LA36

TERMINAL (DL11 & KL11)

MD-11-DZLAC-C

EP-DZLAC-C-DL-A

OCT 1976

digital

Made In U.S.A.

PRODUCT CODE: MAINDEC-11)DZLAC-C-D

PRODUCT NAME: LA36 TERMINAL (DL11 & KL11 INTERFACE)

DATE CREATED: SEPTEMBER 1975

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: ROBERT BAKER

TABLE OF CONTENTS

1.0 ABSTRACT

THIS DIAGNOSTIC IS DIVIDED INTO FOUR BASIC SECTIONS:

1. A CHECK OF THE CONSOLE TERMINAL INTERFACE LOGIC

2. A CHECK OF THE PRINTING CHARACTERISTICS AND CONTROL LOGIC

3. AJ ECHO PORTION DESIGNED TO CHECK THE KEYBOARD AND TO AID IN THE DIAGNOSIS OF TERMINAL PROBLEMS.

4. A CHECK OF THE VARIOUS LA36 OPTIONS.

PATTERNS USED BY THE PRINTING TESTS WERE CHOSEN FOR EASE OF VISUAL VERIFICATION. THE ECHO TESTS WERE DESIGNED FOR MAXIMUM FLEXIBILITY, WITH TEST 24 ALLOWING ANY DESIRED PATTERN TO BE USED.

2.0 REQUIREMENTS

2.1 EQUIPMENT AND ASSIGNMENTS

THE DIAGNOSTIC IS WRITTEN TK RUN ON ALL MODELS OF THE PDP11 COMPUTER SITH EITHER A KL11 OR DL11 CONSOLE TERMINAL INTERFACE. IN ADDITION, THE DIAGNOSTIC IS SETUP SUCH THAT UP TO 16 ADