      XORSVC: POP2SP
43     001316 012667 176462      MOV    (SP)+, 4 ;RESTORE 4
44     001322 005767 014370      STARTO: TST    TIFLG    ; TYPED TITLE?
45     001326 001005      BNE    .+14           ; YES
46     001330 104004      TYPE    "MODEM CONTROL DIAGNOSTIC"
47     001332 016575      MTITLE
48     001334 012767 000001 014354  MOV    #1, TIFLG      ; SET TITLE TYPED FLAG
49     001342 005767 177744      TST    XFLAG          ; X OR ?
50     001346 100422      BMI    VECSTR         ; RESTORE TRAPCATCHER
51     001350 005767 176466      TST    42             ; ACT 11?
52     001354 001403      BEQ    START1         ; NO
53     001356 004767 013744      JSR    PC, XOR        ; YES AUTO VECTOR
54     001362 000414      BR     VECSTR         ; GET VECTOR AND REGISTER ADDRESS
55     001364 005737 000042      START1: TST    @#42    ; UNDER MONITOR?
56     001370 001005      BNE    1$
57     001372 022767 000176 014226  CMP    #SWREG, SWR    ; USING SWREG?
58     001400 001001      BNE    1$

```



```

58 001402 104025          CNTLUU
59 001404 032777 000001 014214 1$: BIT #1,@SWR ;IF SW BIT 0=1, ON PROGRAM RESTART
60 001412 001510          BEQ STARTN ;INPUT VECTOR AND REGISTER ADDRESSES
61 001414 012706 001100          VECSTR: MOV #STACK,SP ;SET UP PROCESSOR STACK POINTER
62 001420 012767 000300 011324 MOV #300,DATA1 ;ADDRESS OF FIRST FLOATING VECTOR
63 001426 012767 000302 011320 MOV #302,DATA2 ;ADDRESS OF STATUS WORD
64 001434 016777 011314 011310 VECSTA: MOV DATA2,@DATA1 ;MOVE ADDRESS OF STATUS WORD TO VECTOR
65 001442 005077 011306          CLR @DATA2 ;CLEAR STATUS WORD
66                                     ;(FOR HALT ON ILLEGAL INTERRUPT)
67 001446 062767 000004 011276 ADD #4,DATA1 ;NEXT VECTOR
68 001454 062767 000004 011272 ADD #4,DATA2 ;NEXT STATUS WORD
69 001462 026727 011264 001000 CMP DATA1,#1000 ;IS TABLE CLEARED
70 C,1470 001361          BNE VECSTA ;IF NOT, CONTINUE
71 001472 005767 177614          TST XFLAG ;XOR ?
72 001476 100523          BMI TSTGO ;YES
73 001500 005767 176336          TST 42 ;ACT 11 ?
74 001504 001120          BNE TSTGO ;YES
75 001506 104013          INSTRG ;GET VECTOR ADDRESS
76 001510 016661          MVECTOR ;MESSAGE "VECTOR ADDRESS-"
77 001512 000300          300 ;LOWER LIMIT FOR ADDRESS
78 001514 000774          774 ;UPPER LIMIT FOR ADDRESS
79 001516 015606          DHMVEC ;STORAGE FOR ADDRESS
80 001520 032767 000003 014060 1$: BIT #3,DHMVEC ;TEST 2 LSB OF ADDRESS
81 001526 001404          BEQ VECST1 ;IF 0, CONTINUE
82 001530 012716 001520          MOV #1$, (SP)
83 001534 000167 013274          JMP INSTER ;INCORRECT ADDRESS, TRY AGAIN
84 001540 016767 014042 014042 VECST1: MOV DHMVEC,DHMLVL ;GENERATE ADDRESS OF
85 001546 062767 000002 014034 ADD #2,DHMLVL ;INTERRUPT STATUS WORD
86 001554 104013          INSTRG ;GET ADDRESS OF CONTROL REGISTER
87 001556 016703          MREGAD ;MESSAGE "REGISTER ADDRESS-"
88 001560 170500          170500 ;LOWER LIMIT FOR ADDRESS
89 001562 177777          177777 ;UPPER LIMIT FOR ADDRESS
90 001564 015612          DHMCSR ;STORAGE FOR ADDRESS
91 001566 032767 000007 014016 1$: BIT #7,DHMCSR ;IF 3 LSB ARE NOT 0
92 001574 001404          BEQ REGST1
93 001576 012716 001566          MOV #1$, (SP)
94 001602 000167 013226          JMP INSTER ;INCORRECT ADDRESS, TRY AGAIN
95 001606 016767 014000 014000 REGST1: MOV DHMCSR,DHMLSR ;SET UP ADDRESS OF LINE STATUS REGISTER
96 001614 062767 000002 013772 ADD #2,DHMLSR
97 001622 104013          INSTRG ;GET LINE SELECT PARAMETER
98 001624 016737          MLINSL
99 001626 000000          0
100 001630 177777          177777
101 001632 015720          LINSEL

```

```

1
2 001634 012706 001100      STARTN: MOV      #STACK,SP      ;SET UP PROCESSOR STACK
3 001640 104013              INSTRG      ;GET TEST NUMBER
4 001642 016771              MTEST      ;MESSAGE "TEST-"
5 001644 000000              0          ;LOWER LIMIT FOR TEST NUMBER
6 001646 000777              777        ;UPPER LIMIT FOR TEST NUMBER
7 001650 015640              TSTNO      ;STORAGE FOR TEST NUMBER
8 001652 016705 013762      X1A:  MOV      TSTNO,R5      ;GET TEST NUMBER
9 001656 042705 177077      BIC      #177077,R5      ;EXTRACT TEST GROUP NUMBER
10
11      .REPT      5
12      ASR      R5
      .ENDR
      ASR      R5
      ASR      R5
      ASR      R5
      ASR      R5
      ASR      R5
13 001674 016567 017532 013772      MOV      GRO(R5),TSTMAX      ;GET HIGHEST TEST IN GROUP
14 001702 016567 017512 013762      MOV      TSTLST(R5),TSTPNT      ;GET POINTER TO TEST TABLE
15 001710 005767 013756              TST      TSTPNT      ;IF 0, INVALID TEST GROUP
16 001714 001004              BNE      STRTOA
17 001716 012716 001652      X1B:  MOV      #X1A,(SP)
18 001722 000167 013106              JMP      INSTER      ;TRY AGAIN
19 001726 042767 177700 013704      STRTOA: BIC      #177700,TSTNO      ;GET NUMBER OF FIRST TEST
20                                          ;TO BE EXECUTED IN SELECTED GROUP
21 001734 026767 013700 013732      CMP      TSTNO,TSTMAX      ;IS NUMBER TOO LARGE
22 001742 003401              BLE
23 001744 000764              BR      X1B
24 001746 012746 000340      TSTGO: MOV      #340,-(SP)      ;SET UP PRIORITY LEVEL
25 001752 005746              PUSH1SP
26 001754 000005              RESET
27 001756 012767 002242 000260      MOV      #DMYRTI,KRET      ;SET UP DUMMY KEYBOARD RETURN
28 001764 005067 013706              CLR      LINFLG      ;CLEAR LINE SELECTED FLAG
29 001770 005067 013640              CLR      TRACON      ;CLEAR TRACE TRAP FLAG
30 001774 005067 013636              CLR      PASCNT      ;CLEAR PASS COUNT
31 002000 104004              TYPE
32 002002 017005              MCRLF
33 002004 012767 000001 000004      1$:  MOV      #1,TIPFLG      ;SET TEST IN PROGRESS FLAG
34 002012 000167 011250              JMP      TSTENT      ;START TESTING
35 002016 000000      TIPFLG: 0
36
37
38
39
40      ;TELETYPE KEYBOARD INTERRUPT SERVICE ROUTINE
41
42 002020 005067 177772      KBDINT: CLR      TIPFLG      ;CLEAR TEST IN PROGRESS FLAG
43 002024 005067 012232              CLR      TMP1
44 002030 005067 000212              CLR      SINTFL      ;CLEAR SOFTWARE INTERRUPT FLAG
45 002034 117767 013560 012220      MOV      @TKDBR,TMP1
46 002042 142767 000200 012212      BIC      #200,TMP1
47 002050 122767 000003 012204      CMP      #3,TMP1
48 002056 001011              BNE      KBDINI      ;IF <CTRL C> WAS TYPED
49 002060 104004              TYPE      ;TYPE "+C" AND
50 002062 017235              MCONTC      ;SELECT NEW TEST
51 002064 022626              POP2SP
52 002066 005077 013520              CLR      @DHMCSR

```



```

002250
302 002250 000001
303 002252 104004
304 002254 016542
305 002254 005777 013332
306 002260 001401
307 002262 104012
308 002264 005777 013324
309 002270 001401
310 002272 104012
311 002274 104002
312
313
314
315 002276
    002276
316 002276 000002
317 002304 012777 000100 013306
318 002304 032777 000100 013300
319 002312 001001
320 002314 104012
321 002316 042777 000100 013266
322 002324 032777 000100 013260
323 002332 001401
324 002334 104012
325 002336 104002
326
327
328 002340
    002340
329 002340 000003
330 002346 012777 000200 013244
331 002346 032777 000200 013236
332 002354 001001
333 002356 104012
334 002360 042777 000200 013224
335 002366 032777 000200 013216
336 002374 001401
337 002376 104012
338 002400 104002

TO:
NO=NO+1
;REFERENCE DESIGNATION

INIT1:
TYPE
M16
;TYPE "16 LINE SCANNER TEST"
TST @DHMCSR
;TEST CONTROL STATUS REGISTER
BEQ .+4
ERROR
;CONTROL STATUS NOT CLEARED, ERROR
TST @DHMLSR
;TEST LINE STATUS REGISTER
BEQ .+4
ERROR
;LINE STATUS NOT CLEARED, ERROR
SCOPE
;CHECK FOR LOOP

;VERIFY THAT "INTERRUPT ENABLE" CAN BE
;SET AND CLEARED.

TS \NO,0
;REFERANCE NUMBER DEFINITION
T1:
NO=NO+1
;REFERENCE DESIGNATION
CSTR1: MOV #INTENA,@DHMCSR
;SET INTERRUPT ENABLE
BIT #INTENA,@DHMCSR
;WAS INTERRUPT ENABLE SET
BNE .+4
ERROR
;NO, ERROR
BIC #INTENA,@DHMCSR
;CLEAR INTERRUPT ENABLE
BIT #INTENA,@DHMCSR
;WAS INTERRUPT ENABLE CLEARED
BEQ .+4
ERROR
;NO, ERROR
SCOPE
;CHECK FOR ITERATIONS, LOOP

;VERIFY THAT "DONE" CAN BE SET AND CLEARED

TS \NO,0
;REFERANCE NUMBER DEFINITION
T2:
NO=NO+1
;REFERENCE DESIGNATION
CSTR2: MOV #DONE,@DHMCSR
;SET DONE
BIT #DONE,@DHMCSR
;WAS DONE SET
BNE .+4
ERROR
;NO, ERROR
BIC #DONE,@DHMCSR
;CLEAR DONE
BIT #DONE,@DHMCSR
;WAS DONE CLEARED
BEQ .+4
ERROR
;NO, ERROR
SCOPE
;CHECK FOR ITERATIONS, LOOP

```

338
339
340

;VERIFY "MAINTENANCE MODE" CAN BE SET AND CLEARED

341 002402

TS \NO,0

;REFERANCE NUMBER DEFINITION

002402

T3:

;REFERENCE DESIGNATION

NO=NO+1

342 002402 012777 001000 013202
343 002410 032777 001000 013174
344 002416 001001
345 002420 104012
346 002422 042777 001000 013162
347 002430 032777 001000 013154
348 002436 001401
349 002440 104012
350 002442 104002

CSTR3:

MOV #MAINT,@DHMCSR
BIT #MAINT,@DHMCSR
BNE .+4
ERROR
BIC #MAINT,@DHMCSR
BIT #MAINT,@DHMCSR
BEQ .+4
ERROR
SCOPE

;SET MAINTENANCE MODE
;WAS MAINTENANCE MODE SET

;NO, ERROR
;CLEAR MAINTENANCE MODE
;WAS MAINTENANCE MODE CLEARED

;NO, ERROR
;CHECK FOR ITERATIONS, LOOP

```

1
2
3
4 002444      TS \NO,0
                ;REFERENCE NUMBER DEFINITION
                T4:
                NO=NO+1
                ;REFERENCE DESIGNATION
5 002444 012777 000040 013140 CSTR4: MOV #SCNENA,@DHMCSR ;SET SCAN ENABLE
6 002452 032777 000040 013132   BIT #SCNENA,@DHMCSR ;WAS SCAN ENABLE SET
7 002460 001001
8
9 002462 104012   ERROR ;NO, ERROR
10 002464 042777 000040 013120  BIC #SCNENA,@DHMCSR ;CLEAR SCAN ENABLE
11 002472 032777 000040 013112  BIT #SCNENA,@DHMCSR ;WAS SCAN ENABLE CLEARED
12 002500 001401   BEQ .+4
13
14 002502 104012   ERROR ;NO, ERROR
15 002504 104002   SCOPE ;CHECK FOR ITERATIONS, LOOP
16
17
18 ;VERIFY THAT "BUSY" IS SET WHEN "SCAN ENABLE" IS SET
19 ;VERIFY THAT "BUSY" IS CLEARED WHEN "SCAN ENABLE" IS CLEARED
20 002506      TS \NO,0
                ;REFERENCE NUMBER DEFINITION
                T5:
                NO=NO+1
                ;REFERENCE DESIGNATION
21 002506 012777 000040 013076 CSTR5: MOV #SCNENA,@DHMCSR ;SET SCAN ENABLE
22 002514 032777 000020 013070   BIT #BUSY,@DHMCSR ;IS BUSY BIT SET
23 002522 001001   BNE .+4
24 002524 104012   ERROR ;BUSY NOT SET, ERROR
25 002526 042777 000040 013056  BIC #SCNENA,@DHMCSR ;CLEAR SCAN ENABLE
26 002534 032777 000020 013050  BIT #BUSY,@DHMCSR ;IS BUSY BIT CLEARED
27 002542 001401   BEQ .+4
28 002544 104012   ERROR ;BUSY NOT CLEARED, ERROR
29 002546 104002   SCOPE ;CHECK FOR LOOP, ITERATIONS
30
31
32 ;VERIFY THAT SETTING "DONE" DOES NOT CCAUSE AN
33 ;INTERRUPT IF "INTERRUPT ENABLE" IS CLEARED.
34 002550      TS \NO,0
                ;REFERENCE NUMBER DEFINITION
                T6:
                NO=NO+1
                ;REFERENCE DESIGNATION
35 002550 052767 000340 175220 INT1: BIS #340,PS ;LOCK OUT INTERRUPTS
36 002556 005077 013030   CLR @DHMCSR ;CLEAR CONTROL REGISTER
37 002562 012777 002616 013016   MOV #INT1A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
38 002570 016777 175202 013012   MOV PS,@DHMLVL ;SET UP INTERRUPT PRIORITY
39 002576 052777 000200 013006   BIS #DONE,@DHMCSR ;SET DONE
40 002604 042767 000340 175164   BIC #340,PS ;ALLOW INTERRUPTS
41 002612 000240   NOP ;DELAY FOR INTERRUPT
42 002614 000402   BR INT1B ;NO INTERRUPT, CONTINUE
43 002616 022626 INT1A: POP2SP ;RESTORE STACK, INTERRUPT
44 002620 104012   ERROR ;OCCURED, ERROR
45 002622 104002 INT1B: SCOPE ;CHECK FOR LOOP, ITERATIONS

```

```

1
2
3
4
5 002624
    002624
        000010
6 002624 052767 000340 175144
7 002632 005077 012754
8 002636 012777 002672 012742
9 002644 016777 175126 012736
10 002652 052777 000100 012732
11 002660 042767 000340 175110
12 002666 000240
13 002670 000402
14 002672 022626
15 002674 104012
16 002676 104002
17
18
19
20
21 002700
    002700
        000011
22 002700 052767 000340 175070
23 002706 005077 012700
24 002712 012777 002764 012666
25 002720 012777 000100 012664
26 002726 016777 175044 012654
27 002734 042767 000340 175034
28 002742 052777 000200 012642
29 002750 000240
30 002752 000240
31 002754 005077 012632
32 002760 104012
33 002762 000401
34 002764 022626
35 002766 104002
36
37
38
39
40
41
42
43
44 002770
    000340
    000007
    000004
    000004
45
46
47
48

;VERIFY THAT NO INTERRUPT OCCURS WITH "INTERRUPT ENABLE"
;SET AND "DONE" CLEARED.

TS \NO,0
;REFERANCE NUMBER DEFINITION
T7:
NO=NO+1
;REFERENCE DESIGNATION
INT2: BIS #340,PS ;LOCK OUT INTERRUPTS
      CLR @DHMCSR ;CLEAR CONTROL REGISTER
      MOV #INT2A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
      MOV PS,@DHMLVL ;SET UP INTERRUPT SERVICE LEVEL
      BIS #INTENA,@DHMCSR ;SET INTERRUPT ENABLE
      BIC #340,PS ;ALLOW INTERRUPTS
      NOP ;DELAY FOR INTERRUPTS
      BR INT2B ;NO INTERRUPT, CONTINUE
INT2A: POP2SP ;RESTORE STACK
      ERROR ;INTERRUPT OCCURED, ERROR
INT2B: SCOPE ;CHECK FOR ITERATIONS, LOOP

;VERIFY THAT SETTING "DONE" CAUSES AN INTERRUPT
;WITH "INTERRUPT ENABLE" SET

TS \NO,0
;REFERANCE NUMBER DEFINITION
T10:
NO=NO+1
;REFERENCE DESIGNATION
INT3: BIS #340,PS ;LOCK OUT INTERRUPTS
      CLR @DHMCSR ;CLEAR CONTROL REGISTER
      MOV #INT3A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
      MOV #INTENA,@DHMCSR ;SET "INTERRUPT ENABLE"
      MOV PS,@DHMLVL ;SET "INTERRUPT LEVEL"
      BIC #340,PS ;ALLOW INTERRUPTS
      BIS #DONE,@DHMCSR ;SET "DONE"
      NOP ;DELAY FOR INTERRUPT
      NOP
      CLR @DHMCSR
      ERROR ;INTERRUPT OCCURED, ERROR
      BR INT3B ;CONTINUE
INT3A: POP2SP ;INTERRUPT OCCURED, RESTOR STACK
INT3B: SCOPE ;CHECK FOR ITERATION, LOOP

.MACRO COMMENT
.NLIST
ST=340
LVL=7
T=4

.LIST
.ENDM

COMMENT
ST=340
LVL=7
T=4

.REPT 4
NOINT \ST,\LVL,\T

.NLIST
ST=ST-40

```

49
50
51
52
53
54

002770

```
LVL=LVL-1
T=T+1
.LIST
.IIF EQ NO-12,.PAGE
.IIF EQ NO-15,.PAGE
.ENDR
NOINT \ST,\LVL,\T
```

```
;VERIFY THAT NO INTERRUPT OCCURS WITH
;"INTERRUPT ENABLE" SET AND "DONE" SET AT PRIORITY 7.
```

002770

```
TS \NO,0
;REFERANCE NUMBER DEFINITION
```

002770

```
T11:
NO=NO+1
INT4: CLR @DHMCSR ;REFERENCE DESIGNATION
```

```
000012
002770 005077 012616
002774 042767 000340 174774
003002 052767 000340 174766
003010 012777 003052 012570
003016 016777 174754 012564
003024 012777 000100 012560
003032 052777 000200 012552
003040 000240
003042 000240
003044 005077 012542
003050 000402
003052 022626
003054 104012
003056 104002
000300
000006
000005
```

```
INT4: CLR @DHMCSR ;CLEAR CONTROL REGISTER
BIC #340,PS ;SET PROCESSOR PRIORITY
BIS #340,PS ;TO LEVEL 7.
MOV #INT4A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
MOV PS,@DHMLVL ;SET UP INTERRUPT SERVICE LEVEL
MOV #INTENA,@DHMCSR ;SET INTERRUPT ENABLE
BIS #DONE,@DHMCSR ;GENERATE INTERRUPT
NOP ;DELAY FOR INTERRUPT
NOP
CLR @DHMCSR
BR INT4B ;NO INTERRUPT, CONTINUE
INT4A: POP2SP ;RESTORE STACK
ERROR ;INTERRUPT OCCURED, ERROR
INT4B: SCOPE ;CHECK FOR ITERATION, LOOP
ST=ST-40
LVL=LVL-1
T=T+1
```



```

0
003060      .IIF EQ NO 15,.PAGE
              NOINT  \ST,\LVL,\T

              ;VERIFY THAT NO INTERRUPT OCCURS WITH
              ;"INTERRUPT ENABLE" SET AND "DONE" SET AT PRIORITY 6.

003060      TS \NO,0
              ;REFERANCE NUMBER DEFINITION

003060      T12:
              NO=NO-1      ;REFERENCE DESIGNATION
INT5:      CLR      @DHMCSR      ;CLEAR CONTROL REGISTER
              BIC      @340,PS   ;SET PROCESSOR PRIORITY
              BIS      @300,PS   ;TO LEVEL 6.
003064      042767 000340 174704      MOV      @INT5A,@DHMVEC      ;SET UP INTERRUPT SERVICE ADDRESS
003072      052767 000300 174676      MOV      PS,@DHMLVL      ;SET UP INTERRUPT SERVICE LEVEL
003100      012777 003142 012500      MOV      @INTENA,@DHMCSR      ;SET INTERRUPT ENABLE
003106      016777 174664 012474      BIS      @DONE,@DHMCSR      ;GENERATE INTERRUPT
003114      012777 000100 012470      NOP      ;DELAY FOR INTERRUPT
003122      052777 000200 012462      NOP
003130      000240
003132      000240
003134      005077 012452      CLR      @DHMCSR
003140      000402      BR      INT5B
003142      022626      INT5A:  POP2SP      ;NO INTERRUPT, CONTINUE
003144      104012      ERROR      ;RESTORE STACK
003146      104002      INT5B:  SCOPE      ;INTERRUPT OCCURED, ERROR
              000240      ST=ST-40      ;CHECK FOR ITERATION, LOOP
              000005      LVL=LVL-1
              000006      T=T+1

.IIF EQ NO 12,.PAGE
.IIF EQ NO-15,.PAGE
003150      NOINT  \ST,\LVL,\T

              ;VERIFY THAT NO INTERRUPT OCCURS WITH
              ;"INTERRUPT ENABLE" SET AND "DONE" SET AT PRIORITY 5.

003150      TS \NO,0
              ;REFERANCE NUMBER DEFINITION

003150      T13:
              NO=NO-1      ;REFERENCE DESIGNATION
INT6:      CLR      @DHMCSR      ;CLEAR CONTROL REGISTER
              BIC      @340,PS   ;SET PROCESSOR PRIORITY
              BIS      @240,PS   ;TO LEVEL 5.
003150      005077 012436      MOV      @INT6A,@DHMVEC      ;SET UP INTERRUPT SERVICE ADDRESS
003154      042767 000340 174614      MOV      PS,@DHMLVL      ;SET UP INTERRUPT SERVICE LEVEL
003162      052767 000240 174606      MOV      @INTENA,@DHMCSR      ;SET INTERRUPT ENABLE
003170      012777 003232 012410      BIS      @DONE,@DHMCSR      ;GENERATE INTERRUPT
003176      016777 174574 012404      NOP      ;DELAY FOR INTERRUPT
003204      012777 000100 012400      NOP
003212      052777 000200 012372      NOP
003220      000240
003222      000240
003224      005077 012362      CLR      @DHMCSR
003230      000402      BR      INT6B
003232      022626      INT6A:  POP2SP      ;NO INTERRUPT, CONTINUE
003234      104012      ERROR      ;RESTORE STACK
003236      104002      INT6B:  SCOPE      ;INTERRUPT OCCURED, ERROR
              000200      ST=ST-40      ;CHECK FOR ITERATION, LOOP
              000004      LVL=LVL-1
              000007      T=T+1

.IIF EQ NO-12,.PAGE
.IIF EQ NO-15,.PAGE

```

003240

NOINT \ST,\LVL,\T

;VERIFY THAT NO INTERRUPT OCCURS WITH
; "INTERRUPT ENABLE" SET AND "DONE" SET AT PRIORITY 4.

003240

TS \NO,0

;REFERENCE NUMBER DEFINITION

003240

000015

T14:

;REFERENCE DESIGNATION

003240

005077

012346

NO=NO-1

003244

042767

000340

174524

INT7:

CLR

@DHMCSR

;CLEAR CONTROL REGISTER

003252

052767

000200

174516

BIC

#340,PS

;SET PROCESSOR PRIORITY

003260

012777

003322

012320

BIS

#200,PS

;TO LEVEL 4.

003266

016777

174504

012314

MOV

#INT7A,@DHMVEC

;SET UP INTERRUPT SERVICE ADDRESS

003274

012777

000100

012310

MOV

PS,@DHMLVL

;SET UP INTERRUPT SERVICE LEVEL

003302

052777

000200

012302

MOV

#INTENA,@DHMCSR

;SET INTERRUPT ENABLE

003310

000240

BIS

#DONE,@DHMCSR

;GENERATE INTERRUPT

003312

000240

NOP

;DELAY FOR INTERRUPT

003314

005077

012272

NOP

003320

000402

CLR

@DHMCSR

003322

022626

INT7A:

BR

INT7B

;NO INTERRUPT, CONTINUE

003324

104012

POP2SP

;RESTORE STACK

003326

104002

INT7B:

ERROR

;INTERRUPT OCCURED, ERROR

000140

SCOPE

;CHECK FOR ITERATION, LOOP

000003

ST=ST-40

000010

LVL=LVL 1

T=T+1

.IIF EQ NO-12..PAGE

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18

003330

000000
000000
000010
000004

```
.MACRO COMMENT
.NLIST
    ST=0
    LVL=0
    T=10
.LIST
.ENDM
COMMENT
ST=0
LVL=0
T=10
.REPT 4
INTS \ST,\LVL,\T
.NLIST
    ST=ST+40
    LVL=LVL+1
    T=T+1
.LIST
.IIF EQ NO-20,.PAGE
.ENDR
INTS \ST,\LVL,\T
```

;VERIFY THAT AN INTERRUPT OCCURS WITH "INTERRUPT
;ENABLE" SET AND "DONE" SET AT PRIORITY 0.

003330

003330

000016
003330 005077 012256
003334 042767 000340 174434
003342 012777 003412 012236
003350 005077 012234
003354 052767 000000 174414
003362 012777 000100 012222
003370 052777 000200 012214
003376 000240
003400 000240
003402 005077 012204
003406 104012
003410 000401
003412 022626
003414 104002
000040
000001
000011

```
TS \NO,0
;REFERANCE NUMBER DEFINITION
T15:
NO=NO+1
;REFERENCE DESIGNATION
INT10: CLR @DHMCSR ;CLEAR CONTROL REGISTER
      BIC #340,PS ;ALLOW INTERRUPTS
      MOV @INT10A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
      CLR @DHMLVL ;SET UP INTERRUPT SERVICE PRIORITY
      BIS #0,PS ;SET PROCESSOR PRIORITY TO LEVEL 0.
      MOV @INTENA,@DHMCSR ;SET INTERRUPT ENABLE
      BIS #DONE,@DHMCSR ;GENERATE INTERRUPT
      NOP ;WAIT FOR INTERRUPT
      NOP
      CLR @DHMCSR
      ERROR ;NO INTERRUPT, ERROR
      BR INT10B ;CONTINUE
INT10A: POP2SP ;INTERRUPT OCCURED, RESTORE STACK
INT10B: SCOPE ;CHECK FOR INTERATIONS, LOOP.
      ST=ST+40
      LVL=LVL+1
      T=T+1
```

003416

```
.IIF EQ NO-20,.PAGE
INTS \ST,\LVL,\T
```

;VERIFY THAT AN INTERRUPT OCCURS WITH "INTERRUPT
;ENABLE" SET AND "DONE" SET AT PRIORITY 1.

003416

003416

000017

```
TS \NO,0
;REFERANCE NUMBER DEFINITION
T16:
NO=NO+1
;REFERENCE DESIGNATION
```

```

003416 005077 012170          INT11: CLR      @DHMCSR          ;CLEAR CONTROL REGISTER
003422 042767 000340 174346    BIC      #340,PS          ;ALLOW INTERRUPTS
003430 012777 003500 012150    MOV      @INT11A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
003436 005077 012146          CLR      @DHMLVL         ;SET UP INTERRUPT SERVICE PRIORITY
003442 052767 000040 174326    BIS      #40,PS          ;SET PROCESSOR PRIORITY TO LEVEL 1.
003450 012777 000100 012134    MOV      @INTENA,@DHMCSR ;SET INTERRUPT ENABLE
003456 052777 000200 012126    BIS      @DONE,@DHMCSR   ;GENERATE INTERRUPT
003464 000240          NOP                       ;WAIT FOR INTERRUPT
003466 000240          NOP
003470 005077 012116          CLR      @DHMCSR
003474 104012          ERROR                    ;NO INTERRUPT, ERROR
003476 000401          BR      INT11B           ;CONTINUE
003500 022626          INT11A: POP2SP           ;INTERRUPT OCCURED, RESTORE STACK
003502 104002          INT11B: SCOPE           ;CHECK FOR INTERATIONS, LOOP.
000100          ST=ST+40
000002          LVL=LVL+1
000012          T=T+1

003504          .IIF EQ NO-20,.PAGE
          INTS      \ST,\LVL,\T

          ;VERIFY THAT AN INTERRUPT OCCURS WITH "INTERRUPT
          ;ENABLE" SET AND "DONE" SET AT PRIORITY 2.

003504          TS \NO,0
          ;REFERANCE NUMBER DEFINITION

003504          T17:
          NO=NO+1          ;REFERENCE DESIGNATION

003504 000020          INT12: CLR      @DHMCSR          ;CLEAR CONTROL REGISTER
003510 042767 000340 174260    BIC      #340,PS          ;ALLOW INTERRUPTS
003516 012777 003566 012062    MOV      @INT12A,@DHMVEC ;SET UP INTERRUPT SERVICE ADDRESS
003524 005077 012060          CLR      @DHMLVL         ;SET UP INTERRUPT SERVICE PRIORITY
003530 052767 000100 174240    BIS      #100,PS         ;SET PROCESSOR PRIORITY TO LEVEL 2.
003536 012777 000100 012046    MOV      @INTENA,@DHMCSR ;SET INTERRUPT ENABLE
003544 052777 000200 012040    BIS      @DONE,@DHMCSR   ;GENERATE INTERRUPT
003552 000240          NOP                       ;WAIT FOR INTERRUPT
003554 000240          NOP
003556 005077 012030          CLR      @DHMCSR
003562 104012          ERROR                    ;NO INTERRUPT, ERROR
003564 000401          BR      INT12B           ;CONTINUE
003566 022626          INT12A: POP2SP           ;INTERRUPT OCCURED, RESTORE STACK
003570 104002          INT12B: SCOPE           ;CHECK FOR INTERATIONS, LOOP.
000140          ST=ST+40
000003          LVL=LVL+1
000013          T=T+1
    
```

0 003572

INTS \ST,\LVL,\T

;VERIFY THAT AN INTERRUPT OCCURS WITH "INTERRUPT
;ENABLE" SET AND "DONE" SET AT PRIORITY 3.

003572

TS \NO,0

;REFERANCE NUMBER DEFINITION

003572

T20:

;REFERENCE DESIGNATION

000021

NO=NO+1

003572 005077 012014

INT13:

CLR @DHMCSR

;CLEAR CONTROL REGISTER

003576 042767 000340 174172

BIC #340,PS

;ALLOW INTERRUPTS

003604 012777 003654 011774

MOV #INT13A,@DHMVEC

;SET UP INTERRUPT SERVICE ADDRESS

003612 005077 011772

CLR @DHMLVL

;SET UP INTERRUPT SERVICE PRIORITY

003616 052767 000140 174152

BIS #140,PS

;SET PROCESSOR PRIORITY TO LEVEL 3.

003624 012777 000100 011760

MOV #INTENA,@DHMCSR

;SET INTERRUPT ENABLE

003632 052777 000200 011752

BIS #DONE,@DHMCSR

;GENERATE INTERRUPT

003640 000240

NOP

;WAIT FOR INTERRUPT

003642 000240

NOP

003644 005077 011742

CLR @DHMCSR

003650 104012

ERROR

;NO INTERRUPT, ERROR

003652 000401

BR INT13B

;CONTINUE

003654 022626

INT13A: POP2SP

;INTERRUPT OCCURED, RESTORE STACK

003656 104002

INT13B: SCOPE

;CHECK FOR INTERATIONS, LOOP.

000200

ST=ST+40

000004

LVL=LVL+1

000014

T=T+1

.IIF EQ NO-20,.PAGE

```

1
2
3
4
5 003660
    003660
        000022
6 003660 005077 011726
7 003664 042767 000340 174104
8 003672 012767 000001 012022
9 003700 005005
10 003702 012700 000020
11 003706 036767 012010 012004
12 003714 001407
13 003716 010577 011670
14 003722 017704 011664
15 003726 020504
16 003730 001401
17 003732 104000
18 003734 104003
19 003736 003706
20 003740 005205
21 003742 006367 011754
22 003746 005300
23 003750 001356
24 003752 104002
25
26
27
28
29 003754
    003754
        000023
30 003754 042767 000340 174014
31 003762 005077 011624
32 003766 005005
33 003770 012767 000001 011724
34 003776 012701 177777
35 004002 012700 000020
36 004006 012777 000017 011576
37 004014 036767 011702 011676
38 004022 001407
39 004024 004767 007354
40 004030 017704 011556
41 004034 020504
42 004036 001401
43 004040 104000

;VERIFY THAT ALL LINE NUMBERS CAN BE WRITTEN INTO AND
;READ BACK FROM LINE COUNTER

TS \NO,0
;REFERANCE NUMBER DEFINITION
T21:
NO=NO+1
;REFERENCE DESIGNATION
LINT1: CLR @DHMCSR ;CLEAR CONTROL STATUS REGISTER
        BIC #340,PS ;ENABLE INTERRUPTS
        MOV #1,SELSK ;INIT LINE SELECT MASK
        CLR R5 ;CLEAR EXPECTED LINE NUMBER
        MOV #16.,R0 ;SET UP TO TEST 16 LINE NUMBERS
LINT1A: BIT SELSK,LINSEL ;THIS LINE SELECTED ??
        BEQ LINT1B ;BR IF NOT
        MOV R5,@DHMCSR ;SET LINE NUMBER
        MOV @DHMCSR,R4 ;READ BACK LINE NUMBER
        CMP R5,R4 ;ARE EXPECTED AND RECEIVED
        BEQ LINT1B ;LINE NUMBERS THE SAME
        ERRORC ;LINE NUMBERS DIFFERENT, ERROR
LINT1B: SCOPEF ;CHECK FOR DATA FREEZE
        LINT1A ;RETURN FOR DATA FREEZE
        INC R5 ;UPDATE LINE COUNT
        ASL SELSK ;SELECT NEXT LINE TO TEST
        DEC R0 ;UPDATE LINE NUMBER
        BNE LINT1A ;CONTINUE
        SCOPE ;CHECK FOR ITERATION, LOOP

;USING "STEP" MODE, VERIFY THAT THE
;LINE COUNTER CAN BE STEPPED THRU ALL STATES.

TS \NO,0
;REFERANCE NUMBER DEFINITION
T22:
NO=NO+1
;REFERENCE DESIGNATION
LINT2: BIC #340,PS ;ENABLE INTERRUPTS
        CLR @DHMCSR ;CLEAR CONTROL STATUS REGISTER
        CLR R5 ;CLEAR EXPECTED LINE COUNT
        MOV #1,SELSK ;SET UP SELECT MASK
        MOV #-1,R1 ;INIT LINE COUNTER
        MOV #16.,R0 ;SET UP TO TEST 16 VALUES
        MOV #17,@DHMCSR ;FIRST VALUE =0
LINT2A: BIT SELSK,LINSEL ;THIS LINE SELECTED ??
        BEQ LINT2B ;BR IF NOT
        CALL STEPER ;STEP LINE COUNTER
        MOV @DHMCSR,R4 ;READ LINE COUNTER
        CMP R5,R4 ;COMPARE EXPECTED AND
        BEQ LINT2B ;RECEIVED LINE NUMBERS
        ERRORC ;LINE COUNTER ERROR

```

44 004042 104003
45 004044 003754
46 004046 005205
47 004050 006367 011646
48 004054 005201
49 004056 010177 011530
50 004062 005300
51 004064 001353
52 004066 104002

LINT2B: SCOPEF
LINT2
INC R5
ASL SELMSK
INC R1
MOV R1, @DHMCSR
DEC R0
BNE LINT2A
SCOPE

;CHECK FOR DATA FREEZE
;UPDATE EXPECTED LINE NUMBER
;SHIFT SELECT MASK
;GEN NEW LINE NO.
;SET NEW LINE NO. IN CSR

;CHECK FOR ITERATIONS, LOOP

```

1
2
3
4
5
6
7
8 004070          TS \NO,0
                   ;WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS.
                   ;VERIFY THAT ALL LOCATIONS HAVE BEEN WRITTEN
                   ;TO 1'S.
                   ;VERIFY THAT "CLEAR SCAN" CLEARS ALL SCANNER
                   ;MEMORY LOCATIONS.

004070          T23:
                   NO=NO+1
                   ;REFERENCE NUMBER DEFINITION
                   ;REFERENCE DESIGNATION
9 004070 000024
10 004076 012777 002000 011514 MEMT1: MOV #CLRMUX,@DHMCSR ;CLEAR CONTROL STATUS REGISTER
11 004104 012700 000340 173672 BIC #340,PS ;ENABLE INTERRUPTS
12 004110 052777 001017 011474 MOV #16.,R0 ;SET UP TO TEST 16 LOCATIONS
13 004116 004767 007262 MEMT1A: BIS #MAINT+17,@DHMCSR ;SET MAINTENANCE MODE
14 004122 005300 CALL STEPER ;SET LINE COUNTER THRU ALL
15 004124 001374 DEC R0 ;STATES, WRITING 1'S INTO
16 004126 012700 000020 BNE MEMT1A ;ALL MEMORY WORDS
17 004132 012705 070000 MOV #16.,R0 ;SET UP TO TEST 16 WORDS
18 004136 012777 000017 011446 MOV #70000,R5 ;SET UP EXPECTED STATUS REGISTER
19 004144 004767 007234 MEMT1B: MOV #17,@DHMCSR ;START WITH LINE 0
20 004150 017704 011436 CALL STEPER ;ACCESS SCANNER MEMORY
21 004154 020504 MOV @DHMCSR,R4 ;READ DATA
22 004156 001403 CMP R5,R4 ;COMPARE EXPECTED AND RECEIVED
23 004160 104000 BEQ MEMT1C ;DATA
24 004162 104003 ERRORC ;CONTROL STATUS OR MEMORY ERROR
25 004164 004070 SCOPEF ;CHECK FOR DATA FREEZE
26 004166 005205 MEMT1C: MEMT1
27 004170 005300 INC R5 ;UPDATE EEXPECTED STATUS
28 004172 001364 DEC R0 ;UPDATE LINE COUNT
29 004174 012777 004000 011410 BNE MEMT1B ;CONTINUE
30 004202 032777 000020 011402 MEMT1D: MOV #CLRSCN,@DHMCSR ;SET "CLEAR SCAN"
31 004210 001374 BIT #BUSY,@DHMCSR ;WAIT FOR "CLEAR CYCLES"
32 004212 012700 000020 BNE .-6
33 004216 005005 MOV #16.,R0 ;SET UP TO TEST 16 MEMORY
34 004220 012777 000017 011364 CLR R5 ;LOCATIONS
35 004226 004767 007152 MEMT1E: MOV #17,@DHMCSR ;FIRST TO BE TESTED=0
36 004232 017704 011354 CALL STEPER ;ACCESS SEANNER MEMORY
37 004236 020504 MOV @DHMCSR,R4 ;READ DATA
38 004240 001403 CMP R5,R4 ;COMPARE EXPECTED AND RECEIVED
39 004242 104000 BEQ MEMT1F ;DATA
40 004244 104003 ERRORC ;CONTROL STATUS OF MEMORY ERROR
41 004246 004174 SCOPEFF ;CHECK FOR DATA FREEZE
42 004250 005205 MEMT1F: MEMT1D
43 004252 005300 INC R5 ;UPDATE EXPECTED DATA
44 004254 001364 DEC R0 ;UPDATE LINE COUNT
45 004256 104002 BNE MEMT1E ;CONTINUE
                   SCOPE ;CHECK FOR ITERATIONS, LOOP

```



```

1
2
3
4
5 004260
    004260
    000025
6 004260 005077 011326
7 004264 042767 000340 173504
8 004272 012700 000020
9 004276 012702 000017
10 004302 012777 004000 011302
11 004310 032777 000020 011274
12 004316 001374
13 004320 012777 001000 011264
14 004326 050277 011260
15 004332 004767 007046
16 004336 042777 001000 011246
17 004344 012703 000020
18 004350 012777 000017 011234
19 004356 005202
20 004360 005001
21 004362 004767 007016
22 004366 117704 011220
23 004372 010105
24 004374 120402
25 004376 001002
26 004400 052705 070000
27 004404 020405
28 004406 001403
29 004410 104000
30 004412 104003
31 004414 004302
32 004416 005201
33 004420 005303
34 004422 001357
35 004424 005300
36 004426 001325
37 004430 104002

;WRITE 1'S INTO SELECTED SCANNER MEMORY LOCATION.
;VERIFY THAT ONLY SELECTED LOCATION WAS WRITTEN INTO.

TS \NO.0
;REFERENCE NUMBER DEFINITION

T24:
NO=NO+1
MEMT2: CLR @DHMCSR ;CLEAR CONTROL STATUS REGISTER
        BIC #340,PS ;ENABLE INTERRUPTS
        MOV #16.,R0 ;SET UP TO TEST 16 ADDRESSES
        MOV #17,R2 ;FIRST ADDRESS TO BE TESTED=0
MEMT2A: MOV #CLRSCN,@DHMCSR ;CLEAR ACANNER MEMORY
        BIT #BUSY,@DHMCSR ;WAIT FOR CLEAR CYCLE
        BNE . 6
        MOV #MAINT,@DHMCSR ;SET "MAINTENANCE MODE"
        BIS R2,@DHMCSR ;SET LINE COUNTER TO TEST ADDRESS-1
        CALL STEPER ;WRITE 1'S INTO TEST ADDRESS
        BIC #MAINT,@DHMCSR ;CLEAR "MAINTENANCE MODE"
        MOV #16.,R3 ;SET UP TO TEST ALL 16
        MOV #17,@DHMCSR ;SCANNER MEMORY LOCATIONS
        INC R2
        CLR R1
MEMT2B: CALL STEPER ;ACCESS SCANNER MEMORY
        MOVB @DHMCSR,R4 ;READ CONPENTS OF MEMORY
        MOV R1,R5 ;SET UP EXPECTED CONTENTS
        CMPB R4,R2 ;OF SCANNER MEMORY
        BNE MEMT2C
MEMT2C: BIS #70000,R5
        CMP R4,R5 ;COMPARE EXPECTED AND RECEIVED
        BEQ MEMT2D ;VALUES
        ERRORC ;SCANNER MEMORY ERROR
        SCOPEF ;CHECK FOR DATA FREEZE
        MEMT2A
MEMT2D: INC R1
        DEC R3 ;TEST NEXT SCANNED LOCATION
        BNE MEMT2B
        DEC R0 ;UPDATE LINE COUNT
        BNE MEMT2A
        SCOPE ;CHECK FOR ITERATION, LOOP

```

```

1
2
3
4
5
6 004432
    004432
7 004432 000026
8 004436 005077 011154
9 004444 042767 000340 173332
10 004450 012700 000020
11 004454 012702 000017
12 004460 012703 000020
13 004466 012703 000020
14 004466 012777 001017 011124
15 004466 004767 006712
16 004472 005303
17 004474 001374
18 004476 010277 011110
19 004502 004767 006676
20 004506 012703 000020
21 004512 012777 000017 011072
22 004520 005202
23 004522 005001
24 004524 004767 006654
25 004530 004767 011056
26 004530 117704
27 004534 010105
28 004536 120402
29 004540 001002
30 004542 052705 070000
31 004546 020405
32 004550 001403
33 004552 104000
34 004554 104003
35 004556 004454
36 004560 005201
37 004562 005303
38 004564 001357
39 004566 005300
40 004570 001331
41 004572 104002

;WITH ALL ACANNER MEMORY LOCATIONS SET TO 1'S.
;WRITE 0'S INTO SELECTED LOCATION
;VERIFY THAT ONLY SELECTED LOCATION WAS CLEARED.

TS \NO 0
;REFERANCE NUMBER DEFINITION

T25:
NO=NO+1
;REFERENCE DESIGNATION

MEMT3: CLR @DHMCSR ;CLEAR CONTROL STATUS REGISTER
      BIC #340,PS ;ENABLE INTERRUPTS
      MOV #16.,R0 ;SET UP TO TEST 16 ADDRESSES
      MOV #17,R2 ;FIRST ADDRESS TO BE TESTED=0
MEMT3A: MOV #16.,R3 ;WRITE 1'S INTO ALL SCANNER
        MOV #MAINT+17,@DHMCSR ;MEMORY LOCATIONS
MEMT3B: CALL STEPER ;ACCESS SCANNER MEM
        DEC R3
        BNE MEMT3B
        MOV R2,@DHMCSR ;SET LINE COUNTER TO TEST ADDRESS-1
        CALL STEPER ;WRITE 0'S INTO TEST ADDRESS
        MOV #16.,R3 ;SET UP TO TEST ALL 16
        MOV #17,@DHMCSR ;SCANNER MEMORY LOCATIONS
        INC R2
        CLR R1
MEMT3C: CALL STEPER ;ACESS SCANNER MEMORY
        MOV @DHMCSR,R4 ;READ CONTENTS OF MEMORY
        MOV R1,R5 ;SET UP EXPECTED CONTENTS
        CMPB R4,R2 ;OF SCANNER MEIORY
        BNE MEMT3D
        BIS #70000,R5
MEMT3D: CMP R4,R5 ;COMPARE EXPECTED AND
        BEQ MEMT3E ;RECEIVED VALUES
        ERRORC ;SCANNER MEMORY ERROR
        SCOPEF ;CHECK FOR DATA FREEZE
MEMT3E: INC R1
        DEC R3 ;TEST NEXT SCANNER LOCATION
        BNE MEMT3C
        DEC R0 ;UPDATE ADDRESS COUNT
        BNE MEMT3A
        SCOPE ;CHECK FOR ITERATION, LOOP

```

```

1 004574      MUXS1      1,LINENA,+/LINE ENABLE/
                ;VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
                ;BE SET AND CLEARED FOR SELECTED LINE

004574      .IFEQ SINGLE
                TS \NO,0
                ;REFERANCE NUMBER DEFINITION

004574      000027      T26:
                NO=NO+1      ;REFERENCE DESIGNATION
                .IFF
                TS \N1,1
                .IFTF
004574 005077 011012      MUX1: CLR      @DHMCSR      ;CLEAR CONTROL STATUS REGISTER
004600 042767 000340 173170      BIC      @340,PS      ;ENABLE INTERRUPTS
                .IFT
004606 012700 000020      MOV      @16.,R0      ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
004612 012767 000001 011102      MOV      @1,SELSK      ;INIT LINE SELECT MASK
004620 005001      CLR      R1      ;START AT LINE 0
                .IFF
                MOV      LINE,R1
                .IFTF
004622 012777 002000 010762      MUX1A: MOV      @CLRMUX,@DHMCSR
004630 012702 000020      MOV      @16.,R2
                .IFT
004634 036767 011062 011056      BIT      SELSK,LINSEL      ;IS THIS LINE SELECTED FOR TEST ?
004642 001464      BEQ      MUX1F      ;BR IF NOT
                .IFTF
004644 010177 010742      MOV      R1,@DHMCSR      ;SELECT LINE TO BE TESTED
004650 012777 000001 010736      MOV      @LINENA,@DHMLSR      ;SET LINE ENABLE FUNCTION FLIP-FLOP
                .IFT
004656 012767 000001 011040      MOV      @1,SLMSK      ;INIT ANOTHER SELECT MASK
                .IFTF
004664 005077 010722      CLR      @DHMCSR
004670 005005      MUX1B: CLR      R5
                .IFT
004672 036767 011026 011020      BIT      SLMSK,LINSEL      ;SELECTED ??
004700 001421      BEQ      MUX1D      ;BR IF NOT
                .IFTF
004702 017704 010706      MOV      @DHMLSR,R4      ;READ LINE STATUS REGISTER
004706 117703 010700      MOV      @DHMCSR,R3      ;READ CONTROL STATUS REGISTER
004712 042703 177760      BIC      @177760,R3      ;CLEAR UNWANTED BITS
004716 020103      CMP      R1,R3      ;IF LINE NUMBER=SELECTED LINE NUMBER,
004720 001002      BNE      MUX1C      ;EXCEPT LINE ENABLE FUNCTION FLIP FLOP
004722 012705 000001      MOV      @LINENA,R5      ;TO BE SET
                MUX1C:
004726 042704 000360      .IF IDN <LINENA>,<LINENA>
                BIC      @360,R4      ;CLEAR RING,CO,CS,SECRV
                ;IF NO LEVEL CONVERTER THESE BITS FLOAT
                .ENDC
004732 020504      CMP      R5,R4      ;CMP EXPECTED AND RECVD
004734 001403      BEQ      MUX1D      ;RESULTS
004736 104001      ERRORL      ;LINE STATUS ERROR
004740 104003      SCOPEF
004742 004744      MUX1D:
004744 004767 006434      CALL     STEPER      ;EXAMINE NEXT LINE
                .IFT
    
```

```

004750 006367 010750          ASL      SLMSK          ;SHIFT MASK
                                .IFTF
004754 005302          DEC      R2
004756 001344          BNE     MUX1B
004760 005005          CLR     R5
004762 010177 010624    MUX1E:  MOV     R1,@DHMCSR
004766 010103          MOV     R1,R3          ;SET LINE COUNTER TO SELECTED LINE
004770 005077 010620    CLR     @DHMLSR        ;CLEAR LINE ENABLE FLIP FLOP
004774 105227 000000    INCB   #0             ;DELAY FOR CABLE
005000 001375          BNE     .-4           ;DITTO
005002 017704 010606    MOV     @DHMLSR,R4    ;READ LINE STATUS REGISTER
005006 005704          TST    R4             ;WAS LINE ENABLE FUNCTION FLIP FLOP
005010 001401          BEQ    MUX1F          ;CLEARED
005012 104001          ERRORL
                                .IFT
005014 104003    MUX1F:  SCOPEF
005016 004622          MUX1A
005020 006367 010676    ASL     SELMSK
005024 005201          INC    R1             ;SHIFT SELECT MASK
005026 005300          DEC   R0             ;SELECT NEXT LINE
005030 001274          BNE   MUX1A          ;DECREMENT LINE COUNT
005032 104002          SCOPE                ;CONTINU IF NOT DONE
                                .IFF
                                MUX1F:  SCOPE                ;CHECK FOR ITERATIONS, LOOP
                                .ENDC

```

```

005034          MUXS1          2,TRMRDY,+/TERMINAL READY/
                                ;VERIFY THAT TERMINAL READY FUNCTION FLIP FLOP CAN
                                ;BE SET AND CLEARED FOR SELECTED LINE

005034          .IFEQ SINGLE
TS \NO,0
                                ;REFERANCE NUMBER DEFINITION

005034          T27:          ;REFERENCE DESIGNATION
000030          NO=NO+1
                                .IFF
TS          \N1,1
                                .IFTF
005034 005077 010552          MUX2:  CLR      @DHMCSR          ;CLEAR CONTROL STATUS REGISTER
005040 042767 000340 172730          BIC      @340,PS          ;ENABLE INTERRUPTS
                                .IFT
005046 012700 000020          MOV      @16.,R0          ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
005052 012767 000001 010642          MOV      @1,SELMSK          ;INIT LINE SELECT MASK
005060 005001          CLR      R1          ;START AT LINE 0
                                .IFF
                                MOV      LINE,R1
                                .IFTF
005062 012777 002000 010522          MUX2A: MOV      @CLRMUX,@DHMCSR
005070 012702 000020          MOV      @16.,R2
                                .IFT
005074 036767 010622 010616          BIT      SELMSK,LINSEL          ;IS THIS LINE SELECTED FOR TEST ?
005102 001462          BEQ      MUX2F          ;BR IF NOT
                                .IFTF
005104 010177 010502          MOV      R1,@DHMCSR          ;SELECT LINE TO BE TESTED
005110 012777 000002 010476          MOV      @TRMRDY,@DHMLSR          ;SET TERMINAL READY FUNCTION FLIP-FLOP
                                .IFT
005116 012767 000001 010600          MOV      @1,SLMSK          ;INIT ANOTHER SELECT MASK
                                .IFTF
005124 005077 010462          CLR      @DHMCSR
005130 005005          MUX2B: CLR      R5
                                .IFT
005132 036767 010566 010560          BIT      SLMSK,LINSEL          ;SELECTED ??
005140 001417          BEQ      MUX2D          ;BR IF NOT
                                .IFTF
005142 017704 010446          MOV      @DHMLSR,R4          ;READ LINE STATUS REGISTER
005146 117703 010440          MOVB    @DHMCSR,R3          ;READ CONTROL STATUS REGISTER
005152 042703 177760          BIC      @177760,R3          ;CLEAR UNWANTED BITS
005156 020103          CMP      R1,R3          ;IF LINE NUMBER=SELECTED LINE NUMBER,
005160 001002          BNE     MUX2C          ;EXCEPT TERMINAL READY FUNCTION FLIP FLOP
005162 012705 000002          MOV      @TRMRDY,R5          ;TO BE SET

005166          MUX2C:
                                .IF IDN <TRMRDY>,<LINENA>
                                BIC      @360,R4          ;CLEAR RING,CO,CS,SECRCV
                                ;IF NO LEVEL CONVERTER THESE BITS FLOAT
                                .ENDC
005166 020504          CMP      R5,R4          ;CMP EXPECTED AND RECVD
005170 001403          BEQ     MUX2D          ;RESULTS
005172 104001          ERRORL          ;LINE STATUS ERROR
005174 104003          SCOPEF
005176 005200          MUX2D:
005200 004767 006200          CALL    STEPER          ;EXAMINE NEXT LINE
                                .IFT

```

```

005204 006367 010514          ASL      SLMSK          ;SHIFT MASK
                                .IFTF
005210 005302          DEC      R2
005212 001346          BNE     MUX2B
005214 005005          CLR     R5
005216 010177 010370      MUX2E:  MOV    R1,@DHMCSR
005222 010103          MOV    R1,R3
005224 005077 010364      CLR     @DHMLSR
005230 105227 000000      INCB   #0
005234 001375          BNE     .-4
005236 017704 010352      MOV    @DHMLSR,R4
005242 005704          TST   R4
005244 001401          BEQ   MUX2F
005246 104001          ERRORL

                                .IFT
005250 104003      MUX2F:  SCOPEF
005252 005062          MUX2A
005254 006367 010442      ASL    SELMSK
005260 005201          INC   R1
005262 005300          DEC   R0
005264 001276          BNE   MUX2A
005266 104002          SCOPE

                                .IFF
                                MUX2F:  SCOPE
                                .ENDC

```

```

;SET LINE COUNTER TO SELECTED LINE
;CLEAR TERMINAL READY FLIP FLOP
;DELAY FOR CABLE
;DITTO
;READ LINE STATUS REGISTER
;WAS TERMINAL READY FUNCTION FLIP FLOP
;CLEARED
;NO, LINE STATUS ERROR

;CHECK FOR LOOP ON SAME DATA

;SHIFT SELECT MASK
;SELECT NEXT LINE
;DECREMENT LINE COUNT
;CONTINU IF NOT DONE
;CHECK FOR ITERATIONS, LOOP

;CHECK FOR ITERATIONS, LOOP

```

```

005270          MUXS1          3,RS,1/REQUEST TO SEND/
                                ;VERIFY THAT REQUEST TO SEND FUNCTION FLIP FLOP CAN
                                ;BE SET AND CLEARED FOR SELECTED LINE

005270          .IFEQ SINGLE
                                TS \NO,0
                                ;REFERANCE NUMBER DEFINITION
005270          000031        T30:
                                NO=NO-1
                                .IFF
                                TS \N1,1
                                .IFTF
005270 005077 010316          MUX3: CLR @DHMCSR          ;CLEAR CONTROL STATUS REGISTER
005274 042767 000340 172474  MUX3: BIC #340,PS          ;ENABLE INTERRUPTS
                                .IFT
005302 012700 000020          MOV #16.,R0          ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
005306 012767 000001 010406  MOV #1,SELMSK      ;INIT LINE SELECT MASK
005314 005001                CLR R1          ;START AT LINE 0
                                .IFF
                                MOV LINE,R1
                                .IFTF
005316 012777 002000 010266  MUX3A: MOV #CLRMUX,@DHMCSR
005324 012702 000020          MOV #16.,R2
                                .IFT
005330 036767 010366 010362  BIT SELMSK,LINSEL      ;IS THIS LINE SELECTED FOR TEST ?
005336 001462                BEQ MUX3F          ;BR IF NOT
                                .IFTF
005340 010177 010246          MOV R1,@DHMCSR      ;SELECT LINE TO BE TESTED
005344 012777 000004 010242  MOV #RS,@DHMLSR    ;SET REQUEST TO SEND FUNCTION FLIP-FLOP
                                .IFT
005352 012767 000001 010344  MOV #1,SLMSK      ;INIT ANOTHER SELECT MASK
                                .IFTF
005360 005077 010226          CLR @DHMCSR
005364 005005          MUX3B: CLR R5
                                .IFT
005366 036767 010332 010324  BIT SLMSK,LINSEL    ;SELECTED ??
005374 001417                BEQ MUX3D          ;BR IF NOT
                                .IFTF
005376 017704 010212          MOV @DHMLSR,R4      ;READ LINE STATUS REGISTER
005402 117703 010204          MOV @DHMCSR,R3      ;READ CONTROL STATUS REGISTER
005406 042703 177760          BIC #177760,R3     ;CLEAR UNWANTED BITS
005412 020103                CMP R1,R3          ;IF LINE NUMBER=SELECTED LINE NUMBER,
005414 001002                BNE MUX3C          ;EXCEPT REQUEST TO SEND FUNCTION FLIP FLOP
005416 012705 000004          MOV #RS,R5          ;TO BE SET
005422          MUX3C:
                                .IF IDN <RS>,<LINENA>
                                BIC #360,R4          ;CLEAR RING,CO,CS,SECRV
                                                ;IF NO LEVEL CONVERTER THESE BITS FLOAT
                                .ENDC
005422 020504                CMP R5,R4          ;CMP EXPECTED AND RECVD
005424 001403                BEQ MUX3D          ;RESULTS
005426 104001                ERRDRL          ;LINE STATUS ERROR
005430 104003                SCOPEF
005432 005434                MUX3D
005434 004767 005744          MUX3D: CALL STEPER          ;EXAMINE NEXT LINE
                                .IFT

```



```

005524          MUXS1          4,SECTX,+/SECONDARY TRANSMIT/
                                ;VERIFY THAT SECONDARY TRANSMIT FUNCTION FLIP FLOP CAN
                                ;BE SET AND CLEARED FOR SELECTED LINE

005524          .IFEQ SINGLE
TS \NO,0
                                ;REFERANCE NUMBER DEFINITION

005524          000032        T31:
                                NO=NO+1
                                .IFF
                                TS \N1,1
                                .IFTF
005524 005077 010062        MUX4: CLR @DHMCSR
005530 042767 000340 172240        BIC #340,PS
                                ;CLEAR CONTROL STATUS REGISTER
                                ;ENABLE INTERRUPTS

005536 012700 000020        .IFT
005542 012767 000001 010152        MOV #16.,R0
005550 005001                MOV #1,SELMSK
                                CLR R1
                                ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
                                ;INIT LINE SELECT MASK
                                ;START AT LINE 0

                                .IFF
                                MOV LINE,R1

                                .IFTF
005552 012777 002000 010032        MUX4A: MOV #CLRMUX,@DHMCSR
005560 012702 000020        MOV #16.,R2

                                .IFT
005564 036767 010132 010126        BIT SELMSK,LINSEL
005572 001462                BEQ MUX4F
                                ;IS THIS LINE SELECTED FOR TEST ?
                                ;BR IF NOT

                                .IFTF
005574 010177 010012        MOV R1,@DHMCSR
005600 012777 000010 010006        MOV #SECTX,@DHMLSR
                                ;SELECT LINE TO BE TESTED
                                ;SET SECONDARY TRANSMIT FUNCTION FLIP-FLOP

                                .IFT
005606 012767 000001 010110        MOV #1,SLMSK
                                ;INIT ANOTHER SELECT MASK

                                .IFTF
005614 005077 007772        CLR @DHMCSR
005620 005005        MUX4B: CLR R5

                                .IFT
005622 036767 010076 010070        BIT SLMSK,LINSEL
005630 001417        BEQ MUX4D
                                ;SELECTED ??
                                ;BR IF NOT

                                .IFTF
005632 017704 007756        MOV @DHMLSR,R4
005636 117703 007750        MOV @DHMCSR,R3
005642 042703 177760        BIC #177760,R3
005646 020103        CMP R1,R3
005650 001002        BNE MUX4C
005652 012705 000010        MOV #SECTX,R5
                                ;EXCEPT SECONDARY TRANSMIT FUNCTION FLIP FLOP

                                ;TO BE SET

005656          MUX4C:
                                .IF IDN <SECTX>,<LINENA>
                                BIC #360,R4
                                ;CLEAR RING,CO,CS,SECRCV
                                ;IF NO LEVEL CONVERTER THESE BITS FLOAT

                                .ENDC

005656 020504        CMP R5,R4
005660 001403        BEQ MUX4D
005662 104001        ERRORL
005664 104003        SCOPEF
005666 005670        MUX4D
005670 004767 005510        MUX4D: CALL STEPER
                                .IFT
                                ;EXAMINE NEXT LINE

```



```

2
3
4
5 005760          TS \NO.0
                    ;REFERANCE NUMBER DEFINITION
                    T32:
                    NO=NO+1
                    ;REFERENCE DESIGNATION
005760
6 005760 000033    007626
7 005764 042767    000340 172004
8 005772 012700    000020
9 005776 012777    000017 007610
10 006004 004767   005374
11 006010 005300
12 006012 001371
13 006014 012767   000001 007700
14 006022 005003
15 006024 012700   000020
16 006030 012777   002000 007554
17 006036 036767   007660 007654
18 006044 001427
19 006046 010377   007540
20 006052 017704   007536
21 006056 005005
22 006060 005704
23 006062 001403
24 006064 104001
25 006066 104003
26 006070 006030
27 006072 005205
28 006074 052777   000001 007512
29 006102 017704   007506
30 006106 042704   000360
31 006112 020504
32 006114 001403
33 006116 104001
34 006120 104003
35 006122 006030
36 006124 005203
37 006126 005077   007462
38 006132 006367   007564
39 006136 005300
40 006140 001336
41 006142 104002

MUX8: CLR @DHMCSR ;CLEAR CONTROL REGISTER
      BIC #340,PS ;ENABLE INTERRUPTS
      MOV #16.,R0 ;SET UP TO TEST 16 LINES
MUX8A: MOV #17,@DHMLSR ;WRITE 1S INTO ALL MULTIPLEXER
      CALL STEPER ;FUNCTION FLIPFLOPS
      DEC R0
      BNE MUX8A
      MOV #1,SELMSK ;INIT SELECT MASK
      CLR R3 ;SET UP FOR 16 LINES
      MOV #16.,R0
MUX8B: MOV @CLRMUX,@DHMCSR ;CLEAR MULTIPLEXER
MUX8C: BIT SELMSK,LINSEL ;SELECTED ??
      BEQ MUX8E ;BR IF NOT
      MOV R3,@DHMCSR ;SELECT LINE
      MOV @DHMLSR,R4 ;READ LINE STATUS REGISTER
      CLR R5 ;EXPECT OS
      TST R4 ;WAS LINE STATUS REGISTER CLEARED
      BEQ MUX8D
      ERRORL ;LINE STATUS ERROR
      SCOPEF ;CHECK FOR LOOP ON SAME DATA
MUX8D: INC R5 ;EXPECT LINE ENABLE
      BIS #LINENA,@DHMLSR ;SET LINE ENABLE ON SELECTED LINE
      MOV @DHMLSR,R4 ;READ LINE STATUS REGISTER
      BIC #360,R4 ;CLEAR RING,CO,CS SECRCV-MAY FLOAT HIGH
      CMP R5,R4 ;IS ANYTHING BUT LINE ENABLE SET
      BEQ MUX8E
      ERRORL ;LINE STATUS ERROR
      SCOPEF ;CHECK FOR LOOP ON SAME DATA
MUX8E: INC R3 ;UPDATE LINE NUMBER
      CLR @DHMLSR ;CLEAR CURRENT LINE
      ASL SELMSK ;SHIFT SELECT MASK
      DEC R0 ;CONTINUE IF ALL LINES NOT
      BNE MUX8C ;TESTED
      SCOPE ;CHECK FOR ITERATIONS, LOOP

```

1
2
3
4
5
6
7
8
9
10

;WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS
;SET "LINE ENABLE FOR ALL LINES
;VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE
;THIS TEST WILL FAIL ON ANY LINE THAT DOES
;NOT HAVE A LEVEL CONVERTER--CO,CS,RING,SECRCV WILL FLOAT
;HIGH,HENCE MAINT. MODE CANNOT TOGGLE THESE SIGNALS TO
;CAUSE A TRANSITION.A PROGRAM TIME OUT WILL OCCUR.....

11 006144

TS \NO,0

;REFERANCE NUMBER DEFINITION

006144

T33:

;REFERENCE DESIGNATION

NO=NO+1

| | | | | | | | | |
|----|--------|--------|--------|--------|---------|--------|--------------------|--------------------------------------|
| 12 | 006144 | 012777 | 002000 | 007440 | SCNT1: | MOV | #CLRMUX,@DHMCSR | ;CLEAR ALL MULTIPLEXER FLIPFLOPS |
| 13 | 006152 | 005077 | 007434 | | | CLR | @DHMCSR | ;CLEAR CONTROL REGISTER |
| 14 | 006156 | 042767 | 000340 | 171612 | | BIC | #340,PS | ;ENABLE INTERRUPTS |
| 15 | 006164 | 012700 | 000020 | | | MOV | #16.,RO | ;SET UP TO WRITE 1'S INTO |
| 16 | 006170 | 012777 | 001017 | 007414 | | MOV | #MAINT+17,@DHMCSR | ;ALL SCANNER MEMORY LOCATION |
| 17 | 006176 | 012767 | 000001 | 007516 | | MOV | #1,SELMSK | ;INIT SELECT MASK |
| 18 | 006204 | 004767 | 005174 | | SCNT1A: | CALL | STEPPER | ;WRITE A LOCATION |
| 19 | 006210 | 012777 | 000001 | 007376 | | MOV | #LINENA,@DHMLSR | ;LET "LINE ENABLE" |
| 20 | 006216 | 005300 | | | | DEC | RO | |
| 21 | 006220 | 001371 | | | | BNE | SCNT1A | |
| 22 | 006222 | 012701 | 177777 | | | MOV | #-1,R1 | ;INIT LINE NO. GEN. |
| 23 | 006226 | 012705 | 070300 | | | MOV | #70300,R5 | ;EXPECT "DONE"+"COF"+"CSF"+"SECRCV" |
| 24 | 006232 | 012777 | 006404 | 007346 | | MOV | #SCNT1C,@DHMVEC | ;SET UP LOCAL INTERRUPT SERVICE |
| 25 | 006240 | 012777 | 000340 | 007342 | | MOV | #340,@DHMLVL | ;SERVICE AT LEVEL 7 |
| 26 | 006246 | 012700 | 000020 | | | MOV | #16.,RO | |
| 27 | 006252 | 012777 | 000117 | 007332 | | MOV | #INTENA+17,@DHMCSR | ;SET INTERRUPT ENABLE |
| 28 | 006260 | 036767 | 007436 | 007432 | SCNT1B: | BIT | SELMSK,LINSEL | ;SELECTED ?? |
| 29 | 006266 | 001456 | | | | BEQ | SCNT1D | ;BR IF NOT |
| 30 | 006270 | 052767 | 000340 | 171500 | | BIS | #340,PS | ;LOCK OUT INTERRUPTS |
| 31 | 006276 | 004767 | 005102 | | | CALL | STEPPER | ;HIT THE SCANNER ONCE |
| 32 | 006302 | 005003 | | | | CLR | R3 | ;CLEAR DELAY |
| 33 | 006304 | 042767 | 000340 | 171464 | | BIC | #340,PS | ;ENABLE INTERRUPTS |
| 34 | 006312 | 005303 | | | 1\$: | DEC | R3 | ;WAIT LONG ENOUGH? |
| 35 | 006314 | 001404 | | | | BEQ | 2\$ | ;WE HAVE AN ERROR |
| 36 | 006316 | 105777 | 007270 | | | TSTB | @DHMCSR | ;DID DONE SET |
| 37 | 006322 | 100373 | | | | BPL | 1\$ | ;NOT YET |
| 38 | 006324 | 100416 | | | | BMI | 3\$ | ;SET BUT NO INTERRUPT |
| 39 | 006326 | 052767 | 000340 | 171442 | 2\$: | BIS | #340,PS | |
| 40 | 006334 | 017704 | 007252 | | | MOV | @DHMCSR,R4 | ;GET FAILING LINE |
| 41 | 006340 | 010402 | | | | MOV | R4,R2 | ;GET CSR |
| 42 | 006342 | 017703 | 007246 | | | MOV | @DHMLSR,R3 | ;GET LSR |
| 43 | 006346 | 042704 | 177760 | | | BIC | #177760,R4 | |
| 44 | 006352 | 104030 | | | | ERRINT | | ;REPORT ERROR HAS OCCURED |
| 45 | 006354 | 104003 | | | | SCOPEF | | |
| 46 | 006356 | 006144 | | | | SCNT1 | | |
| 47 | 006360 | 000421 | | | | BR | SCNT1D | ;CONTINUE THE TEST |
| 48 | 006362 | 052767 | 000340 | 171406 | 3\$: | BIS | #340,PS | ;INTERRUPT DID NOT OCCUR |
| 49 | 006370 | 017704 | 007216 | | | MOV | @DHMCSR,R4 | ;ERROR |
| 50 | 006374 | 104000 | | | | ERRORC | | ;CONTROL STATUS ERROR |
| 51 | 006376 | 104003 | | | | SCOPEF | | ;CHECK FOR LOOP ON SAME DATA |
| 52 | 006400 | 006144 | | | | SCNT1 | | |
| 53 | 006402 | 000410 | | | | BR | SCNT1D | |
| 54 | 006404 | 022626 | | | SCNT1C: | POP2SP | | ;INTERRUPT OCCURED, REPOSITION STACK |

| | | | | | | | | | | |
|----|--------|--------|--------|--------|---------|-------|------------------|--|--|--|
| 55 | 006406 | 017704 | 007200 | | | | | | | |
| 56 | 006412 | 020504 | | | | | | | | |
| 57 | 006414 | 001403 | | | | | | | | |
| 58 | 006416 | 104000 | | | | | | | | |
| 59 | 006420 | 104003 | | | | | | | | |
| 60 | 006422 | 006144 | | | | | | | | |
| 61 | 006424 | 042777 | 000217 | 007160 | SCNT1D: | BIC | #DONE+17,@DHMCSR | | | |
| 62 | 006432 | 005201 | | | | INC | R1 | | | |
| 63 | 006434 | 050177 | 007152 | | | BIS | R1,@DHMCSR | | | |
| 64 | 006440 | 006367 | 007256 | | | ASL | SELSK | | | |
| 65 | 006444 | 005205 | | | | INC | R5 | | | |
| 66 | 006446 | 005300 | | | | DEC | R0 | | | |
| 67 | 006450 | 001303 | | | | BNE | SCNT1B | | | |
| 68 | 006452 | 104002 | | | | SCOPE | | | | |

```

;READ CONTROL STATUS
;ARE EXPECTED AND RECEIVED
;REGISTERS THE SAME
;NO, LINE STATUS ERROR
;CHECK FOR LOOP WITH CURPENT DATA

;CLEAR D DONE
;GEN NXT LINE NO.
;SET LINE NO. BITS
;SHIFT SELECT MASK
;UPDATE EXPECTED RESULT
;CONTINUE IF NOT DONE

;CHECK FOR ITERATIONS, LOOP

```

```

1
2
3      .MACRO COMMENT
4      .NLIST
5      SINGLE=1
6      .LIST
7 006454      .ENDM
           COMMENT
           SINGLE=1
8
9
10      ;SINGLE LINE CABLE TEST
11      ;FOR USE WITH MODEM CABLE AND DC11 TEST CONNECTOR
12
13      ;NOTE: MODEM CONTROL MULTIPLEXER INPUTS SHOULD BE CONNECTED
14      ;TO DISTRIBUTION PANEL WIA DM11-DC
15
16      .MACRO COMMENT
17      .NLIST
18      N1=100
19      N=N1
20      XN=N1
21      .LIST
22      .ENDM
23 0C6454      COMMENT
           N1=100
           N=N1
           XN=N1
           TS
           \N1,1
           ;REFERANCE NUMBER DEFINITION
           T100:
           N1=N1+1
           ;REFERENCE DESIGNATION
25 006454 012767 006474 173562 STRLIN: MOV #STRLNA,KRET
26 006462 042767 000340 171306      BIC #340,PS
           ;SET UP FOR NEW LINE SELECTION
           ;ENABLE INTERRUPTS
           ;TYPE "SINGLE LINE CABLE TEST"
27 006470 104004
           TYPE
           MLINE
           ;GET LINE NUMBER
28 006472 017010
           STRLNA: INSTRG
           MLINEI
           0
           17
           LINE
           TYPE
           MCRLF
29 006474 104013
30 006476 017043
31 006500 000000
32 006502 000017
33 006504 015700
34 006506 104004
35 006510 017005

```

```

1
2 006512          MUXS1  11,LINENA,+/LINE ENABLE/
                   ;VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
                   ;BE SET AND CLEARED FOR SELECTED LINE

                   .IFEQ  SINGLE
                   TS  \NO,0
                   .IFF
006512          TS      \N1,1
                   ;REFERANCE NUMBER DEFINITION
006512          T101:
                   N1=N1+1
                   ;REFERENCE DESIGNATION
                   .IFTF
006512  005077  007074  MUX11: CLR  @DHMCSR
006516  042767  000340  171252 BIC  #340,PS
                   ;CLEAR CONTROL STATUS REGISTER
                   ;ENABLE INTERRUPTS
                   .IFT
                   MOV  #16.,R0
                   MOV  #1,SELMSK
                   CLR  R1
                   ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
                   ;INIT LINE SELECT MASK
                   ;START AT LINE 0
006524  016701  007150
                   .IFF
                   MOV  LINE,R1
                   .IFTF
006530  012777  002000  007054 MUX11A: MOV  #CLRMUX,@DHMCSR
006536  012702  000020          MOV  #16.,R2
                   .IFT
                   BIT  SELMSK,LINSEL
                   BEQ  MUX11F
                   ;IS THIS LINE SELECTED FOR TEST ?
                   ;BR IF NOT
                   .IFTF
006542  010177  007044          MOV  R1,@DHMCSR
006546  012777  000001  007040 MOV  #LINENA,@DHMLSR
                   ;SELECT LINE TO BE TESTED
                   ;SET LINE ENABLE FUNCTION FLIP-FLOP
                   .IFT
                   MOV  #1,SLMSK
                   ;INIT ANOTHER SELECT MASK
                   .IFTF
006554  005077  007032          CLR  @DHMCSR
006560  005005          MUX11B: CLR  R5
                   .IFT
                   BIT  SLMSK,LINSEL
                   BEQ  MUX11D
                   ;SELECTED ??
                   ;BR IF NOT
                   .IFTF
006562  017704  007026          MOV  @DHMLSR,R4 >
006566  117703  007020          MOVB @DHMCSR,R3
006572  042703  177760          BIC  #177760,R3
006576  020103          CMP  R1,R3
006600  001002          BNE  MUX11C
006602  012705  000001          MOV  #LINENA,R5
                   ;EXCEPT LINE ENABLE FUNCTION FLIP FLOP
                   ;TO BE SET
006606          MUX11C:
                   .IF IDN <LINENA>,<LINENA>
006606  042704  000360          BIC  #360,R4
                   ;CLEAR RING,CO,CS,SECRCV
                   ;IF NO LEVEL CONVERTER THESE BITS FLOAT
                   .ENDC
006612  020504          CMP  R5,R4
006614  001403          BEQ  MUX11D
006616  104001          ERRORL
006620  104003          SCOPEF
006622  006624          MUX11D
006624  004767  004554          MUX11D: CALL  STEPER
                   ;EXAMINE NEXT LINE

```



```

1 006672          MUXS1  12,TRMRDY,+/TERMINAL READY/
                   ;VERIFY THAT TERMINAL READY FUNCTION FLIP-FLOP CAN
                   ;BE SET AND CLEARED FOR SELECTED LINE

                   .IFEQ  SINGLE
                   TS  \NO,0
                   .IFF
006672          TS      \N1,1
                   ;REFERANCE NUMBER DEFINITION
006672          T102:   ;REFERENCE DESIGNATION
                   N1=N1+1
                   .IFTF
006672 005077 006714  MUX12: CLR  @DHMCSR           ;CLEAR CONTROL STATUS REGISTER
006676 042767 000340 171072 BIC  @340,PS       ;ENABLE INTERRUPTS
                   .IFT
                   MOV  @16.,R0           ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
                   MOV  @1,SELMSK        ;INIT LINE SELECT MASK
                   CLR  R1                ;START AT LINE 0
                   .IFF
006704 016701 006770          MOV  LINE,R1
                   .IFTF
006710 012777 002000 006674  MUX12A: MOV  @CLRMUX,@DHMCSR
006716 012702 000020          MOV  @16.,R2
                   .IFT
                   BIT  SELMSK,LINSEL     ;IS THIS LINE SELECTED FOR TEST ?
                   BEQ  MUX12F            ;BR IF NOT
                   .IFTF
006722 010177 006664 006660  MOV  R1,@DHMCSR       ;SELECT LINE TO BE TESTED
006726 012777 000002          MOV  @TRMRDY,@DHMLSR    ;SET TERMINAL READY FUNCTION FLIP-FLOP
                   .IFT
                   MOV  @1,SLMSK         ;INIT ANOTHER SELECT MASK
                   .IFTF
006734 005077 006652          CLR  @DHMCSR
006740 005005          MUX12B: CLR  R5
                   .IFT
                   BIT  SLMSK,LINSEL     ;SELECTED ??
                   BEQ  MUX12D            ;BR IF NOT
                   .IFTF
006742 017704 006646          MOV  @DHMLSR,R4       ;READ LINE STATUS REGISTER
006746 117703 006640          MOVB @DHMCSR,R3       ;READ CONTROL STATUS REGISTER
006752 042703 177760          BIC  @177760,R3      ;CLEAR UNWANTED BITS
006756 020103          CMP   R1,R3           ;IF LINE NUMBER=SELECTED LINE NUMBER,
006760 001002          BNE  MUX12C        ;EXCEPT TERMINAL READY FUNCTION FLIP FLOP
006762 012705 000002          MOV  @TRMRDY,R5
                   ;TO BE SET
006766          MUX12C:
                   .IF IDN <TRMRDY>,<LINENA>
                   BIC  @360,R4           ;CLEAR RING.CO,CS,SECRCV
                   ;IF NO LEVEL CONVERTER THESE BITS FLOAT
                   .ENDC
006766 020504          CMP   R5,R4           ;CMP EXPECTED AND RECVD
006770 001403          BEQ  MUX12D        ;RESULTS
006772 104001          ERRORL           ;LINE STATUS ERROR
006774 104003          SCOPEF
006776 007000          MUX12D: CALL  STEPER
007000 004767 004400          .IFT           ;EXAMINE NEXT LINE

```

```

                                ASL      SLMSK      ;SHIFT MASK
                                .IFTF
007004 005302
007006 001354
007010 005005
007012 010177 006574      MUX12E: MOV    R1,@DHMCSR
007016 010103            MOV    R1,R3
007020 005077 006570      CLR    @DHMLSR
007024 105227 000000      INCB   #0
007030 001375            BNE    .-4
007032 017704 006556      MOV    @DHMLSR,R4
007036 005704            TST   R4
007040 001401            BEQ   MUX12F
007042 104001            ERRORL

                                .IFT
MUX12F: SCOPEF
                                MUX12A
                                ASL      SELMSK
                                INC      R1
                                DEC      R0
                                BNE     MUX12A
                                SCOPE

                                .IFF
007044 104002      MUX12F: SCOPE
                                .ENDC

```

```

;SET LINE COUNTER TO SELECTED LINE
;CLEAR TERMINAL READY FLIP FLOP
;DELAY FOR CABLE
;DITTO
;READ LINE STATUS REGISTER
;WAS TERMINAL READY FUNCTION FLIP FLOP
;CLEARED
;NO, LINE STATUS ERROR

;CHECK FOR LOOP ON SAME DATA

;SHIFT SELECT MASK
;SELECT NEXT LINE
;DECREMENT LINE COUNT
;CONTINU IF NOT DONE
;CHECK FOR ITERATIONS, LOOP

;CHECK FOR ITERATIONS, LOOP

```

```

007046          MUXS1  13,RS,+/REQUEST TO SEND/
                  ;VERIFY THAT REQUEST TO SEND FUNCTION FLIP FLOP CAN
                  ;BE SET AND CLEARED FOR SELECTED LINE

                  .IFEQ  SINGLE
                  TS  \NO,0
007046          .IFF
                  TS  \N1,1
                  ;REFERANCE NUMBER DEFINITION
007046          T103:  N1=N1-1          ;REFERENCE DESIGNATION
                  .IFTF
007046 005077 006540 MUX13:  CLR  @DHMCSR          ;CLEAR CONTROL STATUS REGISTER
007052 042767 000340 170716        BIC  @340,PS          ;ENABLE INTERRUPTS

                  .IFT
                  MOV  @16.,R0          ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
                  MOV  @1,SELMSK        ;INIT LINE SELECT MASK
                  CLR  R1                ;START AT LINE 0

                  .IFF
007060 016701 006614          MOV  LINE,R1

                  .IFTF
007064 012777 002000 006520 MUX13A: MOV  @CLRMUX,@DHMCSR
007072 012702 000020          MOV  @16.,R2

                  .IFT
                  BIT  SELMSK,LINSEL        ;IS THIS LINE SELECTED FOR TEST ?
                  BEQ  MUX13F              ;BR IF NOT

                  .IFTF
007076 010177 006510          MOV  R1,@DHMCSR          ;SELECT LINE TO BE TESTED
007102 012777 000004 006504        MOV  @RS,@DHMLSR        ;SET REQUEST TO SEND FUNCTION FLIP-FLOP

                  .IFT
                  MOV  @1,SLMSK          ;INIT ANOTHER SELECT MASK

                  .IFTF
007110 005077 006476          CLR  @DHMCSR
007114 005005          MUX13B: CLR  R5

                  .IFT
                  BIT  SLMSK,LINSEL        ;SELECTED ??
                  BEQ  MUX13D              ;BR IF NOT

                  .IFTF
007116 017704 006472          MOV  @DHMLSR,R4          ;READ LINE STATUS REGISTER
007122 117703 006464          MOV  @DHMCSR,R3          ;READ CONTROL STATUS REGISTER
007126 042703 177760          BIC  @177760,R3        ;CLEAR UNWANTED BITS
007132 020103          CMP  R1,R3            ;IF LINE NUMBER=SELECTED LINE NUMBER,
007134 001002          BNE  MUX13C          ;EXCEPT REQUEST TO SEND FUNCTION FLIP FLOP
007136 012705 000004          MOV  @RS,R5            ;TO BE SET

007142          MUX13C:
                  .IF  IDN <RS>,<LINENA>
                  BIC  @360,R4          ;CLEAR RING.CO.CS,SECRCV
                  ;IF NO LEVEL CONVERTER THESE BITS FLOAT

                  .ENDC
007142 020504          CMP  R5,R4            ;CMP EXPECTED AND RECVD
007144 001403          BEQ  MUX13D          ;RESULTS
007146 104001          ERRORL          ;LINE STATUS ERROR
007150 104003          SCOPEF
007152 007154          MUX13D:
007154 004767 004224 MUX13D: CALL  STEPER          ;EXAMINE NEXT LINE
                  .IFT

```

```

                                ASL      SLMSK      ;SHIFT MASK
                                .IFTF
007160 005302
007162 001354
007164 005005
007166 010177 006420
007172 010103
007174 005077 006414
007200 105227 000000
007204 001375
007206 017704 006402
007212 005704
007214 001401
007216 104001
                                MUX13E: MOV    R1,@DHMCSR
                                MOV    R1,R3
                                CLR    @DHMLSR
                                INCB   #0
                                BNE    .-4
                                MOV    @DHMLSR,R4
                                TST    R4
                                BEQ    MUX13F
                                ERRORL
                                .IFT
                                MUX13F: SCOPEF
                                MUX13A
                                ASL    SELMSK
                                INC    R1
                                DEC    R0
                                BNE    MUX13A
                                SCOPE
                                .IFF
007220 104002
                                MUX13F: SCOPE
                                .ENDC
;SET LINE COUNTER TO SELECTED LINE
;CLEAR REQUEST TO SEND FLIP FLOP
;DELAY FOR CABLE
;DITTO
;READ LINE STATUS REGISTER
;WAS REQUEST TO SEND FUNCTION FLIP FLOP
;CLEARED
;NO. LINE STATUS ERROR
;CHECK FOR LOOP ON SAME DATA
;SHIFT SELECT MASK
;SELECT NEXT LINE
;DECREMENT LINE COUNT
;CONTINU IF NOT DONE
;CHECK FOR ITERATIONS. LOOP
;CHECK FOR ITERATIONS. LOOP

```

```

007222          MUXS1  14,SECTX,+/SECONDARY TRANSMIT/
                  ;VERIFY THAT SECONDARY TRANSMIT FUNCTION FLIP FLOP CAN
                  ;BE SET AND CLEARED FOR SELECTED LINE

                  .IFEQ  SINGLE
                  TS  \NO,0
                  .IFF
007222          TS      \N1,1
                  ;REFERANCE NUMBER DEFINITION
007222          T104:
                  N1=N1+1
                  .IFTF
007222 005077 006364 MUX14: CLR  @DHMCSR
007226 042767 000340 170542 BIC  #340,PS
                  ;CLEAR CONTROL STATUS REGISTER
                  ;ENABLE INTERRUPTS

                  .IFT
                  MOV  #16.,R0
                  MOV  #1,SELMSK
                  CLR  R1
                  ;SET UP TO TEST 16 FUNCTION FLIP-FLOP
                  ;INIT LINE SELECT MASK
                  ;START AT LINE 0

007234 016701 006440          .IFF
                  MOV  LINE,R1
007240 012777 002000 006344 .IFTF
007246 012702 000020 MUX14A: MOV  @CLRMUX,@DHMCSR
                  MOV  #16.,R2
                  .IFT
                  BIT  SELMSK,LINSEL
                  BEQ  MUX14F
                  ;IS THIS LINE SELECTED FOR TEST ?
                  ;BR IF NOT

007252 010177 006334          .IFTF
007256 012777 000010 006330 MOV  R1,@DHMCSR
                  MOV  #SECTX,@DHMLSR
                  ;SELECT LINE TO BE TESTED
                  ;SET SECONDARY TRANSMIT FUNCTION FLIP-FLOP

                  .IFT
                  MOV  #1,SLMSK
                  ;INIT ANOTHER SELECT MASK

007264 005077 006322          .IFTF
007270 005005 MUX14B: CLR  @DHMCSR
                  CLR  R5
                  .IFT
                  BIT  SLMSK,LINSEL
                  BEQ  MUX14D
                  ;SELECTED ??
                  ;BR IF NOT

007272 017704 006316          .IFTF
007276 117703 006310 MOV  @DHMLSR,R4
007302 042703 177760 MOV  @DHMCSR,R3
007306 020103          BIC  #177760,R3
007310 001002          CMP  R1,R3
007312 012705 000010 BNE  MUX14C
                  MOV  #SECTX,R5
                  ;EXCEPT SECONDARY TRANSMIT FUNCTION FLIP FLOP
                  ;TO BE SET

007316          MUX14C:
                  .IF IDN <SECTX>,<LINENA>
                  BIC  #360,R4
                  ;CLEAR RING,CO,CS,SECRCV
                  ;IF NO LEVEL CONVERTER THESE BITS FLOAT

                  .ENDC
007316 020504          CMP  R5,R4
007320 001403          BEQ  MUX14D
007322 104001          ERRORL
007324 104003          SCOPEF
007326 007330          MUX14D
007330 004767 004050 MUX14D: CALL  STEPER
                  .IFT
                  ;EXAMINE NEXT LINE

```

```

007334 005302
007336 001354
007340 005005
007342 010177 006244
007346 010103
007350 005077 006240
007354 105227 000000
007360 001375
007362 017704 006226
007366 005704
007370 001401
007372 104001

.IFTF ASL SLMSK ;SHIFT MASK
      DEC R2
      BNE MUX14B
      CLR R5
MUX14E: MOV R1,@DHMCSR
      MOV R1,R3 ;SET LINE COUNTER TO SELECTED LINE
      CLR @DHMLSR ;CLEAR SECONDARY TRANSMIT FLIP FLOP
      INCB #0 ;DELAY FOR CABLE
      BNE . 4 ;DITTO
      MOV @DHMLSR,R4 ;READ LINE STATUS REGISTER
      TST R4 ;WAS SECONDARY TRANSMIT FUNCTION FLIP FLOP
      BEQ MUX14F ;CLEARED
      ERRORL ;NO, LINE STATUS ERROR

.IFT
MUX14F: SCOPEF ;CHECK FOR LOOP ON SAME DATA
MUX14A
      ASL SELMSK ;SHIFT SELECT MASK
      INC R1 ;SELECT NEXT LINE
      DEC R0 ;DECREMENT LINE COUNT
      BNE MUX14A ;CONTINU IF NOT DONE
      SCOPE ;CHECK FOR ITERATIONS, LOOP

.IFF
MUX14F: SCOPE ;CHECK FOR ITERATIONS, LOOP
.ENDC
007374 104002

```

```

007376          MUXS2  15,TRMRDY,CO+CS,+/CLEAR TO SEND AND CARRIER ARE/,+/TERMINAL/
                ;VERIFY THAT CLEAR TO SEND AND CARRIER ARE SET IF "LINE ENABLE"
                ;AND TERMINAL ARE SET FOR SELECTED LINE.

                .IFEQ  SINGLE
                TS  \NO,0
                .IFF
007376          TS      \N1,1
                ;REFERENCE NUMBER DEFINITION
007376          T105:   ;REFERENCE DESIGNATION
                N1=N1+1
                .IFTF
007376 005077 006210 MUX15: CLR  @DHMCSR      ;CLEAR CONTROL REGISTER
007402 042767 000340 170366 BIC  #340,PS      ;ENABLE INTERRUPTS
                .IFT
                MOV  #16.,R0      ;SET UP TO TEST 16 LINES
                CLR  R1           ;START AT LINE 0
                MOV  #1,SELMSK    ;INIT LINE SELECT MASK
                .IFF
007410 016701 006264      MOV  LINE,R1
                .IFTF
007414 012702 000020 MUX15A: MOV  #16.,R2      ;16 LINES
                .IFT
                BIT  SELMSK,LINSEL ;THIS LINE SELECTED FOR TEST ?
                BEQ  MUX15F      ;BR IF NOT
                .IFTF
007420 010177 006166      MOV  R1,@DHMCSR      ;SELECT A LINE
007424 012777 000003 006162 MOV  #LINENA+TRMRDY,@DHMLSR ;SET LINE ENABLE +TRMRDY

                ;THESE TWO NOP'S ADDED FOR SOME DELAY FOR PDP-11/44
                ;WITH CACHE MEMORY TURNED ON.
007432 000240      NOP
007434 000240      NOP
                ;KR 10-JULY-84 REV E
                ;KR 10-JULY-84 REV E

007436 005077 006150      CLR  @DHMCSR      ;CLEAR CONTROL REGISTER
007442 005005 MUX15B: CLR  R5           ;CLEAR EXPECTED RESULT
007444 017704      MOV  @DHMLSR,R4      ;READ LINE STATUS
007450 117703 006144      MOV  @DHMLSR,R4      ;READ LINE STATUS
007450 117703 006136      MOV  @DHMCSR,R3      ;READ LINE NUMBER
007454 042703 177760      BIC  #177760,R3      ;CLEAR UNWANTED BITS
007460 020103      CMP  R1,R3           ;IF RECEIVED LINE=SELECTED LINE
007462 001002      BNE  MUX15C          ;EXPECT LINE ENABLE AND
007464 012705 000143      MOV  #LINENA+TRMRDY+CO+CS,R5

007470 020405 MUX15C: CMP  R4,R5          ;CLEAR TO SEND AND CARRIER ARE SET
007472 001403      BEQ  MUX15D          ;COMPARE EXPECTED AND
007474 104001      ERRORL              ;RECEIVED RESULTS
007476 104003      SCOPEF              ;LINE STATUS ERROR
007500 007502 MUX15D: MUX15D
007502 004767 003676 MUX15D: CALL  STEPER      ;UPDATE LINE COUNTER
007506 005302      DEC  R2           ;CONTINUE IF ALL CHECKS
007510 001354      BNE  MUX15B          ;ARE NOT DONE FOR THIS LINE
007512 012705 000001 MUX15E: MOV  #LINENA,R5      ;EXPECT LINE ENABLE
007516 010103      MOV  R1,R3           ;ON SELECTED LINE
007520 010177 006066      MOV  R1,@DHMCSR      ;SELECT LINE
007524 042777 000002 006062 BIC  #TRMRDY,@DHMLSR    ;CLEAR TERMINAL
007532 105227 000000      INCB #0       ;DELAY FOR CABLE

```

007536 001375
 007540 017704 006050
 007544 020504
 007546 001401
 007550 104001

```

BNE . 4
MOV @DHMLSR,R4
CMP R5,R4
BEQ MUX15F
ERRORL

.IFT
MUX15F: SCOPEF
MUX15A
INC R1
CLR @DHMLSR
ASL SELMSK
DEC R0
BNE MUX15A
SCOPE
    
```

007552 104002

```

.IFF
MUX15F: SCOPE
.ENDC
    
```

```

;DITTO
;READ LINE STATUS REGISTER
;ONLY LINE ENABLE SHOULD BE
;SET ON THIS LINE
;LINE STATUS ERROR

;CHECK FOR LOOP ON SAME DATA

;UPDATE LINE NUMBER
;CLEAR LINE STATUS REGISTER
;SHIFT MARK TO TEST NEXT LINE
;CONTINUE IF ALL LINES NOT
;TESTED
;CHECK FOR ITERATIONS, LOOP

;CHECK FOR ITERATIONS, LOOP
    
```



```

007554          MUXS2  16,RS,RING,+R/RING IS/,+R/REQUEST TO SEND/
                  ;VERIFY THAT RING IS SET IF "LINE ENABLE"
                  ;AND REQUEST TO SEND ARE SET FOR SELECTED LINE.

                  .IF EQ SINGLE
                  TS \NO,0
                  .IFF
007554          TS      \N1,1
                  ;REFERENCE NUMBER DEFINITION
007554          T106:   NI=N1+1          ;REFERENCE DESIGNATION
                  .IFTF
007554 005077 006032  MUX16: CLR      @DHMCSR          ;CLEAR CONTROL REGISTER
007560 042767 000340 170210          BIC      #340,PS          ;ENABLE INTERRUPTS
                  .IFT
                  MOV      #16.,R0          ;SET UP TO TEST 16 LINES
                  CLR      R1              ;START AT LINE 0
                  MOV      #1,SELMSK       ;INIT LINE SELECT MASK
007566 016701 006106          .IFF
                  MOV      LINE,R1
007572 012702 000020          .IFTF
                  MUX16A: MOV      #16.,R2          ;16 LINES
                  .IFT
                  BIT      SELMSK,LINSEL       ;THIS LINE SELECTED FOR TEST ?
                  BEQ      MUX16F             ;BR IF NOT
007576 010177 006010          .IFTF
007602 012777 000005 006004          MOV      R1,@DHMCSR          ;SELECT A LINE
                  MOV      #LINENA+RS,@DHMLSR    ;SET LINE ENABLE +RS

                  ;THESE TWO NOP'S ADDED FOR SOME DELAY FOR PDP-11/44
                  ;WITH CACHE MEMORY TURNED ON.
007610 000240          NOP
007612 000240          NOP          ;KR 10-JULY-84 REV E
                  ;KR 10-JULY-84 REV E
007614 005077 005772          MUX16B: CLR      @DHMCSR          ;CLEAR CONTROL REGISTER
007620 005005          CLR      R5              ;CLEAR EXPECTED RESULT
007622 017704 005766          MOV      @DHMLSR,R4          ;READ LINE STATUS
007626 117703 005760          MOV      @DHMCSR,R3          ;READ LINE NUMBER
007632 042703 177760          BIC      #177760,R3          ;CLEAR UNWANTED BITS
007636 020103          CMP      R1,R3              ;IF RECEIVED LINE=SELECTED LINE
007640 001002          BNE      MUX16C             ;EXPECT LINE ENABLE AND
007642 012705 000205          MOV      #LINENA+RS+RING,R5

007646 020405          MUX16C: CMP      R4,R5          ;RING IS SET
007650 001403          BEQ      MUX16D             ;COMPARE EXPECTED AND
007652 104001          ERRORL                   ;RECEIVED RESULTS
007654 104003          SCOPEF                   ;LINE STATUS ERROR
007656 007660          MUX16D: CALL     STEPER          ;UPDATE LINE COUNTER
007660 004767 003520          DEC      R2              ;CONTINUE IF ALL CHECKS
007664 005302          BNE      MUX16B             ;ARE NOT DONE FOR THIS LINE
007666 001354          MOV      #LINENA,R5          ;EXPECT LINE ENABLE
007670 012705 000001          MUX16E: MOV      R1,R3              ;ON SELECTED LINE
007674 010103          MOV      R1,@DHMCSR          ;SELECT LINE
007676 010177 005710          BIC      #RS,@DHMLSR          ;CLEAR REQUEST TO SEND
007702 042777 000004 005704          INCB     #0              ;DELAY FOR CABLE
007710 105227 000000

```

```
007714 001375          BNE      .-4          ;DITTO
007716 017704 005672  MOV      @DHMLSR,R4      ;READ LINE STATUS REGISTER
007722 020504          CMP      R5,R4          ;ONLY LINE ENABLE SHOULD BE
007724 001401          BEQ      MUX16F        ;SET ON THIS LINE
007726 104001          ERRORL          ;LINE STATUS ERROR

      .IFT
MUX16F: SCOPEF          ;CHECK FOR LOOP ON SAME DATA
MUX16A INC      R1          ;UPDATE LINE NUMBER
      CLR      @DHMLSR    ;CLEAR LINE STATUS REGISTER
      ASL      SELMSK    ;SHIFT MARK TO TEST NEXT LINE
      DEC      R0          ;CONTINUE IF ALL LINES NOT
      BNE      MUX16A    ;TESTED
      SCOPE          ;CHECK FOR ITERATIONS, LOOP

007730 104002          .IFF
MUX16F: SCOPE          ;CHECK FOR ITERATIONS, LOOP
      .ENDC
```

```

007732          MUXS2  17,SECTX,SECRX,+/SECONDARY RECEIVE IS/,+/SECONDARY TRANSMIT/
                ;VERIFY THAT SECONDARY RECEIVE IS SET IF "LINE ENABLE"
                ;AND SECONDARY TRANSMIT ARE SET FOR SELECTED LINE.

                .IFEQ  SINGLE
                TS  \NO,0
                .IFF
007732          TS      \N1,1
                ;REFERANCE NUMBER DEFINITION
007732          T107:   ;REFERENCE DESIGNATION
                N1=N1+1
                .IFTF
007732 005077 005654 MUX17: CLR  @DHMCSR           ;CLEAR CONTROL REGISTER
007736 042767 000340 170032 BIC  #340,PS       ;ENABLE INTERRUPTS

                .IFT
                MOV  #16.,R0           ;SET UP TO TEST 16 LINES
                CLR  R1                 ;START AT LINE 0
                MOV  #1,SELSK          ;INIT LINE SELECT MASK

                .IFF
                MOV  LINE,R1

                .IFTF
007750 012702 000020 MUX17A: MOV  #16.,R2           ;16 LINES
                .IFT
                BIT  SELSK,LINSEL       ;THIS LINE SELECTED FOR TEST ?
                BEQ  MUX17F             ;BR IF NOT

                .IFTF
007754 010177 005632 MUX17: MOV  R1,@DHMCSR           ;SELECT A LINE
007760 012777 000011 005626 MOV  #LINENA+SECTX,@DHMLSR       ;SET LINE ENABLE +SECTX

                ;THESE TWO NOP'S ADDED FOR SOME DELAY FOR PDP-11/44
                ;WITH CACHE MEMORY TURNED ON.
007766 000240          NOP
007770 000240          NOP
                ;KR 10-JULY-84 REV E
                ;KR 10-JULY-84 REV E

007772 005077 005614 MUX17B: CLR  @DHMCSR           ;CLEAR CONTROL REGISTER
007776 005005          CLR  R5           ;CLEAR EXPECTED RESULT
010000 017704 005610 MOV  @DHMLSR,R4       ;READ LINE STATUS
010004 117703 005602 MOV  @DHMCSR,R3       ;READ LINE NUMBER
010010 042703 177760 BIC  #177760,R3     ;CLEAR UNWANTED BITS
010014 020103          CMP  R1,R3       ;IF RECEIVED LINE=SELECTED LINE
010016 001002          BNE  MUX17C      ;EXPECT LINE ENABLE AND
010020 012705 000031 MOV  #LINENA+SECTX+SECRX,R5

                ;SECONDARY RECEIVE IS SET
010024 020405 MUX17C: CMP  R4,R5           ;COMPARE EXPECTED AND
010026 001403          BEQ  MUX17D      ;RECEIVED RESULTS
010030 104001          ERRORL          ;LINE STATUS ERROR
010032 104003          SCOPEF
010034 010036          MUX17D
010036 004767 003342 MUX17D: CALL  STEPER           ;UPDATE LINE COUNTER
010042 005302          DEC  R2           ;CONTINUE IF ALL CHECKS
010044 001354          BNE  MUX17B      ;ARE NOT DONE FOR THIS LINE
010046 012705 000001 MOV  #LINENA,R5       ;EXPECT LINE ENABLE
010052 010103 MUX17E: MOV  R1,R3       ;ON SELECTED LINE
010054 010177 005532 MOV  R1,@DHMCSR       ;SELECT LINE
010060 042777 000010 005526 BIC  #SECTX,@DHMLSR   ;CLEAR SECONDARY TRANSMIT
010066 105227 000000          INCB  #0     ;DELAY FOR CABLE

```

```
010072 001375          BNE      . 4          ;DITTO
010074 017704 005514  MOV      @DHMLSR,R4    ;READ LINE STATUS REGISTER
010100 020504          CMP      R5,R4         ;ONLY LINE ENABLE SHOULD BE
010102 001401          BEQ      MUX17F       ;SET ON THIS LINE
010104 104001          ERRORL          ;LINE STATUS ERROR

          .IFT
MUX17F: SCOPEF          ;CHECK FOR LOOP ON SAME DATA
        MUX17A
        INC      R1          ;UPDATE LINE NUMBER
        CLR      @DHMLSR    ;CLEAR LINE STATUS REGISTER
        ASL      SELMSK    ;SHIFT MARK TO TEST NEXT LINE
        DEC      R0         ;CONTINUE IF ALL LINES NOT
        BNE      MUX17A    ;TESTED
        SCOPE          ;CHECK FOR ITERATIONS, LOOP

010106 104002          .IFF
MUX17F: SCOPE          ;CHECK FOR ITERATIONS, LOOP
        .ENDC
```

```

2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21 010110
      000200
      000200
      000200
22 010110
      010110
      000201
      000005
23 010110 000340 167656
24 010112 012767 000340 167656
25 010120 104004
26 010122 016227
27 010124 022767 000176 005474
28 010132 001001
29 010134 104025
30 010136 012767 010154 001600
31 010144 012767 010152 172072
32 010152 104017
33
34 010154 104020
35
36 010156 010166
37 010160 010162
38 010162 104012
39 010164 000772
40
41
42
43
44
45
46 010166 104021
47
48
49
50 010170 010206
51

```

;MODEM CONTROL ON LINE TEST USING 103A TYPE MODEMS
;ANSWER STATION TO BE OPERATED IN AUTO ANSWER MODE
;THIS TEST VERIFIES THE CONNECT AND DISCONNECT SEQUENCES
;USING THE MODEM CONTROL TO CONTROL 103A TYPE MODEMS

;NOTE: IF THE DM11-AA IS NOT CONNECTED TO THE
;DISTRIBUTION PANEL, AN M974 DM11 MAINTENANCE JUMPER
;SHOULD BE INSTALLED IN SLOT B1 OR B3 OF THE DISTRIBUTION
;PANNEL TO PREVENT A POSSIBLE LONG SPACE
;DISCONNECT FROM HANGING UP THE MODEM

```

.MACRO COMMENT
.NLIST
N2=200
N=N2
XN=N2
.LIST
.ENDM
COMMENT
N2=200
N=N2
XN=N2
TS \N2,2
;REFERANCE NUMBER DEFINITION
T200:
N2=N2+1
;REFERENCE DESIGNATION
ST103A: RESET
MOV #340,PS
TYPE
MT103T
CMP #SWREG,SWR
BNE 1$
CNTLUU
1$: MOV #T103A,FATRET
MOV #ST103B,KRET
ST103B: GETLNS
;INITIALIZE INTERFACE
;DISABLE ALL INTERRUPTS
;TYPE "103A MODEM CONNECT-
;DISCONNECT TEST"
T103A: SETUP
;SET UP FOR FATAL ERROR
;SET UP FOR LINE CHANGE
;INPUT ORIGINATE AND
;AND ANSWER LINE NUMBERS
;SET UP TO RECEIVE INTERRUPTS
;WAIT FOR RING
;GO HERE IF RING OK
;GO HERE IF NO RING
;NO RING WITHIN 5 MINUTES
;SELECT NEW LINES AND REDIAL
T103A1: ERROR
BR ST103B
;CHECK FOR RING INTERRUPT ON SELECTED ANSWER LINE
;IF AN INCORRECT TRANSITION OCCURS, THE PROGRAM
;WILL TYPE AN ERROR MESSAGE, AND THE OPERATOR
;WILL BE REQUESTED TO RESELECT LINES AND REDIAL
T103B: CKRING
;CHECK FOR RING INTERRUPT
;ONLY ON ANSWER LINE
;AND NO TRANSITIONS ON
;ORIGINATE LINE
;GO HERE IF TRANSITIONS
;ARE CORRECT
T103C

```

52 010172 010176
53
54 010174 010202
55
56 010176 104014
57 010200 000207
58 010202 104014
59 010204 000762

T103B1
T103B2
T103B1: ERRORT
RTS PC
T103B2: ERRORT
BR ST103B

:GO HERE IF INCORRECT
:TRANSITION ON ANSWER LINE
:GO HERE IF INCORRECT TRANSITION
:ON ORIGINATE LINE
:TRANSITION ERROR ON ANSWER LINE
:CONTINUE CHECKING
:TRANSITION ERROR ON ORIGINATE LINE
:RESELECT LINES AND REDIAL

```

1
2
3 ;SET TERMINAL READY ON SELECTED ANSWER LINE
4 ;WAIT FOR TRANSITIONS TO OCCUR ON SELECTED LINES
5 010206 016777 005472 005376 T103C: MOV LINANS,@DHMCSR ;SET LINE COUNTER TO
6 ;ANSWER LINE NUMBER
7 010214 052777 000002 005372 BIS @TRMRDY,@DHMLSR ;SET TERMINAL READY ON
8 ;SELECTED ANSWER LINE
9 010222 104026 CKINTT
10 010224 104022 WAITRN ;WAIT FOR TRANSITIONS TO OCCUR
11
12 ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
13 ;SELECTED ORIGINATE AND ANSWER LINES
14
15 010226 104023 CKTRAN ;CHECK TRANSITIONS AND
16 ;STATUS ON SELECTED
17 ;ANSWER AND ORIGINATE LINES
18 010230 000143 CO+CS+LINENA+TRMRDY ;EXPECT CARRIER, CLEAR TO SEND,
19 ;LINE ENABLE AND TERMINAL
20 ;READY STATUS BITS SET ON
21 ;ANSWER LINE
22 010232 000143 CO+CS+LINENA+TRMRDY ;EXPECT CARRIER, CLEAR TO SEND,
23 ;LINE ENABLE , AND TERMINAL
24 ;READY STATUS BITS ON
25 ;ORIGINATE LINE
26 010234 100006 RINGF+XCO+XCS ;EXPECT CARRIER, CLEAR TO SEND
27 ;AND POSSIBLE RING TRANSITIONS
28 ;ON ANSWER LINE
29 010236 000006 XCO+XCS ;EXPECT CARRIER AND CLEAR
30 ;TO SEND TRANSITIONS ON
31 ;ORIGINATE LINE
32 010240 010252 T103D1 ;GO HERE ON ANSWER LINE STATUS ERROR
33
34 010242 010256 T103D2 ;GO HERE ON ORIGINATE LINE STATUS ERROR
35 010244 010262 T103D3 ;GO HERE ON ANSWER LINE TRANSITION ERROR
36 010246 010266 T103D4 ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
37 010250 010272 T103E ;GO TO NEXT TEST IF NO ERRORS
38 010252 104015 T103D1: ERRORS ;ANSWER LINE STATUS ERROR
39 010254 000207 RTS PC ;CONTINUE CHECKING
40 010256 104015 T103D2: ERRORS ;ORIGINATE LINE STATUS ERROR
41 010260 000207 RTS PC ;CONTINUE CHECKING
42 010262 104014 T103D3: ERRORT ;ANSWER LINE TRANSITION ERROR
43 010264 000207 RTS PC ;CONTINUE CHECKING
44 010266 104014 T103D4: ERRORT ;ORIGINATE LINE TRANSITION ERROR
45 010270 000207 RTS PC ;CONTINUE CHECKING

```

```

1
2
3
4
5
6
7
8
9
10
11 010272 104004          T103E: TYPE          ;TYPE 'STRIKE ANY TTY KEY
12 010274 016472          MDISC              ;TEST DISCONNECT"
13 010276 012767 000340 167472  MOV  #340,PS        ;LOCK OUT INTERRUPTS
14 010304 012777 011766 005274  MOV  #TRNTYP,@DHMVEC ;SET UP TO DETECT TRANSITIONS
15
16 010312 012767 010332 005412  MOV  #T103ES,RNGRET ;BEFORE DISCONNECT SEQUENCE STARTS
17
18 010320 012777 000140 005264  MOV  #SCNENA+INTENA,@DHMCSR ;SET UP DUMMY RETURN FOR
19 010326 005067 167444          CLR  PS            ;RING INTERRUPT
20 010332 005077 005262          CLR  @TKDBR        ;SET SCAN ENABLE AND INTERRUPT ENABLE
21 010336 105777 005254          T103ES: CLR  @TKCSR ;ALLOW INTERRUPTS
22 010342 100375          1$:  TSTB @TKCSR    ;WAIT FOR TTY TO HIT
23 010344 005777 005250          BPL  1$
24 010350 012767 000340 167420  TST  @TKDBR
25 010356 005077 005230          MOV  #340,PS      ;START DISCONNECT SEQUENCE
26 010362 016777 005314 005222  CLR  @DHMCSR      ;CLEAR CONROL REGISTER
27 010370 042777 000002 005216  MOV  LINORG,@DHMCSR ;SET LINE COUNTER TO SELECTED ORIGINATE LINE
28 010376 104026          BIC  #TRMRDY,@DHMLSR ;SET TERMINAL READY ON SELECTED LINE
29 010400 104022          CKINTT
          WAITRN      ;WAIT FOR TRANSITIONS TO OCCUR

```



```

1
2
3
4
5 010402 104023
6
7
8 010404 000003
9
10
11 010406 000001
12
13 010410 000006
14
15
16 010412 000006
17
18
19 010414 010426
20
21 010416 010432
22 010420 010436
23 010422 010442
24 010424 010446
25 010426 104015
26 010430 000207
27 010432 104015
28 010434 000207
29 010436 104014
30 010440 000207
31 010442 104014
32 010444 000207
33
34 010446
    010446
35 010446 104004
36 010450 016414
37 010452 005067 005162
38 010456 104026
39 010460 000167 177466
40

```

```

;CHECK FOR CORRECT STATUS AND TRANSITIONS ON SELECTED
;ORIGINATE AND ANSWER LINES

CKTRAN
;CHECK TRANSITIONS AND
;STATUS ON SELECTED
;ANSWER AND ORIGINATE LINES
;EXPECT LINE ENABLE AND
;TERMINAL READY STATUS BITS
;SET ON ANSWER LINE
;EXPECT LINE ENABLE STATUS BIT
;SET ON ORIGINATE LINE
;EXPECT CARRIER AND CLEAR
;TO SEND TRANSITIONS ON
;ANSWER LINE
;EXPECT CARRIER AND CLEAR
;TO SEND TRANSITIONS ON
;ORIGINATE LINE
;GO HERE ON ANSWER LINE STATUS ERROR

T103E1
T103E2
T103E3
T103E4
T103EN
;GO HERE ON ORIGINATE LINE STATUS ERROR
;GO HERE ON ANSWER LINE TRANSITION ERROR
;GO HERE ON ORIGINATE LINE TRANSITION ERROR
;GO TO NEXT TEST IF NO ERRORS
;ANSWER LINE STATUS ERROR
;CONTINUE CHECKING
;ORIGINATE LINE STATUS ERROR
;CONTINUE CHECKING
;ANSWER LINE TRANSITION ERROR
;CONTINUE CHECKING
;ORIGINATE LINE TRANSITION ERROR
;CONTINUE CHECKING

TS \N2,2
;REFERANCE NUMBER DEFINITION

T201:
N2=N2+1
T103EN: TYPE
MT103A
CLR TSTNO
CKINTT
JMP ST103B
;REFERENCE DESIGNATION
;TYPE " 103A TEST COMPLETE"
;CLEAR TEST NUMBER FOR LOOPING
;SELECT NEW LINE NUMBERS AND
;RESTART TEST

```

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17 010464
18 010464
19 010464
20 010466
21 010474
22 010476
23 010500
24 010506
25 010510
26 010512
27 010520
28 010526
29
30 010530
31
32 010532
33 010534
34 010536
35 010540

;MODEM CONTROL ON LINE TEST USING 202C TYPE MODEMS
;ANSWER STATION TO BE OPERATED IN AUTO-ANSWER MODE
;THIS TEST VERIFIES THE CONNECT AND DISCONNECT SEQUENCES
;USING THE MODEM CONTROL TO CONTROL 202C TYPE MODEMS

;ALSO TESTED ARE LINE TURN AROUND AND
;SECONDARY TRANSIT SECONDARY RECEIVE

.MACRO COMMENT
.NLIST
N3=300
N=N3
XN=N3
.LIST
.ENDM COMMENT

000300 N3=300
000300 N=N3
000300 XN=N3
TS \N3,3 ;REFERENCE NUMBER DEFINITION

010464 T300: ;REFERENCE DESIGNATION
000301 N3=N3+1
000005 ST202A: RESET ;INITIALIZE INTERFACE
012767 000340 167302 MOV #340,PS ;DISABLE ALL INTERRUPTS
104004 TYPE ;TYPE "202C MODEM CONNECT-
016300 MT202T ;DISCONNECT TEST"
022767 000176 005120 CMP #SWREG,SWR
001001 BNE 1$
104025 CNTLUU
012767 010530 001224 1$: MOV #T202A,FATRET ;SET UP FOR FATAL ERROR
012767 010526 171516 MOV #ST202B,KRET ;SET UP FOR LINE CHANGE
104017 ST202B: GETLNS ;INPUT ORIGINATE AND
;ANSWER LINE NUMBERS
104020 T202A: SETUP ;SET UP TO RECEIVE INTERRUPTS
;WAIT FOR RING
010542 T202B ;GO HERE IF RING OK
010536 T202A1 ;GO HERE IF NO RING
104012 T202A1: ERROR ;NO RING WITHIN 5 MINUTES
000772 BR ST202B ;SELECT NEW LINES AND REDIAL

```

```
36
37
38 ;CHECK FOR RING INTERRUPT ON SELECTED ANSWER LINE
39 ;IF AN INCORRECT TRANSITION OCCURS, THE PROGRAM
40 ;WILL TYPE AN ERROR MESSAGE, AND THE OPERATOR
41 ;WILL BE REQUESTED TO RESELECT LINES AND REDIAL
42 010542 104021          T202B: CKRING          ;CHECK FOR RING INTERRUPT
43                                     ;ONLY ON ANSWER LINE
44                                     ;AND NO TRANSITIONS ON
45                                     ;ORIGINATE LINE
46 010544 010562          T202C          ;GO HERE IF TRANSITIONS
47                                     ;ARE CORRECT
48 010546 010552          T202B1        ;GO HERE IF INCORRECT
49                                     ;TRANSITION ON ANSWER LINE
50 010550 010556          T202B2        ;GO HERE IF INCORRECT
51                                     ;TRANSITION ON ORIGINATE LINE
52 010552 104014          T202B1: ERRORT ;ANSWER LINE TRANSITION ERROR
53 010554 000207          RTS           PC      ;CONTINUE CHECKING
54 010556 104014          T202B2: ERRORT ;ORIGINATE LINE TRANSITION ERROR
55 010560 000762          BR           ST202B ;RESELECT LINES AND REDIAL
```

```

1
2
3
4
5
6 010562 016777 005116 005022 T202C: MOV LINANS,@DHMCSR ;SET LINE COUNTER TO ANSWER LINE
7 010570 052777 000002 005016 BIS #TRMRDY,@DHMLSR ;SET TERMINAL READY ON ANSWER LINE
8 010576 016777 005100 005006 T202J: MOV LINORG,@DHMCSR ;SET LINE COUNT'R TO ORIGINATE LINE
9 010604 052777 000004 005002 BIS #RS,@DHMLSR ;SET REQUEST TO SEND ON ORIGINATE LINE
10 010612 104026 CKINTT
11 010614 104022 WAITRN ;WAIT FOR TRANSITIONS TO OCCUR
12
13
14
15
16 010616 104023 CKTRAN ;CHECK TRANSITIONS AND STATUS
17 ;ON SELECTED ANSWER AND
18 ;ORIGINATE LINES
19 010620 000103 CO+LINENA+TRMRDY ;EXPECT CARRIER, LINE ENABLE
20 ;AND TERMINAL READY STATUS
21 ;BITS SET ON ANSWER LINE
22 010622 000147 RS+CO+CS+LINENA+TRMRDY ;EXPECT REQUEST TO SEND, CLEAR
23 ;TO SEND, CARRIER, LINE ENABLE
24 ;AND TERMINAL READY STATUS BITS
25 ;SET ON ORIGINATE LINE
26 010624 100004 RINGF+XCO ;EXPECT CARRIER AND POSSIBLE
27 ;RING TRANSITIONS ON
28 ;ANSWER LINE
29 010626 000006 XCO+XCS ;EXPECT CARRIER AND CLEAR
30 ;TO SEND TRANSITIONS ON
31 ;ORIGINATE LINE
32 010630 010642 T202D1 ;GO HERE ON ANSWER LINE STATUS ERROR
33 010632 010646 T202D2 ;GO HERE ON ORIGINATE LINE STATUS ERROR
34 010634 010652 T202D3 ;GO HERE ON ANSWER LINE STATUS ERROR
35 010636 010656 T202D4 ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
36 010640 010662 T202E ;GO TO NEXT TEST IF NO ERRORS
37 010642 104015 T202D1: ERRORS ;ANSWER LINE SATUS ERROR
38 010644 000207 RTS PC ;CONTINUE CHECKING
39 010646 104015 T202D2: ERRORS ;ORIGINATE LINE STATUS ERROR
40 010650 000207 RTS PC ;CONTINUE CHECKING
41 010652 104014 T202D3: ERRORT ;ANSWER LINE TRANSITION ERROR
42 010654 000207 RTS PC ;CONTINUE CHECKING
43 010656 104014 T202D4: ERRORT ;ORIGINATE LINE TRANSITION ERROR
44 010660 000207 RTS PC ;CONTINUE CHECKING

```

```

1
2
3
4
5 010662 016777 005016 004722 T202E: MOV LINANS,@DHMCSR ;SET LINE COUNTER TO ANSWER LINE
6 010670 052777 000010 004716 BIS #SECTX,@DHMLSR ;SET SECONDARY RECEIVE ON ANSWER LINE
7 010676 104026 CKINTT
8 010700 104022 WAITRN ;WAIT FOR TRANSITIONS TO OCCUR
9
10 ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
11 ;SELECTED ORIGINATE AND ANSWER LINES
12
13 010702 104023 CKTRAN ;CHECK TRANSITIONS AND STATUS
14 ;ON SELECTED ANSWER AND
15 ;ORIGINATE LINES
16 010704 000133 SECTX+CO+LINENA+TRMRDY+SECRX ;EXPECT SECONDARY TRANSMIT
17 ;SECONDARY RECEIVE, CARRIER
18 ;LINE ENABLE AND TERMINAL READY
19 ;STATUS BITS SET ON ANSWER LINE
20 010706 000167 SECRX+RS+CO+CS+LINENA+TRMRDY ;EXPECT SECONDARY RECEIVE,
21 ;REQUEST TO SEND, CLEAR TO SEND
22 ;CARRIER, LINE ENABLE AND
23 ;TERMINAL READY STATUS BITS
24 ;SET ON ORIGINATE LINE
25 010710 000001 XSCRX ;EXPECT SECONDARY RECEIVE
26 ;TRANSITION ON ANSWER LINE
27 010712 000001 XSCRX ;EXPECT SECONDARY RECEIVE
28 ;TRANSITION ON ORIGINATE LINE
29 010714 010726 T202E1: ERRORS ;GO HERE ON ANSWER LINE STATUS ERROR
30 010716 010732 T202E2: RTS PC ;GO HERE ON ORIGINATE LINE STATUS ERROR
31 010720 010736 T202E3: ERRORS PC ;GO HERE ON ANSWER LINE TRANSITION ERROR
32 010722 010742 T202E4: ERRORT PC ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
33 010724 010746 T202F ;GO TO NEXT TEST IF NO ERRORS
34 010726 104015 T202E1: ERRORS ;ANSWER LINE STATUS ERROR
35 010730 000207 RTS PC ;CONTINUE CHECKING
36 010732 104015 T202E2: ERRORS ;ORIGINATE LINE STATUS ERROR
37 010734 000207 RTS PC ;CONTINUE CHECKING
38 010736 104014 T202E3: ERRORT PC ;ANSWER LINE TRANSITION ERROR
39 010740 000207 RTS PC ;CONTINUE CHECKING
40 010742 104014 T202E4: ERRORT PC ;ORIGINATE LINE TRANSITION ERROR
41 010744 000207 RTS PC ;CONTINUE CHECKING

```

```

1
2
3
4
5
6 010746 016777 004730 004636 T202F: MOV LINORG,@DHMCSR ;SET LINE COUNTER TO ORIGINATE LINE
7 010754 042777 000004 004632 BIC @RS,@DHMLSR ;DROP REQUEST TO SEND
8 010762 016777 004716 004622 MOV LINANS,@DHMCSR ;SET LINE COUNTER TO ANSWER LINE
9 010770 042777 000010 004616 BIC @SECTX,@DHMLSR ;DROP SECONDARY RECEIVE
10 010776 104026 CKINTT
11 011000 104022 WAITRN ;WAIT FOR TRANSITIONS TO OCCUR
12
13 ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
14 ;SELECTED ORIGINATE AND ANSWER LINES
15
16 011002 104023 CKTRAN ;CHECK TRANSITIONS AND STATUS
17 ;ON SELECTED ANSWER AND
18 ;ORIGINATE LINES
19 011004 000003 LINENA+TRMRDY ;EXPECT LINE ENABLE AND
20 ;TERMINAL READY STATUS BITS
21 ;SET ON ANSWER LINE
22 011006 000003 LINENA+TRMRDY ;EXPECT LINE ENABLE AND
23 ;TERMINAL READY STATUS BITS
24 ;SET ON ORIGINATE LINE
25 011010 000005 XCO+XSCRX ;EXPECT CARRIER AND SECONDARY
26 ;RECEIVE TRANSITIONS ON
27 ;ANSWER LINE
28 011012 000007 XCO+XCS+XSCRX ;EXPECT CARRIER, CLEAR TO SEND
29 ;AND SECONDARY RECEIVE
30 ;TRANSITIONS ON ORIGINATE LINE
31 011014 011026 T202F2 ;GO HERE ON ANSWER LINE STAATUS ERROR
32 011016 011032 T202F3 ;GO HERE ON ORIGINATE LINE STATUS ERROR
33 011020 011036 T202F4 ;GO HERE ON ANSWER LINE TRANSITION ERROR
34 011022 011042 T202F5 ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
35 011024 011046 T202G ;GO TO NEXT TEST IF NO ERRORS
36 011026 104015 T202F2: ERRORS ;ANSWER LINE STATUS ERROR
37 011030 000207 RTS PC ;CONTINUE CHECKING
38 011032 104015 T202F3: ERRORS ;ORIGINATE LINE STATUS ERROR
39 011034 000207 RTS PC ;CONTINUE CHECKING
40 011036 104014 T202F4: ERRORT ;ANSWER LINE TRANSITION ERROR
41 011040 000207 RTS PC ;CONTINUE CHECKING
42 011042 104014 T202F5: ERRORT ;ORIGINATE LINE TRANSITION ERROR
43 011044 000207 RTS PC ;CONTINUE CHECKING

```

```

1
2
3
4
5
6 011046 016777 004632 004536 T202G: MOV LINANS,@DHMCSR ;SET LINE COUNTER TO ANSWER LINE
7 011054 052777 000004 004532 BIS #RS,@DHMLSR ;SET REQUEST TO SEND
8 011062 104026 CKINTT
9 011064 104022 WAITRN ;WAIT FOR TRANSITIONS TO OCCUR
10
11 ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
12 ;SELECTED ORIGINATE AND ANSWER LINES
13
14 011066 104023 CKTRAN ;CHECK TRANSITIONS AND STATUS
15 ;ON SELECTED ANSWER AND
16 ;ORIGINATE LINES
17 011070 000147 RS+CO+CS+LINENA+TRMRDY ;EXPECT LINE ENABLE, TERMINAL
18 ;READY, REQUEST TO SEND, CLEAR
19 ;TO SEND, AND CARRIER
20 ;STATUS BITS SET ON ANSWER LINE
21 011072 000103 CO+LINENA+TRMRDY ;EXPECT LINE ENABLE, TERMINAL
22 ;READY AND CARRIER STATUS
23 ;BITS SET ON ORIGINATE LINE
24 011074 000006 XCO+XCS ;EXPECT CARRIER AND CLEAR
25 ;TO SEND TRANSITIONS ON
26 ;ANSWER LINE
27 011076 000004 XCO ;EXPECT CARRIER TRANSITION
28 ;ON ORIGINATE LINE
29 011100 011112 T202G1 ;GO HERE ON ANSWER LINE STATUS ERROR
30 011102 011116 T202G2 ;GO HERE ON ORIGINATE LINE STATUS ERROR
31 011104 011122 T202G3 ;GO HERE ON ANSWER LINE TRANSITION ERROR
32 011106 011126 T202G4 ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
33 011110 011132 T202H ;GO TO NEXT TEST IF NO ERRORS
34 011112 104015 T202G1: ERRORS ;ANSWER LINE STATUS ERROR
35 011114 000207 RTS PC ;CONTINUE TESTING
36 011116 104015 T202G2: ERRORS ;ORIGINATE LINE STATUS ERROR
37 011120 000207 RTS PC ;CONTINUE TESTING
38 011122 104014 T202G3: ERRORT ;ANSWER LINE TRANSITION ERROR
39 011124 000207 RTS PC ;CONTINUE TESTING
40 011126 104014 T202G4: ERRORT ;ORIGINATE LINE TRANSITION ERROR
41 011130 000207 RTS PC ;CONTINUE TESTING

```



```

1
2
3           ;DROP REQUEST TO SEND ON ANSWER LINE
4           ;WAIT FOR TRANSITIONS TO OCCUR ON SELECTED LINES
5 011216 016777 004462 004366 T202I: MOV    LINANS,@DHMCSR           ;SET LINE COUNTER TO ANSWER LINE
6 011224 042777 000004 004362      BIC    #RS,@DHMLSR           ;CLEAR REQUEST TO SEND
7 011232 016777 004444 004352      MOV    LINORG,@DHMCSR        ;SET LINE COUNTER TO ORIGINATE LINE
8 011240 042777 000010 004346      BIC    #SECTX,@DHMLSR       ;CLEAR SECONDARY TRANSMIT
9 011246 104026
10 011250 104022      CKINTT
11                          WAITRN           ;WAIT FRO TRANSITIONS TO OCCUR
12
13           ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
14           ;SELECTED ORIGINATE AND ANSWER LINES
15 011252 104023      CKTRAN           ;CHECK TRANSITION S AND STATUS
16                          ;ON SELECTED ANSWE AND
17                          ;ORIGINATE LINES
18 011254 000003      LINENA+TRMRDY       ;EXPECT LINE ENABLE AND
19                          ;TERMINAL READY STATUS BITS SET
20                          ;ON ANSWER LINE
21 011256 000003      LINENA+TRMRDY       ;EXPECT LINE ENABLE AND
22                          ;TERMINAL READY STATUS BITS
23                          ;SET ON ORIGINATE LINE
24 011260 000007      XCO+XCS+XSCRX       ;EXPECT CARRIER, CLEAR TO SEND
25                          ;AND SECONDARY RECEIVE TRANSITIONS
26                          ;ON ANSWER LINE
27 011262 000005      XCO+XSCRX         ;EXPECT CARRIER AND SECONDARY
28                          ;RECEIVE TRANSITIONS ON
29                          ;ORIGINATE LINE
30 011264 011276      T202I2           ;GO HERE ON ANSWER LINE STATUS ERROR
31 011266 011302      T202I3           ;GO HERE ON ORIGINATE LINE STATUS ERROR
32 011270 011306      T202I4           ;GO HERE ON ANSWER LINE TRANSITIN ERROR
33 011272 011312      T202I5           ;GO HERE ON ORIGINATE LINE TRANSITION ERROR
34 011274 011316      T202J            ;GO TO NEXT TEST IF NO ERRORS
35 011276 104015      T202I2: ERRORS     ;ANSWER LINE STATUS ERROR
36 011300 000207      RTS             PC   ;CONTINUE CHECKING
37 011302 104015      T202I3: ERRORS     ;ORIGINATE LINE STATUS ERROR
38 011304 000207      RTS             PC   ;CONTINUE CHECKING
39 011306 104014      T202I4: ERRORT    ;ANSWE LINE TRANSITION ERROR
40 011310 000207      RTS             PC   ;CONTINUE CHECKING
41 011312 104014      T202I5: ERRORT    ;ORIGINATE LINE TRANSITION ERROR
42 011314 000207      RTS             PC   ;CONTINUE CHECKING

```

```

1
2
3 ;SET UP TO TEST DISCONNECT SEQUENCE
4 ;THE PROGRAM WILL REQUEST THE OPERATOR TO TYPE A CHARACTER
5 ;TO INITIATE THE DISCONNECT SEQUENCE
6 ;THE OPERATOR MAY MANUALLY SWITCH THE DATA SETS FROM
7 ;DATA TO TALK MODE AS MANY TIMES AS DESIRED
8 ;BEFORE THE SWITCH SEETIN IS MADE
9 ;ANY TRANSITIONS DETECTED DURING THIS TIME WILL BE
10 ;REPORTED BY TYPEOUT
11 011316 104004          T202J: TYPE          ;TYPE "STRIKE ANY TTY KEY
12 011320 016472          MDISC          ;TEST DISCONNECT"
13 011322 012767 000340 166446      MOV      #340,PS      ;LOCK OUT INTERRUPTS
14 011330 012777 011766 004250      MOV      #TRNTYP,@DHMVEC ;SET UP TO DETECT TRANSITIONS
15 011336 012767 011356 004366      MOV      #T202JS,RNGRET ;SET UP DUMMY RETURN FOR RING
16                                     ;FROM RING INTERRUPT
17 011344 012777 000140 004240      MOV      #SCNENA+INTENA,@DHMCSR ;ENABLE LINE SCANNER
18                                     ;START SCANNER
19 011352 005067 166420          CLR      PS          ;ENABLE INTERRUPTS
20 011356 005077 004236          T202JS: CLR      @TKDBR
21 011362 105777 004230          1$:  TSTB   @TKCSR
22 011366 100375          BPL      1$
23 011370 005777 004224          TST      @TKDBR
24
25 ;DISCONNECT SEQUENCE REQUESTED
26
27 011374 012767 000340 166374      MOV      #340,PS      ;LOCK OUT INTERRUPTS
28 011402 005077 004204          CLR      @DHMCSR      ;STOP SCANNER
29 011406 016777 004270 004176      MOV      LINORG,@DHMCSR ;SET LINE COUNTER TO SELECTED ORIGINATE LINE
30 011414 042777 000002 004172      BIC      #TRMRDY,@DHMLSR ;SET TERMINAL READY ON SELECTED LINE
31 011422 104024          WAITS
32 011424 104026          CKINTT
33 011426 104022          WAITRN          ;WAIT FOR TRANSITIONS TO OCCUR

```

```

1
2
3 ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON SELECTED
4 ;ORIGINATE AND ANSWER LINES
5 011430 104023 CKTRAN ;CHECK TRANSITIONS AND STATUS
6 ;ON SELECTED ASNWER AND
7 ;ORIGINATE LINES
8 011432 000003 LINENA*TRMRDY ;EXPECT LINE ENABLE AND
9 ;TERMINAL READY STATUS BITS
10 ;SET ON ANSWER LINE
11 011434 000001 LINENA ;EXPECT LINE ENABLE STATUS
12 ;BIT SET ON ORIGINATE LINE
13 011436 000000 0 ;EXPECT NO TRANSITIONS ON
14 ;ANSWER LINE
15 011440 000000 0 ;EXPECT NO TRANSITIONS ON
16 ;ORIGINATE LINE
17 011442 011454 T202J1 ;GO HERE IF ANSWER LINE STATUS ERROR
18 011444 011460 T202J2 ;GO HERE IF ORIGINATE LINE STATUS ERROR
19 011446 011464 T202J3 ;GO HERE IF ANSWER LINE TRANSITION ERROR
20 011450 011470 T202J4 ;GO HERE IF ORIGINATE LINE TRANSITIONS ERROR
21 011452 011474 T202JN ;GO TO END OF TEST IF NO ERRORS
22 011454 104015 T202J1: ERRORS ;ANSWER LINE STATUS ERROR0
23 011456 000207 RTS PC ;CONTINUE CHECKING
24 011460 104015 T202J2: ERRORS ;ORIGINATE LINE STATUS ERROR
25 011462 000207 RTS PC ;CONTINUE CHECKING
26 011464 104014 T202J3: ERRORT ;ANSWER LINE TRANSITION ERROR
27 011466 000207 RTS PC ;CONTINUE CHECKING
28 011470 104014 T202J4: ERRORT ;ORIGINATE LINE TRANSITION ERROR
29 011472 000207 RTS PC ;CONTINUE CHECKING
30
31 011474 104004 T202JN: TYPE ;TYPE "202C TEST COMPLETE"
32 011476 016443 MT202A
33 011500 104026 CKINTT
34 011502 000167 177020 JMP ST202B ;GET NEW LINE NUMBERS
35 ;RESTART TEST

```

```

1
2
3           ;DETECT AND RECORD TRANSITIONS ON SELECTED
4           ;ORIGINATE AND ANSWER LINES
5
6           ;TRANSITION DATA IS STORED IN LOCATIONS ANSFLG AND ORGFLG
7           ;FOR ANSWER AND ORIGINATE LINES RESPECTIVELY
8           ;FORMAT OF DATA IS (FOR BOTH LINES)
9
10          ;BIT0=1, SECONDARY RECEIVE CAUSED INTERRUPT
11          ;BIT1=1, CLEAR TO SEND CAUSED INTERRUPT
12          ;BIT2=1, CARRIER CAUSED INTERRUPT
13          ;BIT3=1, RING CAUSED INTERRUPT
14 011506  017704  004100          TRANS:  MOV      @DHMCSR,R4          ;GET LINE NUMBER AND
15                                          ;INTERRUPT FLAGS
16 011512  010405          MOV      R4,R5
17 011514  042705  177760          BIC      #177760,R5
18 011520  026705  004156          CMP      LINORG,R5          ;EXTRACT LINE NUMBER
19 011524  001411          BEQ      ORGTR              ;DID ORIGINATE LINE INTERRUPT
20 011526  026705  004152          CMP      LINANS,R5         ;IF YES, SERVICE
21 011532  001443          BEQ      ANSTR              ;DID ANSWER LINE INTERRUPT
22 011534  010577  004052          MOV      R5,@DHMCSR        ;IF YES, SERVICE
23 011540  017703  004050          MOV      @DHMLSR,R3
24 011544  104016          ERRORN
25 011546  000471          BR       FATEX              ;INTERRUPT ON INCORRECT LINE
26
27          ;RECORD TRANSITIONS FOR ORIGINATE LINE
28
29 011550  032704  100000          ORGTR:  BIT      #RINGF,R4          ;IF RING CAUSED INTERRUPT,
30 011554  001403          BEQ      ORGTR1              ;SET RING TRANSITION BIT
31 011556  052767  000010  004124          BIS      #10,ORGFLG
32 011564  032704  040000          ORGTR1: BIT      #COF,R4          ;IF CARRIER CAUSED INTERRUPT
33 011570  001403          BEQ      ORGTR2              ;SET CARRIER TRANSITION BIT
34 011572  052767  000004  004110          BIS      #4,ORGFLG
35 011600  032704  020000          ORGTR2: BIT      #CSF,R4          ;IF CLEAR TO SEND
36                                          ;CAUSED INTERRUPT
37 011604  001403          BEQ      ORGTR3              ;SET CLEAR TO SEND
38                                          ;TRANSITION BIT
39 011606  052767  000002  004074          BIS      #2,ORGFLG
40 011614  032704  010000          ORGTR3: BIT      #SECRXF,R4       ;IF SECONDARY RECEIVE
41                                          ;CAUSED INTERRUPT
42 011620  001403          BEQ      ORGTR4              ;SET SECONDARY RECEIVE
43 011622  052767  000001  004060          BIS      #1,ORGFLG          ;TRANSITION BIT
44 011630  032704  170000          ORGTR4: BIT      #RINGF+COF+CSF+SECRXF,R4
45
46 011634  001044          BNE      TRANEX              ;IF NO INTERRUPT FLAGS SET
47 011636  104016          ORGTRR: ERRORN              ;EXIT TRANSITION DETECTION
48 011640  000434          BR       FATEX

```

```

1
2
3
4 011642 032704 100000 ANSTR: BIT @RINGF,R4 ;IF RING CAUSED INTERRUPT,
5 011646 001403 BEQ ANSTR1 ;SET RING TRANSITION BIT
6 011650 052767 000010 004030 BIS @10,ANSFLG
7 011656 032704 040000 ANSTR1: BIT @COF,R4 ;IF CARRIER CAUSED INTERRUPT
8 011662 001403 BEQ ANSTR2 ;SET CARRIER TRANSITION BIT
9 011664 052767 000004 004014 BIS @4,ANSFLG
10 011672 032704 020000 ANSTR2: BIT @CSF,R4 ;IF CLEAR TO SEND
11 CAUSED INTERRUPT
12 011676 001403 BEQ ANSTR3 ;SET CLEAR TO SEND
13 TRANSITION BIT
14 011700 052767 000002 004000 BIS @2,ANSFLG
15 011706 032704 010000 ANSTR3: BIT @SECRXF,R4 ;IF SECONDARY RECEIVE
16 CAUSED INTERRUPT
17 011712 001403 BEQ ANSTR4 ;SET SECONDARY RECEIVE
18 011714 052767 000001 003764 BIS @1,ANSFLG ;TRANSITION BIT
19 011722 032704 170000 ANSTR4: BIT @RINGF+COF+CSF+SECRXF,R4
20 ;IF NO INTERRUPT FLAGS SET
21 011726 001007 BNE TRANEX ;EXIT TRANSITION DETECTION
22 011730 104016 ANSTRR: ERRORN
23 011732 005067 003702 FATEX: CLR TSTNO
24 011736 022626 POP2SP
25 011740 000177 000000 JMP @FATRET
26 011744 000000 FATRET: 0
27
28 ;EXIT TRANSITION DETECTION
29
30 011746 005704 TRANEX: TST R4 ;IF RING FLAG WAS SET
31 011750 100002 BPL .+6 ;SET UP SPECIAL RETURN
32 011752 016716 003754 MOV RRGRET,(SP)
33 011756 012777 000140 003626 TRANX1: MOV @SCNENA+INTENA,@DHMCSR ;RESTART SCANNER
34 011764 000002 RTI
35
36 ;TYPE TRANSITION DATA AND RETURN
37
38 011766 017767 003620 000756 TRNTYP: MOV @DHMCSR,DATA1
39 011774 017767 003614 000752 MOV @DHMLSR,DATA2
40 012002 104004 TYPE
41 012004 017114 MTRNDET
42 012006 104006 OCTASC
43 012010 012014 TRNTAB
44 012012 000761 BR TRANX1
45 012014 000002 TRNTAB: 2
46 012016 000006 6
47 012020 012752 DATA1
48 012022 000003 3
49 012024 012754 DATA2

```

```

1
2
3           ;INPUT ORIGINATE AND ANSWER LINES FROM TELETYPE KEYBOARD
4 012026 000005      GETLIN: RESET
5 012030 104013      INSTRG
6 012032 016351      MSELOR           ;TYPE "ORIGINATE LINE "
7 012034 000000      0                 ;AND GET LINE NUMBER
8 012036 000017      17
9 012040 015702      LINORG
10 012042 104013     INSTRG
11 012044 016375     MSELANS          ;TYPE "ANSWER LINE-"
12 012046 000000      0                 ;AND GET LINE NUMBER
13 012050 000017      17
14 012052 015704     LINANS
15 012054 104004     TYPE
16 012056 017005     MCRLF
17 012060 000002     RTI                 ;RETURN TO CALLING ROUTINE
18
19           ;INITIALIZE INTERFACE
20
21 012062 000005      SETUPS: RESET
22 012064 012767 000340 165704      MOV #340,PS           ;LOCK OUT ALL INTERRUPTS
23 012072 011605      MOV (SP),R5
24 012074 012567 000662      MOV (R5)+,NXTTS
25 012100 012567 000636      MOV (R5)+,ERR1
26 012104 010516      MOV R5,(SP)
27 012106 012777 006000 003476      MOV #CLRSCN+CLRMUX,@DHMCSR ;CLEAR LINE SCANNER AND MULTIPLEXER
28 012114 032777 000020 003470      SETUP1: BIT #BUSY,@DHMCSR ;WAIT FOR SCANNER TO CLEAR
29 012122 001374      BNE SETUP1
30 012124 005067 003502      CLR ERRFLG
31
32           ;ENABLE SELECTED LINES
33           ;SET TERMINAL READY ON SELECTED ORIGINATE LINE
34
35 012130 016777 003546 003454      SETUP2: MOV LINORG,@DHMCSR ;SET UP TO ENABLE ORIGINATE LINE
36                                     ;ORIGINATE LINE NUMBER
37 012136 012777 000003 003450      MOV #LINENA+TRMRDY,@DHMLSR ;SET LINE ENABLE AND
38                                     ;TERMINAL READY ON ORIGINATE LINE
39 012144 016777 003534 003440      MOV LINANS,@DHMCSR ;SET LINE COUNTER TO ANSWER LINE
40 012152 012777 000001 003434      MOV #LINENA,@DHMLSR ;SET LINE ENABLE ON ANSWER LINE
41
42           ;REQUEST OPERATOR TO DIAL SELECTED ANSWER TERMINAL
43           ;SET UP TO RECEIVE INTERRUPTS
44           ;START LINE SCANNER
45
46 012160 012777 011506 003420      MOV #TRANS,@DHMVEC ;SET UP INTERRUPT VECTOR
47                                     ;FOR TRANSITION DETECTION
48 012166 012777 000340 003414      MOV #340,@DHMLVL ;SET UP INTERRUPT SERVICE LEVEL
49 012174 012777 000140 003410      MOV #SCNENA+INTENA,@DHMCSR ;START SCANNER, ENABLE INTERRUPTS
50 012202 005067 003500      CLR ANSFLG ;CLEAR TRANSITION DETECTED FLAGS
51 012206 005067 003476      CLR ORGFLG
52 012212 012767 012242 003512      MOV #SETUP4,RNGRET ;SET UP RETURN FROM
53                                     ;DETECTION OF RING INTERRUPT
54 012220 104004      TYPE ;REQUEST OPERATOR TO DIAL
55 012222 016171      DIALM
56 012224 005067 165546      CLR PS ;CLEAR PROCESSOR STATUS WORD
57 012230 005067 003456      CLR TIME1 ;CLEAR TIMER

```

```

58 012234 012767 001000 003452      MOV      #1000,TIME2      ;SET UP FOR 5 MINUTE DELAY
59 012242 005767 003440      SETUP4: TST      ANSFLG      ;IF TRANSITION HAS OCCURED,
60 012246 001014                      BNE      SETUPB          ;EXIT WAIT LOOP
61 012250 005767 003434      TST      ORGFLG
62 012254 001011                      BNE      SETUPB
63 012256 005267 003430      INC      TIME1          ;ALLOW OPERATOR 5 MINUTES TO DIAL
64 012262 001367                      BNE      SETUP4
65 012264 005367 003424      DEC      TIME2
66 012270 001364                      BNE      SETUP4
67 012272 022626                      POP2SP
68 012274 000177 000442      JMP      @ERR1
69 012300 022626                      SETUPB: POP2SP
70 012302 000177 000454      JMP      @NXTTS
71 012306 012766 000340 000002      MOV      #340,+2(SP)
72 012314 000002                      RTI
73
74                      ;CHECK FOR RING INTERRUPT ON SELECTED ANSWER LINE
75
76 012316 011605      CKRNG:  MOV      (SP),R5
77 012320 012567 000436      MOV      (R5)+,NXTTS
78 012324 012567 000412      MOV      (R5)+,ERR1
79 012330 012567 000410      MOV      (R5)+,ERR2
80 012334 010516      MOV      R5,(SP)
81 012336 012705 000010      MOV      #10,R5          ;EXPECT RING ONLY ON ANSWER LINE
82 012342 016704 003340      MOV      ANSFLG,R4      ;GET ACTUAL TRANSITION DATA
83 012346 016703 003332      MOV      LINANS,R3      ;SET UP LINE NUMBER
84 012352 020504      CMP      R5,R4          ;DID RING CAUSE INTERRUPT
85 012354 001402      BEQ      CKRNG1        ;ON ANSWER LINE
86 012356 004777 000360      JSR      PC,@ERR1
87 012362 005005      CKRNG1: CLR      R5
88 012364 016704 003320      MOV      ORGFLG,R4
89 012370 016703 003306      MOV      LINORG,R3
90 012374 005704      TST      R4
91 012376 001403      BEQ      CKRNG2        ;IF TRANSITION OCCURED
92 012400 022626      POP2SP          ;ON ORIGINATE LINE, ERROR
93 012402 000177 000336      JMP      @ERR2
94 012406 022626      CKRNG2: POP2SP
95 012410 000177 000346      JMP      @NXTTS

```

```

1
2 012414 005067 003266          WAITR: CLR   ANSFLG
3 012420 005067 003264          CLR   ORGFLG
4 012424 012777 011506 003154   MOV   #TRANS,@DHMVEC
5 012432 012767 012452 003272   MOV   #WAITRR,RNGRET          ;SET UP FOR RETURN
6                                     ;FROM RING DETECTION
7 012440 012777 000140 003144   MOV   #SCNENA+INTENA,@DHMCSR ;START SCANNER
8 012446 005067 165324          CLR   PS
9 012452 005067 003234          WAITRR: CLR  TIME1
10 012456 012767 000025 003230  MOV   #25,TIME2
11 012464 005267 003222          WAITR1: INC  TIME1          ;WAIT FOR TRANSITIONS OF
12 012470 001375          BNE   WAITR1          ;CARRIER AND CLEAR TO SEND
13 012472 005367 003216          DEC  TIME2
14 012476 001372          BNE   WAITR1
15 012500 000002          RTI
16
17                                     ;CHECK FOR CORRECT STATUS AND TRANSITIONS ON
18                                     ;SELECTED ORIGINATE AND ANSWER LINES
19
20 012502 012767 000340 165266  CKTRN: MOV   #340,PS          ;LOCK OUT FURTHER INTERRUPTS
21 012510 005077 003076          CLR   @DHMCSR          ;STOP LINE SCANNER
22 012514 011605          MOV   (SP),R5
23 012516 012567 000230          MOV   (R5)+,DATA1
24 012522 012567 000226          MOV   (R5)+,DATA2
25 012526 012567 000224          MOV   (R5)+,DATA3
26 012532 012567 000222          MOV   (R5)+,DATA4
27 012536 012567 000200          MOV   (R5)+,ERR1
28 012542 012567 000176          MOV   (R5)+,ERR2
29 012546 012567 000174          MOV   (R5)+,ERR3
30 012552 012567 000172          MOV   (R5)+,ERR4
31 012556 012567 000200          MOV   (R5)+,NXTTS
32 012562 010516          MOV   R5,(SP)
33 012564 016705 000162          MOV   DATA1,R5
34 012570 016777 003110 003014  MOV   LINANS,@DHMCSR          ;SET LINE COUNTER TO ANSWER LINE
35 012576 017704 003012          MOV   @DHMLSR,R4          ;GET ACTUAL ANSWER LINE STATUS
36 012602 016703 003076          MOV   LINANS,R3
37 012606 020504          CMP   R5,R4          ;COMPARE
38 012610 001402          BEQ   CKTRN1
39 012612 004777 000124          JSR   PC,@ERR1
40 012616 016777 003060 002766  CKTRN1: MOV   LINORG,@DHMCSR          ;SET LINE COUNTER TO ORIGINATE LINE
41 012624 017704 002764          MOV   @DHMLSR,R4          ;GET ACTUAL ORIGINATE LINE STATUS
42 012630 016705 000120          MOV   DATA2,R5
43 012634 016703 003042          MOV   LINORG,R3
44 012640 020504          CMP   R5,R4          ;COMPARE
45 012642 001402          BEQ   CKTRN2
46 012644 004777 000074          JSR   PC,@ERR2

```



```

1
2
3
4
5 012650 105767 000103
6 012654 100003
7 012656 042767 000010 003022
8 012664 116704 003016
9 012670 116705 000062
10 012674 016703 003004
11 012700 020504
12 012702 001402
13 012704 004777 000036
14 012710 016704 002774
15 012714 016705 000040
16 012720 016703 002756
17 012724 020504
18 012726 001402
19 012730 004777 000014
20 012734 022626
21 012736 000177 000020
22 012742 000000
23 012744 000000
24 012746 000000
25 012750 000000
26 012752 000000
27 012754 000000
28 012756 000000
29 012760 000000
30 012762 000000

;CHECK FOR CORRECT TRANSITIONS ON
;SELECTED ORIGINATE AND ANSWER LINES

CKTRN2: TSTB DATA3+1
        BPL .+10
        BIC #10,ANSFLG
        MOVB ANSFLG,R4 ;GET TRANSITION DATA FOR
        MOVB DATA3,R5
        MOV LINANS,R3
        CMP R5,R4 ;DID CORRECT TRANSITIONS OCCUR
        BEQ CKTRN3
        JSR PC,@ERR3
CKTRN3: MOV ORGFLG,R4 ;GET TRANSITION DATA FOR
        MOV DATA4,R5
        MOV LINORG,R3
        CMP R5,R4 ;DID CORRECT TRANSITIONS OCCUR
        BEQ CKTRN4
        JSR PC,@ERR4
CKTRN4: POP2SP
        JMP @NXTTS

ERR1: 0
ERR2: 0
ERR3: 0
ERR4: 0
DATA1: 0
DATA2: 0
DATA3: 0
DATA4: 0
NXTTS: 0

```

```

1
2
3 ;END OF PASS
4 ;UPDATE PASS COUNT
5 ;TYPE END OF PASS MESSAGE
6 012764 EOP:
7 012764 005267 002646 INC PASCNT ;UPDATE PASS COUNT
8 012770 012767 000001 002642 MOV #1,TSTNO ;START AT FIRST TEST OF GROUP
9 012776 000005 RESET ;CLEAR THE WORLD
10 013000 005067 002724 CLR FILLA ;INIT COUNTER
11 013004 005367 002720 1$: DEC FILLA ;COUNT THE CTR
12 013010 001375 BNE 1$ ;BR TIL STALL TIMES OUT
13 013012 104004 TYPE ; RING BELL
14 013014 017271 MEPASS
15 013016 016701 165020 MOV 42,R1 ;ARE YOU ON ACT11?
16 013022 001521 BEQ TSTENT ;NO
17 013024 000005 RESET
18 013026 004711 LOGICAL: JSR PC,(R1)
19 013030 000240 NOP
20 013032 000240 NOP
21 013034 000240 NOP
22 013036 000240 NOP
23 013040 000167 000222 JMP TSTENT ;GET ADDRESS OF FIRST TEST
24
25 ;EMT DISPATCH SERVICE
26 ;ARGUMENT OF EMT IS EXTRACTED
27 ;AND USED AS OFFSET TO OBTAIN POINTER
28 ;TO SELECTED SUBROUTINE
29
30 013044 011646 EMTSRV: MOV (SP),-(SP) ;GET PC OF RETURN
31 013046 162716 000002 SUB #2,(SP) ;=PC OF EMT
32 013052 017616 000000 MOV @ (SP),(SP) ;GET EMT
33 013056 006316 EMTOK: ASL (SP) ;MULTIPLY EMT ARG BY 2
34 013060 042716 177001 BIC #177001,(SP) ;CLEAR UNWANTED BITS
35 013064 062716 017426 ADD #EMTTAB,(SP) ;POINTER TO SUBROUTINE ADDRESS
36 013070 017616 000000 MOV @ (SP),(SP) ;SUBROUTINE ADDRESS
37 013074 000136 JMP @ (SP)+ ;GO TO SUBROUTINE
38
39 013076 105777 002514 CKINT: TSTB @TKCSR
40 013102 100001 BPL 1$
41 013104 104027 KBDIN
42 013106 000002 1$: RTI
43

```

```

1
2
3
4
5
6
7
8
9 013110 005767 166176
10 013114 100022
11 013116 016746 164662
12 013122 012767 013142 164654
13 013130 005767 163724
14 013134 012667 164644
15 013140 000404
16 013142 022626
17 013144 012667 164634
18 013150 000402
19 013152 000167 000104
20 013156 000167 000104
21 013162
22 013162 005067 164610
23 013166 052777 000100 002422
24 013174 005767 002436
25 013200 001430
26 013202 005767 002424
27 013206 001404
28 013210 032777 002000 002410
29 013216 001021
30 013220 032777 040000 002400
31 013226 001041
32 013230 032777 004000 002370
33 013236 001011
34 013240 005367 002400
35 013244 001406
36 013246 016716 002370
37 013252 042777 000100 002336
38 013260 000002
39 013262 005267 002352
40 013266 016705 002346
41 013272 006305
42 013274 006305
43 013276 066705 002370
44 013302 011567 002334
45 013306 001626
46 013310 012516
47 013312 011567 002326
48 013316 005067 002310
49 013322 042777 000100 002266
50 013330 000002
51 013332 012767 000001 002304
52 013340 000742
53
54
55
56
57 013342 005767 002264

```

```

;END OF SUBTEST SERVICE
;CHECK FOR LOOP ON CURRENT TEST
;CHECK FOR ESCAPE TO NEXT TEST ON ERROR
;UPDATE ITERATION COUNT AND EXIT TO NEXT TEST IF 0

;TEST XOR FLAG (XFLAG) FOR EXISTANCE OF XOR TESTER.

LOOP: TST XFLAG ;IS THERE AN XOR TESTER OUT THERE ?
      BPL 4$ ;NO
      MOV 4,-(SP) ;SAVE 4
      MOV #1$,4 ;SET UP SVC ROUTINE
      TST 177060 ;GOT SOMETHING LIKE SLAVE SYNC
      MOV (SP)+,4 ;YOU BETCHUM
      BR 2$

1$: POP2SP ;RESTORE STACK
   MOV (SP)+,4 ;RESTORE 4
   BR 3$

2$: JMP LOOPX ;GO TO NEXT TEST
3$: JMP TSTENT ;GO
4$:

CLR PS
BIS #INTENA,@TKCSR
TST PASCNT ;1ST PASS
BEQ LOOPX ;NO ITERATIONS
5$: TST ERRFLG ;IF ERROR OCCURED FLAG=1,
    BEQ LOOPS ;CHECK FOR ESCAPE TO NEXT TEST
    BIT #SW10,@SWR ;IF SW10=1,
    BNE LOOPX ;ESCAPE TO NEXT TEST
    BIT #SW14,@SWR ;IF SW14=1,
    BNE LOOPL ;LOOP ON CURRENT TEST
    BIT #SW11,@SWR ;IF SW11=1,
    BNE LOOPX ;INHIBIT ITERATIONS
    DEC ICOUNT ;UPDATE ITERATION COUNT
    BEQ LOOPX ;IF ICOUNT=0, GO TO NEXT TEST
    MOV RETURN,(SP) ;SET UP FOR RETURN TO CURRENT TEST
    BIC #INTENA,@TKCSR

;RETURN TO CURRENT TEST
;UPDATE TEST NUMBER
LOOPX: INC TSTNO ;GET TEST NUMBER
TSTENT: MOV TSTNO,R5 ;MULTIPLY TEST NUMBER BY 4
        ASL R5
        ASL R5
        ADD TSTPNT,R5 ;GET POINTER FOR TEST ENTRY
        MOV (R5),RETURN ;GET STARTING ADDRESS OF NEXT TEST
        BEQ EOP ;IF ADDRESS=0, GO TO END OF PASS
        MOV (R5)+,(SP) ;PUT STARTING ADDRSS ON STACK
        MOV (R5),ICOUNT ;GET ITERATION COUNT FOR TEST
        CLR ERRFLG ;CLEAR ERROR OCCURED FLAG
        BIC #INTENA,@TKCSR
        RTI ;GO TO TEST
        MOV #1,ICOUNT ;SET UP TO EXIT TEST AFTER LOOP
        BR LOOPER ;GO TO LOOP SERVICE

;CHECK FOR LOOPING WITH SAME DATA
;CHECK FOR ESCAPE TO NEXT TEST ON ERROR

FREEZE: TST ERRFLG ;IF ERROR FLAG=0,

```

```
58 013346 001413          BEQ  FREEZX
59 013350 032777 002000 002250    BIT  #SW10,@SWR          ;DO NOT TEST FOR ESCAPE
60 013356 001341          BNE  LOOPX             ;IF SW10=1,
61 013360 032777 001000 002240    BIT  #SW09,@SWR          ;ESCAPE TO NEXT TEST
62 013366 001403          BEQ  FREEZX             ;IF SW09=1,
63 013370 017616 000000          MOV  @2,(SP),(SP)        ;FREEZE CURRENT DATA
64 013374 000002          RTI                                ;GET LOOPING ADDRESS
65 013376 062716 000002          FREEZX: ADD  #2,(SP)    ;LOOP
66 013402 000002          RTI                                ;CONTINUE IN CURRENT TEST
67
68
69
70 013404 052777 000400 002200    ;ROUTINE TO STEP THRU LINES
71 013412 032777 000020 002172    STEPER: BIS  #STEP,@DHMCSR
72 013420 001374          1$: BIT  #BUSY,@DHMCSR
73 013422 000207          BNE  1$
74          RETURN
75
```

```

1
2
3
4
5 013424 005067 002202      ERR:  CLR      ERRFLG      ;ALWAYS TYPE PC+2
6                                ;OF TEST THAT FAILED
7 013430 005067 000210      CLR      ERRMSG      ;NO MESSAGE
8 013434 005067 000216      CLR      ERTAB       ;NO TABLE OF DATA
9 013440 000451              BR      ERRGEN       ;OUTPUT ERROR MESSAGE
10
11
12
13
14
15
16
17
18
19
20
21
22
23 013442 005067 002164      ERR:  CLR      ERRFLG      ;ALWAYS OUTPUT ALL DATA
24 013446 012767 016132 000170  MOV      #MTRANE,ERRMSG ;TYPE "TRANSITION ERROR"
25 013454 012767 013750 000174  MOV      #ERTAB1,ERTAB  ;TABLE OF DATA
26 013462 000440              BR      ERRGEN       ;OUTPUT ERROR MESSAGE
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42 013464 005067 002142      ERRS: CLR      ERRFLG      ;ALWAYS OUTPUT ALL DATA
43 013470 012767 016101 000146  MOV      #MLINE1,ERRMSG ;TYPE "LINE ERROR"
44                                ;EXP REC LINE"
45 013476 012767 013766 000152  MOV      #ERTAB2,ERTAB ;TABLE OF DATA
46 013504 000427              BR      ERRGEN       ;OUTPUT ERROR MESSAGE

```

```

1
2
3           ;FATAL TRANSITION ERROR
4           ;FORMAT FOR FATAL ERROR TYPEOUT IS
5
6           ;XXXXXX FATAL ERROR
7           ;CSTAT LSTAT
8           ;AAAAAA BBB
9
10          ;WHERE XXXXXX=PC+2 OF CALL TO ERROR ROUTINE
11          ;   AAAAAA=RECEIVED CONTROL STATUS ON LINE THAT INTERRUPTED
12          ;   BBB=RECEIVED LINE STATUS ON LINE THAT INTERRUPTED
13 013506 005067 002120      ERRN:  CLR   ERRFLG           ;ALWAYS OUTPUT ALL DATA
14 013512 012767 017062 000124  MOV   #MFATAL,ERRMSG      ;TYPE "FATAL ERROR"
15                                     ;CSTAT LSTAT"
16 013520 012767 014004 000130  MOV   #ERTAB3,ERTAB      ;TABLE OF DATA
17 013526 000416             BR    ERRGEN           ;OUTPUT ERROR MESSAGE
18
19          ;"CONTROL STATUS" ERROR SERVICE
20          ;FORMAT FOR CONTROL STATUS ERROR IS
21
22          ;XXXXXX STATUS ERROR
23          ;EXP REC
24          ;AAAAAA BBBB
25
26          ;WHERE XXXXXX=PC+2 OF CALL TO ERROR ROUTINE
27          ;   AAAAAA=EXPECTED CONTROL STATUS AT TIME OF ERROR
28          ;   BBBB=RECEIVED(ACTUAL) CONTROL STATUS AT TIME OF ERROR
29
30
31 013530 012767 016013 000106  ERRCS: MOV   #MSTATE,ERRMSG      ;TYPE "STATUS ERROR"
32                                     ;"EXP REC"
33 013536 012767 014016 000112  MOV   #ERTAB4,ERTAB      ;TABLE OF DATA
34 013544 000407             BR    ERRGEN           ;OUTPUT DATA
35
36          ;LINE STATUS ERROR SERVICE
37
38          ;FORMAT FOR LINE STATUS ERROR IS
39
40          ;XXXX LINE ERROR
41          ;EXP REC LINE SEL
42          ;AAA DDD CC DD
43
44          ;WHERE XXXXXX=PC+2 OF CALL TO ERROR ROUTINE
45          ;   AAA=EXPECTED LINE STATUS AT TIME OF ERROR
46          ;   BBB=RECEIVED LINE STATUS AT TIME OF ERROR
47          ;   CC=LINE ON WHICH ERROR OCCURED
48          ;   DD=THE LINE ON WHICH THE PROGRAM WAS OPERATING
49
50
51 013546 012767 016044 000070  ERRLS: MOV   #MLINER,ERRMSG
52 013554 012767 014030 000074  MOV   #ERTAB5,ERTAB
53 013562 000400             BR    ERRGEN

```

```

1
2
3
4
5
6
7 013564 005067 164206
8 013570 012777 000100 002020
9 013576 032777 020000 002022
10 013604 001026
11 013606 021667 002052
12 013612 001402
13 013614 005067 002012
14 013620 104005
15 013622 005767 002004
16 013626 001007
17 013630 104006
18 013632 013742
19 013634 005767 000004
20 013640 001407
21 013642 104004
22 013644 000000
23 013646 005767 000004
24 013652 001402
25 013654 104006
26 013656 000000
27 013660 104007
28
29
30
31 013662 032777 100000 001736
32 013670 001406
33 013672 000000
34 013674 022767 000176 001724
35 013702 001001
36 013704 104025
37 013706 012767 000001 001716
38 013714 042777 000100 001674
39 013722 000002
40
41
42 013724 012767 015734 177712
43 013732 012767 014060 177716
44 013740 000711
45
46
;GENERAL ERROR HANDLER
;TYPE PC+2 OF TEST THAT FAILED
;TYPE ERROR MESSAGE (IF ANY)
;TYPE DATA RELATING TO FAILURE (IF ANY)

ERRGEN: CLR PS
MOV #INTENA,@TKCSR
BIT #SW13,@SWR ;IF SW13=1, DO NOT
BNE .3 ;TYPE ERROR MESSAGE
CMP (SP),SAVPC ;SAME ERROR AGAIN
BEQ .+6
CLR ERRFLG
SAV05P
TST ERRFLG ;IF ERROR OCCURED FLAG=1,
BNE .1 ;TYPE DATA ONLY
OCTASC ;TYPE PC+2 OF CALL TO ERROR ROUTINE
ERTAB0
TST ERRMSG
BEQ .2
TYPE ;TYPE ERROR MESSAGE

ERRMSG: 0
.1: TST ERTAB
BEQ .2
OCTASC ;TYPE DATA
ERTAB: 0
.2: RES05 ;RESTORE R0 R5

;ERROR HALT SERVICE
.3: BIT #SW15,@SWR ;IF SW15=0, DO NOT
BEQ .4 ;HALT ON ERROR
HALT ;HALT AND DISPLAY ADDRESS OF FAILING TEST
CMP #SWREG,SWR
BNE .4
CNTLUU
.4: MOV #1,ERRFLG ;SET ERROR OCCURED FLAG
BIC #INTENA,@TKCSR ;RETURN TO TEST
RTI

;TIMEOUT ERROR WAITING FOR INTERRUPT ON TEST 33
ERRQ: MOV #MNOINT,ERRMSG
MOV #ERTAB6,ERTAB ;TYPE LN#,CSR,LSR+MSG
BR ERRGEN ;OUTPUT DATA

```

```

1                                     ;TABLE S OF DATA FOR ERROR TYPEOUT
2
3                                     ;TABLE FOR TRANSITION STATUS ERROR
4
5 013742 000001      ERTAB0: 1
6 013744 000006      6
7 013746 015664      SAVPC
8 013750 000003      ERTAB1: 3
9 013752 000002      2
10 013754 015660      SAVR5      ;CONTAINS EXPECTED TRANSITION STATUS
11 013756 000002      2
12 013760 015656      SAVR4      ;CONTAINS RECEIVED TRANSITION STATUS
13 013762 000002      2
14 013764 015654      SAVR3      ;CONTAINS NUMBER OF LINE WHERE ERROR OCCURED
15 013766 000003      ERTAB2: 3
16 013770 000003      3
17 013772 015660      SAVR5      ;CONTAINS EXPECTED LINE STATUS
18 013774 000003      3
19 013776 015656      SAVR4      ;CONTAINS RECEIVED LINE STATUS
20 014000 000002      2
21 014002 015654      SAVR3      ;CONTAINS NUMBER OF LINE WHERE ERROR OCCURED
22 014004 000002      ERTAB3: 2
23 014006 000006      6
24 014010 015656      SAVR4
25 014012 000003      3
26 014014 015654      SAVR3
27 014016 000002      ERTAB4: 2
28 014020 000006      6
29 014022 015660      SAVR5      ;CONTAINS EXPECTED CONTROL STATUS
30 014024 000006      6
31 014026 015656      SAVR4      ;CONTAINS RECEIVED CONTROL STATUS
32 014030 000004      ERTAB5: 4
33 014032 000003      3
34 014034 015660      SAVR5      ;CONTAINS EXPECTED LINE STATUS
35 014036 000003      3
36 014040 015656      SAVR4      ;CONTAINS RECEIVED LINE STATUS
37 014042 000002      2
38 014044 015654      SAVR3      ;CONTAINS NUMBER OF LINE WHERE ERROR OCCURED
39 014046 000002      2
40 014050 015650      SAVR1      ;CONTAINS NUMBER OF LINE UNDER TEST
41
42
43 014052 000001      SWRTB: 1
44 014054 000006      6
45 014056 000176      SWREG
46
47 014060 000003      ERTAB6: 3
48 014062 000003      3
49 014064 015656      SAVR4      ;FAILING LINE #
50 014066 000006      6
51 014070 015652      SAVR2      ;CSR OF DEVICE
52 014072 000006      6
53 014074 015654      SAVR3      ;LSR OF DEVICE
54

```



```

1
2
3
4
5 014076 017605 000000
6 014102 062716 000002
7 014106 012767 000010 000322
8 014114 012704 017311
9 014120 012567 001542
10 014124 012567 001540
11 014130 013567 000276
12 014134 104010
13 014136 005367 001524
14 014142 001370
15 014144 112714 000100
16 014150 005767 000112
17 014154 001002
18 014156 104004
19 014160 017307
20 014162 000002
21
22
23
24 014164 005067 000072
25 014170 012767 000001 000066
26 014176 104004
27 014200 017246
28 014202 052767 000001 000056
29 014210 104006
30 014212 014052
31 014214 104004
32 014216 017311
33 014220 104013
34 014222 017257
35 014224 000000
36 014226 177777
37 014230 014262
38 014232 126727 000724 000015
39 014240 001403
40 014242 016777 000014 001356
41 014250 005067 000010
42 014254 005067 000006
43 014260 000002
44 014262 000000
45 014264 000000
46 014266 000000
47
48
;CONVERT OCTAL TO ASCII AND
;OUTPUT ON TELETYPE
OCTASN: MOV @ (SP),R5 ;GET POINTER TO TABLE OF DATA
ADD @2,(SP)
MOV #10,RADIX
MOV #MBCD+2,R4
MOV (R5)+,WRDCNT ;SET UP POINTER FOR CONVERTED DATA
OCTAS1: MOV (R5)+,CHRCNT ;GET NUMBER OF WORDS TO BE CONVERTED
MOV @ (R5)+,BINWRD ;GET NUMBER OF DIGITS IN WORD
CONVERT ;GET DATA TO BE CONVERTED
DEC WRDCNT ;CONVERT TO ASCII
BNE OCTAS1 ;IF ALL DATA IS NOT CONVERTED
MOVB #100,(R4) ;CONTINUE
TST SMLN ;PUT TERMINATOR AT END OF MESSAGE
BNE 1$
TYPE ;OUTPUT CONVERTED DATA
MBCD ;TO TELETYPE
RTI ;RETURN TO CALLING ROUTINE

1$:
CNTLU: CLR TMP1
MOV #1,TMP2
TYPE
$SWREQ
BIS #1,SMLN
OCTASC
SWRTB
TYPE
MBCD+2
INSTRG
$NEWIS
0
177777
TMP1
CMPB INBUF,#15
BEQ 1$
MOV TMP1,@SWR
1$:
CLR TMP2
CLR SMLN
RTI
TMP1: 0
TMP2: 0
SMLN: 0

```

```

1
2
3           ;INTEGER BINARY TO ASCII CONVERSION COMMON ROUTINE
4 014270 016700 001374      BINASC: MOV     CHRCNT,R0           ;SET UP COUNT FOR DIGITS TO BE CONVERTED
5 014274 012701 017412      MOV     @TEMPTAB,R1        ;SET UP POINTER FOR TEMPORARY STORAGE
6 014300 104011              BINASA: EXTRACT                ;EXTRACT ONE DIGIT
7 014302 062767 000060 000124 ADD     #60,DIGIT          ;CONVERT FROM BCD TO ASCII
8 014310 116721 000120      MOVVB  DIGIT,(R1).         ;STORE DIGIT
9 014314 005300              DEC     R0                 ;IF ALL DIGITS NOT DONE,
10 014316 001370              BNE    BINASA             ;CONTINUE
11 014320 114124              BINASB: MOVB   (R1),(R4).    ;REVERSE ORDER OF DIGITS
12 014322 005367 001342      DEC     CHRCNT            ;IF ALL CHARACTERS ARE NOT
13 014326 001374              BNE    BINASB             ;IN ORDER, CONTINUE
14 014330 112724 000040      MOVVB  #40,(R4).         ;INSERT SPACE AFTER LAST DIGIT
15 014334 000002              RTI                       ;RETURN TO CALLING ROUTINE
16
17           ;SINGLE PRECISION UNSIGNED DIVIDE LOOP
18
19 014336 005067 000072      DIVI:  CLR     DIVIDH      ;DIVIDE
20 014342 026767 000066 000066 DIVIU:  CMP     DIVIDH,DIVIS ;
21 014350 103027              BHIS   DIVIB              ;
22 014352 012767 000021 000032 MOV     #17.,DIVCNT        ;
23 014360 000407              BR     DIVIC              ;
24 014362 026767 000046 000046 DIVIA:  CMP     DIVIDH,DIVIS ;
25 014370 103403              BLO   DIVIC              ;
26 014372 166767 000040 000034 SUB     DIVIS,DIVIDH       ;
27 014400 006167 000026      DIVIC:  ROL     DIVIDL      ;
28 014404 006167 000024      ROL     DIVIDH             ;
29 014410 005327              DEC     (PC).             ;
30 014412 000000              DIVCNT: 0                  ;
31 014414 001362              BNE    DIVIA              ;
32 014416 006067 000012      ROR     DIVIDH             ;
33 014422 005167 000004      COM     DIVIDL             ;
34 014426 000002              RTI                       ;
35 014430 000000              DIVIB:  HALT              ;
36 014432 000000              DIVIDL: 0                  ;
37 014434 000000              DIVIDH: 0                  ;
38 014436 000000              DIVIS:  0                  ;
39
40           ;SAVE PC OF TEST THAT FAILED AND R0 R5
41
42 014440 016667 000004 001216 SV05P:  MOV     4(SP),SAVPC
43
44           ;SAVE R0-R5
45
46 014446 010567 001206      SV05:  MOV     R5,SAVR5
47 014452 010467 001200      MOV     R4,SAVR4
48 014456 010367 001172      MOV     R3,SAVR3
49 014462 010267 001164      MOV     R2,SAVR2
50 014466 010167 001156      MOV     R1,SAVR1
51 014472 010067 001150      MOV     R0,SAVR0
52 014476 000002              RTI

```

```

1
2
3
4 014500 016700 001142      RS05:  MOV     SAVR0,R0
5 014504 016701 001140      MOV     SAVR1,R1
6 014510 016702 001136      MOV     SAVR2,R2
7 014514 016703 001134      MOV     SAVR3,R3
8 014520 016704 001132      MOV     SAVR4,R4
9 014524 016705 001130      MOV     SAVR5,R5
10 014530 000002              RTI
11
12
13
14 014532 017605 000000      TYPER:  MOV     @ (SP),R5      ;GET POINTER TO MESSAGE (ON STACK)
15 014536 062716 000002      ADD     @2,(SP)              ;CORRECT STACK FOR RETURN
16 014542 105777 001054      TYPERA: TSTB   @TPCSR        ;WAIT FOR TELEPRINTER READY
17 014546 100375              BPL     TYPERA
18 014550 122765 000012 177777      CMPB   @12,-1(R5)           ;WAS LAST ONE A L.F. ??
19 014556 001405              BEQ     1$                  ;BR IF YES
20 014560 122765 000015 177777      CMPB   @15,-1(R5)           ;WAS LAST ONE A C.R. ??
21 014566 001401              BEQ     1$                  ;BR IF YES
22 014570 000402              BR      2$                  ;CONTINUE IF NEITHER
23 014572 004767 000044      1$:    JSR     PC,TYFILL      ;GO OUT PUT FILLERS
24 014576 122715 000100      2$:    CMPB   @100,(R5)      ;IF CHARACTER IS NOT TERMINATOR, TYPE IT
25 014602 001001              BNE     TYPER1
26 014604 000002              RTI
27 014606 122715 000042      TYPER1: CMPB   @42,(R5)      ;CHARACTER IS TERMINATOR, EXIT
28 014612 001406              BEQ     TYPECL              ;IF CHARACTER=42,
29 014614 122715 000045      CMPB   @45,(R5)            ;TYPE LINE FEED
30 014620 001403              BEQ     TYPECL              ;IF CHARACTER=45,
31 014622 112577 000776      TYPER2: MOVB  (R5)+,@TPDBR   ;TYPE CARRIAGE RETURN
32 014626 000745              BR      TYPERA              ;GET CHARACTER
33 014630 142715 000040      TYPECL: BICB   @40,(R5)      ;TYPE IT
34 014634 152715 000010      BISB   @10,(R5)           ;CONVERT CODE OF 42 OR 45
35 014640 000770              BR      TYPER2              ;TO 12 OR 15
36
37
38
39
40 014642 116767 001060 001060  TYFILL: MOVB  FILL,FILLA      ;GET FILL COUNT
41 014650 116777 001053 000746  1$:    MOVB  FILL+1,@TPDBR     ;OUT PUT ONE FILLER
42 014656 105777 000740      2$:    TSTB   @TPCSR        ;WAIT FOR TTY TO FINISH OUTPUT
43 014662 100375              BPL     2$                  ;BR IF TTY NOT DONE
44 014664 105367 001040      DECB   FILLA              ;COUNT ONE FILLER
45 014670 001367              BNE     1$                  ;BR TIL ALL DONE
46 014672 000207              RTS     PC                  ;RETURN TO CALLER ABOVE
47
48
49
50
51
52
53
54 014674
55 014674 011605      INSTR:  MOV     (SP),R5      ;GET POINTER TO ARGUMENTS
56 014676 012567 000020      MOV     (R5)+,MSG        ;GET MESSAGE TO BE TYPED
57 014702 012567 000246      MOV     (R5)+,LOLIM     ;GET LOWER LIMIT

```

```

58 014706 012567 000244      MOV      (R5)+,HILIM      ;GET UPPER LIMIT
59 014712 012567 000242      MOV      (R5)+,STORE     ;GET DATA STORAGE LOCATION
60 014716 010516              MOV      R5,(SP)        ;RESTORE STACK
61 014720 104004              INSTR1: TYPE           ;TYPE MESSAGE
62 014722 000000              MSG:      C
63 014724 012704 015162      MOV      @INBUF,R4      ;SET UP CHARACTER INPUT BUFFER
64 014730 012703 000007      MOV      @7,R3         ;SET UP INPUT COUNT
65 014734 105777 000656      INSTRB: TSTB          @TKCSR      ;WAIT FOR CHARACTER
66 014740 100375              BPL     INSTRB
67 014742 005067 165300      INSTRB: CLR          SINTFL
68 014746 017767 000646      MOV      @TKDBR,TMP1    177306
69 014754 142767 000200      BICB    @200,TMP1      177300
70 014762 116714 177274      MOVB    TMP1,(R4)
71 014766 121427 000007      CMPB    (R4),@7
72 014772 001420              BEQ     INSTR
73 014774 121427 000015      CMPB    (R4),@15      ;IS CHARACTER TERMINATOR
74 015000 001420              BEQ     INSTR2        ;IF IT IS, CONVERT INPUT STRING
75 015002 121427 000025      CMPB    (R4),@25
76 015006 001003              BNE     1$
77 015010 005067 177246      CLR     TMP1
78 015014 000741              BR      INSTR1
79 015016 112477 000602      1$:     MOVB    (R4)+,@TPDBR
80 015022 105777 000574      INSTRC: TSTB          @TPCSR
81 015026 100375              BPL     INSTRC
82 015030 005303              DEC     R3
83 015032 001340              BNE     INSTRB
84 015034 104004              INSTRE: TYPE
85 015036 017001              MQM
86 015040 000727              BR      INSTR1
87
88                               ;CONVERT ASCII STRING TO OCTAL
89
90 015042 104004              INSTR2: TYPE
91 015044 017005              MCRLF
92 015046 012704 015162      MOV      @INBUF,R4      ;GET POINTER TO ASCII STRING
93 015052 005003              CLR     R3
94 015054 122714 000015      CMPB    @15,(R4)      ;IS TERMINATOR FIRST
95                               ;CHARACTER IN STRING
96 015060 001431              BEQ     CHCK
97 015062 121427 000060      INSTRD: CMPB          (R4),@60    ;IS CHARACTER OCTAL DIGIT
98 015066 002762              BLT     INSTR          ;IF 67>=CHAR>=60
99 015070 121427 000067      CMPB    (R4),@67      ;CHARACTER IS OCTAL DIGIT
100 015074 003357              BGT     INSTR
101 015076 142714 000060      BICB    @60,(R4)      ;STRIP ASCII
102 015102 152403              BISB    (R4)+,R3      ;GENERATE OCTAL NUMBER
103 015104 121427 000015      CMPB    (R4),@15      ;IF END OF STRING, CHECK LIMITS
104 015110 001404              BEQ     INSTR3
105 015112 006303              ASL     R3
106 015114 006303              ASL     R3             ;MULTIPLY DIGIT BY 10 (OCTAL)
107 015116 006303              ASL     R3
108 015120 000760              BR      INSTRD
109
110                               ;GET NEXT DIGIT
111                               ;TEST NUMBER TO SEE IF IT IS WITHIN LIMITS
112 015122 020367 000030      INSTR3: CMP          R3,HILIM    ;TEST HI LIMIT
113 015126 101342              BHI     INSTR          ;IF R3>HILIM, ERROR
114 015130 020367 000020      CMP     R3,LOLIM      ;TEST LOW LIMIT

```

```

115 015134 103737
116 015136 010377 000016
117 015142 000002
118 015144 005767 177114
119 015150 001731
120 015152 000002
121 015154 000000
122 015156 000000
123 015160 000000
124 015162 000000
125 015204
126
127
128
129 015204 010046
130 015206 010146
131 015210 010246
132 015212 010346
133 015214 010446
134 015216 010546
135 015220 016746 162600
136 015224 010667 000432
137 015230 012767 015242 162566
138 015236 000000
139 015240 000776
140
141
142
143 015242 016706 000414
144 015246 012605
145 015250 012604
146 015252 012603
147 015254 012602
148 015256 012601
149 015260 012600
150 015262 012767 015204 162534
151 015270 005726
152 015272 104004
153 015274 017160
154 015276 005767 164514
155 015302 001002
156 015304 000167 164012
157 015310 104004
158 015312 017200
159 015314 012746 000340
160 015320 005746
161 015322 000167 175740
162
163
164
165 015326 016746 162466
166 015332 016746 162464
167 015336 012767 015530 162454
168 015344 012767 000340 162450
169 015352 012767 000300 175372
170 015360 012767 000302 175366
171 015366 016777 175362 175356 1$:

```

BLO INSTER ;IF R3<LOLIM, ERROR
 MOV R3,@STORE ;STORE NUMBER
 RTI ;EXIT
 CHCK: TST TMP2
 BEQ INSTER
 RTI
 LOLIM: 0
 HILIM: 0
 STORE: 0
 INBUF: 0
 .+.20
 ;ENTER HERE ON POWER FAILURE
 PFAIL: MOV R0,-(SP) ;SAVE R0-R5 ON PROCESSOR STACK
 MOV R1,-(SP)
 MOV R2,-(SP)
 MOV R3,-(SP)
 MOV R4,-(SP)
 MOV R5,-(SP)
 MOV 24,-(SP)
 MOV SP,SAVSP ;SAVE STACK POINTER
 MOV #RESTART,24 ;SET UP FOR POWER UP TRAP
 HALT ;HALT ON POWER DOWN NORMAL
 BR .-2
 ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
 RESTAR: MOV SAVSP,SP ;RESTORE STACK POINTER
 MOV (SP)+,R5 ;RESTORE R0-R5
 MOV (SP)+,R4
 MOV (SP)+,R3
 MOV (SP)+,R2
 MOV (SP)+,R1
 MOV (SP)+,R0
 MOV #PFAIL,24 ;SET UP FOR POWER FAILURE
 POP1SP
 TYPE
 MPFAIL
 TST TIPFLG
 BNE RESTA1
 JMP STARTO
 RESTA1: TYPE
 MPF1
 MOV #340,-(SP)
 PUSH1SP
 JMP TSTENT
 ;THE FOLLOWING AUTO VECTORS USING THE FIRST BASE ADDRESS
 XOR: MOV 20,-(SP) ;SAVE 20
 MOV 22,-(SP) ;SAVE 22
 MOV #2\$,20 ;IOT INTR VECTOR
 MOV #340,22 ;IOT INTR LVL
 MOV #300,DATA1
 MOV #302,DATA2
 1\$: MOV DATA2,@DATA1

```

172 015374 012777 000004 175352      MOV      #IOT,@DATA2      ;IOT TRAP
173 015402 062767 000004 175342      ADD      #4,DATA1
174 015410 062767 000004 175336      ADD      #4,DATA2
175 015416 026727 175330 001000      CMP      DATA1,#1000
176 015424 001360                      BNE      1$
177 015426 012767 000000 000204      MOV      #0,TSTNO        ;SET UP DEFAULT
178 015434 012767 017542 000230      MOV      #TSTBO,TSTPNT  ;
179 015442 052767 000340 162326      BIS      #340,PS        ;PREVENT INTERRUPTS
180 015450 005077 000136                      CLR      @DHMCSR        ;
181 015454 012777 000100 000130      MOV      #INTENA,@DHMCSR ;SET INTERRUPT ENABLE
182 015462 042767 000340 162306      BIC      #340,PS ;ALLOW INTERRUPTS
183 015470 052777 000200 000114      BIS      #DONE,@DHMCSR  ;SET DONE..AND INTERRUPT
184 015476 000240                      NOP
185 015500 012667 162316                      MOV      (SP)+,22      ;YOU DIDN'T INTERRUPT ?
186 015504 012667 162310                      MOV      (SP)+,20      ;RESTORE 20 & 22
187 015510 005077 000076                      CLR      @DHMCSR      ;STOP ALL INTERRUPT
188 015514 052767 000340 162254      BIS      #340,PS      ;
189 015522 104012                      ERROR
190 015524 000000                      HALT      ;YOU SHOULD HAVE INTERRUPTED
191 015526 000426                      BR       3$
192 015530 011667 000052                      MOV      (SP),DHMVEC   ;EXTRACT VECTOR +4
193 015534 162767 000002 000044      SUB      #2,DHMVEC     ;CREATE LVL
194 015542 016767 000040 000040      MOV      DHMVEC,DHMLVL ;SAVE
195 015550 162767 000002 000030      SUB      #2,DHMVEC     ;CREATE AND SAVE VEC
196 015556 012767 000340 162212      MOV      #340,PS      ;PREVENT INTERRUPTS
197 015564 005077 000022                      CLR      @DHMCSR      ;
198 015570 022626                      POP2SP
199 015572 022626                      POP2SP
200 015574 012667 162222                      MOV      (SP)+,22      ;RESTORE 22
201 015600 012667 162214                      MOV      (SP)+,20      ;RESTORE 20
202 015604 000207                      RTS      PC
203

```

```

1
2
3
4 015606 000300          DHMVEC: 300
5 015610 000302          DHMLVL: 302
6 015612 170500          DHMCSR: 170500
7 015614 170502          DHMLSR: 170502
8 015616 177560          TKCSR: 177560
9 015620 177562          TKDBR: 177562
10 015622 177564         TPCSR: 177564
11 015624 177566         TPDBR: 177566
12 015626 177570         SWR: 177570
13 015630 177570         DISPLAY:177570
14
15
16
17 015632 000000         ERRFLG: 0
18 015634 000000         TRACON: 0
19 015636 000000         PASCNT: 0
20 015640 000000         TSTNO: 0
21 015642 000000         RETURN: 0
22 015644 000000         ICOUNT: 0
23 015646 000000         SAVRO: 0
24 015650 000000         SAVR1: 0
25 015652 000000         SAVR2: 0
26 015654 000000         SAVR3: 0
27 015656 000000         SAVR4: 0
28 015660 000000         SAVR5: 0
29 015662 000000         SAVSP: 0
30 015664 000000         SAVPC: 0
31 015666 000000         WRDCNT: 0
32 015670 000000         CHRCNT: 0
33 015672 017542         STPNT: TSTT80
34 015674 000000         TSTMAX: 0
35 015676 000000         LINFLG: 0
36 015700 000000         LINE: 0
37 015702 000000         LINORG: 0
38 015704 000000         LINANS: 0
39 015706 000000         ANSFLG: 0
40 015710 000000         ORGFLG: 0
41 015712 000000         TIME1: 0
42 015714 000000         TIME2: 0
43 015716 000000         TIFLG: 0
44 015720 177777         LINSSEL: 177777
45 015722 000000         SELMSK: 0
46 015724 000000         SLMSK: 0
47 015726 000002         FILL: 2
48 015730 000000         FILLA: 0
49 015732 000000         RNGRET: 0
50
51
52 015734 124 111 115 .NLIST BEX
53 015774 114 116 040 MNOINT: .ASCII ;TIME OUT WAITING FOR INTERRUPT";
54 016013 123 124 101 MSTATE: .ASCII ;LN CSR LSR;
55 016044 114 111 116 MLINER: .ASCII ;STATUS ERROR"EXP REC;
56 016101 114 111 116 MLINE1: .ASCII ;LINE ERROR"EXP REC LINE SEL;
57 016132 124 122 101 MTRANE: .ASCII ;TRANSITION ERROR"EXP REC LINE;

```

```

;FILL CHAR/COUNT
;TEMP STORAGE FOR FILL COUNT

```

| | | | | | | |
|-----|--------|--------|-----|-----|---------------------|---|
| 58 | 016171 | 045 | 042 | 045 | DIALM: .ASCII | ;"DIAL ANSWERING DATA SET"0; |
| 59 | 016227 | 045 | 042 | 045 | MT103T: .ASCII | ;"103A MODEM CONNECT-DISCONNECT TEST"0; |
| 60 | 016300 | 045 | 042 | 045 | MT202T: .ASCII | ;"202C MODEM CONNECT-DISCONNECT TEST"0; |
| 61 | 016351 | 045 | 042 | 045 | MSELOR: .ASCII | ;"ORIGINATE LINE-0; |
| 62 | 016375 | 045 | 042 | 101 | MSELAN: .ASCII | ;"ANSWER LINE 0; |
| 63 | 016414 | 045 | 042 | 061 | MT103A: .ASCII | ;"103A TEST COMPLETE"0; |
| 64 | 016443 | 045 | 042 | 062 | MT202A: .ASCII | ;"202C TEST COMPLETE"0; |
| 65 | 016472 | 045 | 042 | 123 | MDISC: .ASCII | ;"STRIKE ANY TTY KEY TO TEST DISCONNECT0; |
| 66 | 016542 | 045 | 042 | 045 | M16: .ASCII | ;"16 LINE SCANNER TEST"0; |
| 67 | 016575 | 045 | 042 | 045 | MTITLE: .ASCII | ;"CZDHK-F - -MODEM CONTROL DIAGNOSTIC - - "0; |
| 68 | 016661 | 045 | 042 | 126 | MVECTO: .ASCII | ;"VECTOR ADDRESS-0; |
| 69 | 016703 | 045 | 042 | 103 | MREGAD: .ASCII | ;"CONTROL REGISTER ADDRESS-0; |
| 70 | 016737 | 045 | 042 | 114 | MLINSL: .ASCII | ;"LINE SELECT PARAMETER -0; |
| 71 | 016771 | 045 | 042 | 124 | MTEST: .ASCII | ;"TEST 0; |
| 72 | 017001 | 040 | 040 | 077 | MQM: .ASCII | ; ?0; |
| 73 | 017005 | 045 | 042 | 100 | MCRLF: .ASCII | ;"0; |
| 74 | 017010 | 045 | 042 | 123 | MLINE: .ASCII | ;"SINGLE LINE CABLE TEST"0; |
| 75 | 017043 | 045 | 042 | 114 | MLINEI: .ASCII | ;"LINE NUMBER-0; |
| 76 | 017062 | 106 | 101 | 124 | MFATAL: .ASCII | ;FATAL ERROR"CSTAT LSTAT0; |
| 77 | 017114 | 045 | 042 | 124 | MTRNDE: .ASCII | ;"TRANSITION DETECTED"CSTAT LSTAT0; |
| 78 | 017160 | 045 | 042 | 120 | MPFAIL: .ASCII | ;"POWER FAILURE0; |
| 79 | 017200 | 055 | 103 | 125 | MPF1: .ASCII | ;-CURRENT TEST WILL RESTART"0; |
| 80 | 017235 | 136 | 103 | 100 | MCONTC: .ASCII | ;+C0; |
| 81 | 017240 | 136 | 126 | 100 | MCONTV: .ASCII | ;+V0; |
| 82 | 017243 | 136 | 114 | 100 | MCONTL: .ASCII | ;+L0; |
| 83 | 017246 | 045 | 042 | 123 | \$SWREQ: .ASCII | ;"SWR= 0; |
| 84 | 017257 | 040 | 040 | 040 | \$NEWIS: .ASCII | ; NEW= 0; |
| 85 | 017271 | 045 | 042 | 105 | MEPASS: .ASCII | ;"END PASS 0; |
| 86 | 017307 | 045 | 042 | | MBCD: .ASCII | ;"0; |
| 87 | | 017411 | | | .+.100 | |
| 88 | | | | | .EVEN | |
| 89 | 017412 | 000000 | | | TEMTAB: 0 | |
| 90 | | 017424 | | | .+.10 | |
| 91 | | | | | | |
| 92 | 017424 | 000000 | | | 0 | |
| 93 | | | | | | |
| 94 | | | | | | |
| 95 | | | | | ;EMT DISPATCH TABLE | |
| 96 | 017426 | 013530 | | | EMTTAB: ERRCS | |
| 97 | 017430 | 013546 | | | ERRLS | |
| 98 | 017432 | 013110 | | | LOOP | |
| 99 | 017434 | 013342 | | | FREEZE | |
| 100 | 017436 | 014532 | | | TYPER | |
| 101 | 017440 | 014440 | | | SV05P | |
| 102 | 017442 | 014076 | | | OCTASN | |
| 103 | 017444 | 014500 | | | RS05 | |
| 104 | 017446 | 014270 | | | BINASC | |
| 105 | 017450 | 014336 | | | DIVI | |
| 106 | 017452 | 013424 | | | ERR | |
| 107 | 017454 | 014674 | | | INSTR | |
| 108 | 017456 | 013442 | | | ERRT | |
| 109 | 017460 | 013464 | | | ERRS | |
| 110 | 017462 | 013506 | | | ERRN | |
| 111 | 017464 | 012026 | | | GETLIN | |
| 112 | 017466 | 012062 | | | SETUPS | |
| 113 | 017470 | 012316 | | | CKRNG | |
| 114 | 017472 | 012414 | | | WAITR | |

| | | | | |
|-----|--------|--------|----------------|-----------------------|
| 115 | 017474 | 012502 | CKTRN | |
| 116 | 017476 | 012452 | WAITRR | |
| 117 | 017500 | 014164 | CNTLU | :CALL BY EMT CNTLUU |
| 118 | 017502 | 013076 | CKINT | :CALL BY EMT CKINTT |
| 119 | 017504 | 002020 | KBDINT | :CALLBY EMT KBDIN |
| 120 | 017506 | 013724 | ERRQ | :CALLED BY EMT ERRINT |
| 121 | 017510 | 000000 | EMTLIM: 0 | |
| 122 | 017512 | 017542 | TSTLST: TSTB0 | |
| 123 | 017514 | 017724 | TSTB1 | |
| 124 | 017516 | 017766 | TSTB2 | |
| 125 | 017520 | 017774 | TSTB3 | |
| 126 | 017522 | 000000 | 0 | |
| 127 | 017524 | 000000 | 0 | |
| 128 | 017526 | 000000 | 0 | |
| 129 | 017530 | 000000 | 0 | |
| 130 | 017532 | 000033 | GRO: NO-1 | |
| 131 | 017534 | 000007 | N1-100-1 | |
| 132 | 017536 | 000001 | N2-200-1 | |
| 133 | 017540 | 000000 | N3-300 1 | |
| 134 | 017542 | 002250 | TSTB0: T0 | |
| 135 | 017544 | 000001 | 1 | |
| 136 | | | .MACRO COMMENT | |
| 137 | | | .NLIST | |
| 138 | | | MO=1 | |
| 139 | | | XM=MO | |
| 140 | | | .LIST | |
| 141 | | | .ENDM | |
| 142 | 017546 | | COMMENT | |
| | | 000001 | MO=1 | |
| | | 000001 | XM=MO | |
| 143 | | 000033 | .REPT NO-1 | |
| 144 | | | TM \MO,0 | |
| 145 | | | .ENDR | |
| | 017546 | | TM \MO,0 | |
| | | 004000 | TIMES=4000 | |
| | 017546 | 002276 | T1 | |
| | 017550 | 004000 | TIMES | |
| | | 000002 | MO=MO+1 | |
| | 017552 | | TM \MO,0 | |
| | | 004000 | TIMES=4000 | |
| | 017552 | 002340 | T2 | |
| | 017554 | 004000 | TIMES | |
| | | 000003 | MO=MO+1 | |
| | 017556 | | TM \MO,0 | |
| | | 004000 | TIMES=4000 | |
| | 017556 | 002402 | T3 | |
| | 017560 | 004000 | TIMES | |
| | | 000004 | MO=MO+1 | |
| | 017562 | | TM \MO,0 | |
| | | 004000 | TIMES=4000 | |
| | 017562 | 002444 | T4 | |
| | 017564 | 004000 | TIMES | |
| | | 000005 | MO=MO+1 | |
| | 017566 | | TM \MO,0 | |
| | | 004000 | TIMES=4000 | |
| | 017566 | 002506 | T5 | |
| | 017570 | 004000 | TIMES | |

| | | | |
|--------|--------|------------|-------|
| 017572 | 000006 | MO=MO+1 | |
| | 004000 | TM | \MO,0 |
| 017572 | 002550 | TIMES=4000 | |
| 017574 | 004000 | T6 | |
| | 000097 | TIMES | |
| 017576 | 004000 | MO=MO+1 | |
| | 004000 | TM | \MO,0 |
| 017576 | 002624 | TIMES=4000 | |
| 017609 | 004000 | T7 | |
| | 000010 | TIMES | |
| 017602 | 004000 | MO=MO+1 | |
| | 002700 | TM | \MO,0 |
| 017602 | 004000 | TIMES=4000 | |
| 017604 | 004000 | T10 | |
| | 000011 | TIMES | |
| 017606 | 004000 | MO=MO+1 | |
| | 004000 | TM | \MO,0 |
| 017606 | 002770 | TIMES=4000 | |
| 017610 | 004000 | T11 | |
| | 000012 | TIMES | |
| 017612 | 004000 | MO=MO+1 | |
| | 003060 | TM | \MO,0 |
| 017612 | 004000 | TIMES=4000 | |
| 017614 | 004000 | T12 | |
| | 000013 | TIMES | |
| 017616 | 004000 | MO=MO+1 | |
| | 004000 | TM | \MO,0 |
| 017616 | 003150 | TIMES=4000 | |
| 017620 | 004000 | T13 | |
| | 000014 | TIMES | |
| 017622 | 004000 | MO=MO+1 | |
| | 003240 | TM | \MO,0 |
| 017622 | 004000 | TIMES=4000 | |
| 017624 | 004000 | T14 | |
| | 000015 | TIMES | |
| 017626 | 004000 | MO=MO+1 | |
| | 003330 | TM | \MO,0 |
| 017626 | 004000 | TIMES=4000 | |
| 017630 | 004000 | T15 | |
| | 000016 | TIMES | |
| 017632 | 004000 | MO=MO+1 | |
| | 003416 | TM | \MO,0 |
| 017632 | 004000 | TIMES=4000 | |
| 017634 | 004000 | T16 | |
| | 000017 | TIMES | |
| 017636 | 004000 | MO=MO+1 | |
| | 003504 | TM | \MO,0 |
| 017636 | 004000 | TIMES=4000 | |
| 017640 | 004000 | T17 | |
| | 000020 | TIMES | |
| 017642 | 004000 | MO=MO+1 | |
| | 003572 | TM | \MO,0 |
| 017642 | 004000 | TIMES=4000 | |
| 017644 | 004000 | T20 | |
| | 000021 | TIMES | |
| 017646 | | MO=MO+1 | |
| | | TM | \MO,0 |

| | | | |
|--------|--------|------------|-------|
| | 004000 | TIMES=4000 | |
| 017646 | 003660 | T21 | |
| 017650 | 004000 | TIMES | |
| | 000022 | MO=MO+1 | |
| 017652 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| 017652 | 003754 | T22 | |
| 017654 | 000400 | TIMES | |
| | 000023 | MO=MO+1 | |
| 017656 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| 017656 | 004070 | T23 | |
| 017660 | 000400 | TIMES | |
| | 000024 | MO=MO+1 | |
| 017662 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| 017662 | 004260 | T24 | |
| 017664 | 000400 | TIMES | |
| | 000025 | MO=MO+1 | |
| 017666 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| | 000200 | TIMES=200 | |
| 017666 | 004432 | T25 | |
| 017670 | 000200 | TIMES | |
| | 000026 | MO=MO+1 | |
| 017672 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| | 000200 | TIMES=200 | |
| 017672 | 004574 | T26 | |
| 017674 | 000200 | TIMES | |
| | 000027 | MO=MO+1 | |
| 017676 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| | 000200 | TIMES=200 | |
| 017676 | 005034 | T27 | |
| 017700 | 000200 | TIMES | |
| | 000030 | MO=MO+1 | |
| 017702 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| | 000200 | TIMES=200 | |
| 017702 | 005270 | T30 | |
| 017704 | 000200 | TIMES | |
| | 000031 | MO=MO+1 | |
| 017706 | | TM | \MO,0 |
| | 004000 | TIMES=4000 | |
| | 000400 | TIMES=400 | |
| | 000200 | TIMES=200 | |
| 017706 | 005524 | T31 | |
| 017710 | 000200 | TIMES | |
| | 000032 | MO=MO+1 | |

| | | | | |
|------------|--------|--|----------------|-------|
| 017712 | | | TM | \M0.0 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017712 | 005760 | | T32 | |
| 017714 | 000200 | | TIMES | |
| | 000033 | | MO=M0+1 | |
| 017716 | | | TM | \M0.0 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017716 | 006144 | | T33 | |
| 017720 | 000200 | | TIMES | |
| | 000034 | | MO=M0+1 | |
| 146 017722 | 000000 | | 0 | |
| 147 017724 | 006454 | | TSTTB1: T100 | |
| 148 017726 | 000001 | | 1 | |
| 149 | | | .MACRO COMMENT | |
| 150 | | | .NLIST | |
| 151 | | | M1=101 | |
| 152 | | | XM=M1 | |
| 153 | | | .LIST | |
| 154 | | | .ENDM | |
| 155 017730 | | | COMMENT | |
| | 000101 | | M1=101 | |
| | 000101 | | XM=M1 | |
| 156 | 000007 | | .REPT N1-101 | |
| 157 | | | TM | \M1.1 |
| 158 | | | .ENDR | |
| 017730 | | | TM | \M1.1 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017730 | 006512 | | T101 | |
| 017732 | 000200 | | TIMES | |
| | 000102 | | M1=M1+1 | |
| 017734 | | | TM | \M1.1 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017734 | 006672 | | T102 | |
| 017736 | 000200 | | TIMES | |
| | 000103 | | M1=M1+1 | |
| 017740 | | | TM | \M1.1 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017740 | 007046 | | T103 | |
| 017742 | 000200 | | TIMES | |
| | 000104 | | M1=M1+1 | |
| 017744 | | | TM | \M1.1 |
| | 004000 | | TIMES=4000 | |
| | 000400 | | TIMES=400 | |
| | 000200 | | TIMES=200 | |
| 017744 | 007222 | | T104 | |
| 017746 | 000200 | | TIMES | |
| | 000105 | | M1=M1+1 | |

```
017750      004000      TM      \M1.1
              000400      TIMES=4000
              000200      TIMES=400
017750      007376      T105
017752      000200      TIMES
              000106      M1=M1+1
017754      004000      TM      \M1.1
              000400      TIMES=4000
              000200      TIMES=400
              000200      TIMES=200
017754      007554      T106
017756      000200      TIMES
              000107      M1=M1+1
017760      004000      TM      \M1.1
              000400      TIMES=4000
              000200      TIMES=400
              000200      TIMES=200
017760      007732      T107
017762      000200      TIMES
              000110      M1=M1+1
159 017764      000000      0
160 017766      010110      TSTTB2: T200
161 017770      000001      1
162 017772      000000      0
163 017774      010464      TSTTB3: T300
164 017776      000001      1
165      000001      .END
```

| | | | | | | | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| AND | - | ***** | DIGIT | - | 014434 | FILLA | 015730 | INT7B | 003326 | MLINER | 016044 | |
| ANSFLG | 015706 | | DISPLA | 015630 | | FLAG | - | ***** | ITERAT | - | ***** | |
| ANSTR | 011642 | | DISPRE | 000174 | | FOR | - | ***** | KBDIN | - | 104027 | |
| ANSTRR | 011730 | | DIVCNT | 014412 | | FREEZE | 013342 | | KBDINT | 002020 | MLINSL | 016737 |
| ANSTR1 | 011656 | | DIVI | 014336 | | FREEZX | 013376 | | KBDIN1 | 002102 | MNOINT | 015734 |
| ANSTR2 | 011672 | | DIVIA | 014362 | | GETLIN | 012026 | | KBDIN2 | 002134 | MPFAIL | 017160 |
| ANSTR3 | 011706 | | DIVIB | 014430 | | GETLNS | - | 104017 | KBDIN3 | 002176 | MPF1 | 017200 |
| ANSTR4 | 011722 | | DIVIC | 014400 | | GRO | 017532 | | KRET | 002244 | MQM | 017001 |
| ASCII | - | ***** | DIVIDH | 014434 | | HILIM | 015156 | | LINANS | 015704 | MREGAD | 016703 |
| BINASA | 014300 | | DIVIDL | 014432 | | ICOUNT | 015644 | | LINE | 015700 | MSELAN | 016375 |
| BINASB | 014320 | | DIVIS | 014436 | | INBUF | 015162 | | LINENA | - | 000001 | |
| BINASC | 014270 | | DIVIU | 014342 | | INIT1 | 002254 | | LINFLG | 015676 | MSG | 014722 |
| BINWRD | - | 014432 | DMYRTI | 002242 | | INPUT | - | ***** | LINORG | 015702 | MSTATE | 016013 |
| BUSY | - | 000020 | DONE | - | 000200 | INSTBB | 014742 | | LINSEL | 015720 | MTEST | 016771 |
| CHANGE | - | ***** | EMTDEF | - | ***** | INSTER | 015034 | | LINT1 | 003660 | MTITLE | 016575 |
| CHCK | 015144 | | EMTLIM | 017510 | | INSTR | 014674 | | LINT1A | 003706 | MTRANE | 016132 |
| CHECK | - | ***** | EMTOK | 013056 | | INSTRB | 014734 | | LINT1B | 003734 | MTRNDE | 017114 |
| CHRCNT | 015670 | | EMTSRV | 013044 | | INSTRC | 015022 | | LINT2 | 003754 | MT103A | 016414 |
| CKINT | 013076 | | EMTAB | 017426 | | INSTRD | 015062 | | LINT2A | 004014 | MT103T | 016227 |
| CKINTT | - | 104026 | ENTRY | - | ***** | INSTRG | - | 104013 | LINT2B | 004042 | MT202A | 016443 |
| CKRING | - | 104021 | EOP | 012764 | | INSTR1 | 014720 | | LOGICA | 013026 | MT202T | 016300 |
| CKRNG | 012316 | | ERR | 013424 | | INSTR2 | 015042 | | LOLIM | 015154 | MUX1 | 004574 |
| CKRNG1 | 012362 | | ERRCS | 013530 | | INSTR3 | 015122 | | LOOP | 013110 | MUX1A | 004622 |
| CKRNG2 | 012406 | | ERRFLG | 015632 | | INTENA | - | 000100 | LOOPER | 013246 | MUX1B | 004670 |
| CKTRAN | - | 104023 | ERRGEN | 013564 | | INTERR | - | ***** | LOOPL | 013332 | MUX1C | 004726 |
| CKTRN | 012502 | | ERRINT | - | 104030 | INT1 | 002550 | | LOOPS | 013220 | MUX1D | 004744 |
| CKTRN1 | 012616 | | ERRLS | 013546 | | INT1A | 002616 | | LOOPX | 013262 | MUX1E | 004762 |
| CKTRN2 | 012650 | | ERRMSG | 013644 | | INT1B | 002622 | | LVL | - | 000004 | |
| CKTRN3 | 012710 | | ERRN | 013506 | | INT10 | 003330 | | MAINT | - | 001000 | |
| CKTRN4 | 012734 | | ERROR | - | 104012 | INT10A | 003412 | | MBCD | 017307 | MUX11A | 006512 |
| CLRMUX | - | 002000 | ERRORC | - | 104000 | INT10B | 003414 | | MCONTC | 017235 | MUX11B | 006530 |
| CLRSCN | - | 004000 | ERRORL | - | 104001 | INT108 | 003414 | | MCONTL | 017243 | MUX11C | 006560 |
| CNTLU | 014164 | | ERRORN | - | 104016 | INT11 | 003416 | | MCONTV | 017240 | MUX11D | 006606 |
| CNTLUU | - | 104025 | ERRORS | - | 104015 | INT11A | 003500 | | MCRLF | 017005 | MUX11E | 006624 |
| CO | - | 000100 | ERRORT | - | 104014 | INT11B | 003502 | | MDISC | 016472 | MUX11F | 006636 |
| COF | - | 040000 | ERRQ | 013724 | | INT12 | 003504 | | MENT1 | 004070 | MUX12 | 006670 |
| CONTR | - | ***** | ERRS | 013464 | | INT12A | 003566 | | MENT1A | 004116 | MUX12A | 006672 |
| CONVER | - | 104010 | ERRT | 013442 | | INT12B | 003570 | | MENT1B | 004144 | MUX12B | 006710 |
| CORRER | - | ***** | ERR1 | 012742 | | INT13 | 003572 | | MENT1C | 004166 | MUX12C | 006740 |
| CS | - | 000040 | ERR2 | 012744 | | INT13A | 003654 | | MENT1D | 004174 | MUX12D | 006766 |
| CSF | - | 020000 | ERR3 | 012746 | | INT13B | 003656 | | MENT1E | 004226 | MUX12E | 007000 |
| CSTR1 | 002276 | | ERR4 | 012750 | | INT2 | 002624 | | MENT1F | 004250 | MUX12F | 007012 |
| CSTR2 | 002340 | | ERTAB | 013656 | | INT2A | 002672 | | MENT2 | 004260 | MUX13 | 007044 |
| CSTR3 | 002402 | | ERTAB0 | 013742 | | INT2B | 002676 | | MENT2A | 004302 | MUX13A | 007064 |
| CSTR4 | 002444 | | ERTAB1 | 013750 | | INT3 | 002700 | | MENT2B | 004362 | MUX13B | 007114 |
| CSTR5 | 002506 | | ERTAB2 | 013766 | | INT3A | 002764 | | MENT2C | 004404 | MUX13C | 007142 |
| DATA | - | ***** | ERTAB3 | 014004 | | INT3B | 002766 | | MENT2D | 004416 | MUX13D | 007154 |
| DATA1 | 012752 | | ERTAB4 | 014016 | | INT4 | 002770 | | MENT3 | 004432 | MUX13E | 007166 |
| DATA2 | 012754 | | ERTAB5 | 014030 | | INT4A | 003052 | | MENT3A | 004454 | MUX13F | 007220 |
| DATA3 | 012756 | | ERTAB6 | 014060 | | INT4B | 003056 | | MENT3B | 004466 | MUX14 | 007222 |
| DATA4 | 012760 | | EXTRAC | - | 104011 | INT5 | 003060 | | MENT3C | 004524 | MUX14A | 007240 |
| DELAY | - | ***** | FAILIN | - | ***** | INT5A | 003142 | | MENT3D | 004546 | MUX14B | 007270 |
| DHMCSR | 015612 | | FAKE | - | ***** | INT5B | 003146 | | MENT3E | 004560 | MUX14C | 007316 |
| DHMLSR | 015614 | | FATAL | - | ***** | INT6 | 003150 | | MEPASS | 017271 | MUX14D | 007330 |
| DHMLVL | 015610 | | FATEX | 011732 | | INT6A | 003232 | | MFATAL | 017062 | MUX14E | 007342 |
| DHMVEC | 015606 | | FATRET | 011744 | | INT6B | 003236 | | MLINE | 017010 | MUX14F | 007374 |
| DIALP | 016171 | | FILL | 015726 | | INT7 | 003240 | | MLINEI | 017043 | MUX15 | 007376 |
| | | | | | | INT7A | 003322 | | | | MUX15A | 007414 |

| | | | | | | |
|----------------|----------|-----------------|------------------|-----------------|--------|--------|
| MUX15B | 007442 | OCTAL = ***** | SCOPEF = 104003 | TIME = ***** | T103E1 | 010426 |
| MUX15C | 007470 | OCTASC = 104006 | SECRX = 000020 | TIMES = 000200 | T103E2 | 010432 |
| MUX15D | 007502 | OCTASN = 014076 | SECRXF = 010000 | TIME1 = 015712 | T103E3 | 010436 |
| MUX15E | 007516 | OCTAS1 = 014124 | SECTX = 000010 | TIME2 = 015714 | T103E4 | 010442 |
| MUX15F | 007552 | OF = ***** | SELMASK = 015722 | TIPFLG = 002016 | T104 | 007222 |
| MUX16 | 007554 | ON = ***** | SERVIC = ***** | TKCSR = 015616 | T105 | 007376 |
| MUX16A | 007572 | ONLY = ***** | SET = ***** | TKDBR = 015620 | T106 | 007554 |
| MUX16B | 007620 | ORGFLG = 015710 | SETUP = 104020 | TMP1 = 014262 | T107 | 007732 |
| MUX16C | 007646 | ORGTR = 011550 | SETUPB = 012300 | TMP2 = 014264 | T11 | 002770 |
| MUX16D | 007660 | ORGTRR = 011636 | SETUP8 = 012062 | TO = ***** | T12 | 003060 |
| MUX16E | 007674 | ORGTR1 = 011564 | SETUPS = 012062 | TPCSR = 015622 | T13 | 003150 |
| MUX16F | 007730 | ORGTR2 = 011600 | SETUP1 = 012114 | TPDBR = 015624 | T14 | 003240 |
| MUX17 | 007732 | ORGTR3 = 011614 | SETUP2 = 012130 | TRACON = 015634 | T15 | 003330 |
| MUX17A | 007750 | ORGTR4 = 011614 | SETUP4 = 012242 | TRANEX = 011746 | T16 | 003416 |
| MUX17B | 007776 | OUT = ***** | SINGLE = 000001 | TRANS = 011506 | T17 | 003504 |
| MUX17C | 010024 | OUTPUT = ***** | SINTFL = 002246 | TRANSI = ***** | T2 | 002340 |
| MUX17D | 010036 | PASCNT = 015636 | SLMSK = 015724 | TRANX1 = 011756 | T20 | 003572 |
| MUX17E | 010052 | PFAIL = 015204 | SMLN = 014266 | TRMRDY = 000002 | T200 | 010110 |
| MUX17F | 010106 | POINT = ***** | ST = 000200 | TRNTAB = 012014 | T201 | 010446 |
| MUX2 | 005034 | POPRO = 012600 | STACK = 001100 | TRNTYP = 011766 | T202A | 010530 |
| MUX2A | 005062 | POP1SP = 005726 | START = 001100 | TSTENT = 013266 | T202A1 | 010536 |
| MUX2B | 005130 | POP2SP = 022626 | STARTN = 001634 | TSTGO = 001746 | T202B | 010542 |
| MUX2C | 005166 | PS = 177776 | STARTO = 001322 | TSTLST = 017512 | T202B1 | 010552 |
| MUX2D | 005200 | PSW = 177776 | START1 = 001364 | TSTMAX = 015674 | T202B2 | 010556 |
| MUX2E | 005216 | PUSHRO = 010046 | STATUS = ***** | TSTNO = 015640 | T202C | 010562 |
| MUX2F | 005250 | PUSH15 = 005746 | STEP = 000400 | TSTPNT = 015672 | T202D | 010576 |
| MUX3 | 005270 | PUSH25 = 024646 | STEPR = 013404 | TSTTBO = 017542 | T202D1 | 010642 |
| MUX3A | 005316 | RADIX = 014436 | STORE = 015160 | TSTTB1 = 017724 | T202D2 | 010646 |
| MUX3B | 005364 | REGST1 = 001606 | STRING = ***** | TSTTB2 = 017766 | T202D3 | 010652 |
| MUX3C | 005422 | RESTAR = 015242 | STRLIN = 006454 | TSTTB3 = 017774 | T202D4 | 010656 |
| MUX3D | 005434 | RESTA1 = 015310 | STRLNA = 006474 | TYFILL = 014642 | T202E | 010662 |
| MUX3E | 005452 | RESTOR = ***** | STRTOA = 001726 | TYPE = 104004 | T202E1 | 010726 |
| MUX3F | 005504 | RES05 = 104007 | STYLE = ***** | TYPECL = 014630 | T202E2 | 010732 |
| MUX4 | 005524 | RETURN = 015642 | ST103A = 010110 | TYPER = 014532 | T202E3 | 010736 |
| MUX4A | 005552 | RING = 000200 | ST103B = 010152 | TYPERA = 014542 | T202E4 | 010742 |
| MUX4B | 005620 | RINGF = 100000 | ST202A = 010464 | TYPER1 = 014606 | T202F | 010746 |
| MUX4C | 005656 | RNGRET = 015732 | ST202B = 010526 | TYPER2 = 014622 | T202F2 | 011026 |
| MUX4D | 005670 | ROUTIN = ***** | SUSWR = 001162 | TO = 002250 | T202F3 | 011032 |
| MUX4E | 005706 | RS = 000004 | SV05 = 014446 | T1 = 002276 | T202F4 | 011036 |
| MUX4F | 005740 | RS05 = 014500 | SV05P = 014440 | T10 | 002700 | 011042 |
| MUXB | 005760 | SAVE = ***** | SWR = 015626 | T100 | 006454 | 011046 |
| MUXBA | 005776 | SAVPC = 015664 | SWREG = 000176 | T101 | 006512 | 011112 |
| MUXBB | 006030 | SAVRO = 015646 | SWRTB = 014052 | T102 | 006672 | 011116 |
| MUXBC | 006036 | SAVR1 = 015650 | SW06 = 000100 | T103 | 007046 | 011122 |
| MUXBD | 006072 | SAVR2 = 015652 | SW08 = 000400 | T103A | 010154 | 011126 |
| MUXBE | 006124 | SAVR3 = 015654 | SW09 = 001000 | T103A1 | 010162 | 011132 |
| MVECTO | 016661 | SAVR4 = 015656 | SW10 = 002000 | T103B | 010166 | 011136 |
| M0 | = 000034 | SAVR5 = 015660 | SW11 = 004000 | T103B1 | 010176 | 011176 |
| M1 | = 000110 | SAVR6 = 015662 | SW12 = 010000 | T103B2 | 010202 | 011202 |
| M16 | 016542 | SAVSP = 104005 | SW13 = 020000 | T103C | 010206 | 011206 |
| N | = 000300 | SCNENA = 000040 | SW14 = 040000 | T103D | 010252 | 011212 |
| NUMBER = ***** | | SCNT1 = 006144 | SW15 = 100000 | T103D1 | 010256 | 011216 |
| NXTTS | 012762 | SCNT1A = 006204 | T = 000014 | T103D2 | 010256 | 011276 |
| N0 | = 000034 | SCNT1B = 006260 | TELETY = ***** | T103D3 | 010262 | 011302 |
| N1 | = 000110 | SCNT1C = 006404 | TEMTAB = 017412 | T103D4 | 010266 | 011306 |
| N2 | = 000202 | SCNT1D = 006424 | TEST = ***** | T103E | 010272 | 011312 |
| N3 | = 000301 | SCOPE = 104002 | TESTS = ***** | T103EN | 010446 | 011316 |
| | | | TIFLG = 015716 | T103ES | 010332 | 011474 |

| | | | | | | | | | | | | |
|--------|--------|------|--------|--------|--------|--------|--------|--------|--------|---------|--------|--------|
| T202J5 | 011356 | T26 | 004574 | T6 | 002550 | WAITS | = | 104024 | XSCRX | = | 000001 | |
| T202J1 | 011454 | T27 | 005034 | T7 | 002624 | WRDCNT | | 015666 | X1A | | 001652 | |
| T202J2 | 011460 | T3 | 02402 | UP | = | ***** | X | = | 000000 | X1B | 001716 | |
| T202J3 | 011464 | T30 | 005270 | VECSTA | | 001434 | XCO | = | 000004 | Y | = | 000000 |
| T202J4 | 011470 | T300 | 010464 | VECSTR | | 001414 | XCS | = | 000002 | \$NEWIS | | 017257 |
| T21 | 003660 | T31 | 005524 | VECST1 | | 001540 | XFLAG | | 001312 | \$SWREQ | | 017246 |
| T22 | 003754 | T32 | 005760 | WAITR | | 012414 | XM | = | 000101 | .1 | | 013646 |
| T23 | 004070 | T33 | 006144 | WAITRN | = | 104022 | XN | = | 000300 | .2 | | 013660 |
| T24 | 004260 | T4 | 002444 | WAITRR | | 012452 | XOR | | 015326 | .3 | | 013662 |
| T25 | 004432 | T5 | 002506 | WAITR1 | | 012464 | XORSVC | | 001314 | .4 | | 013706 |

. ABS. 020000 000
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 19456 WORDS (76 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 71 PAGES
CZDHKF.BIN,CZDHKF.SEG=DHMACA.MAC,CZDHKF.P11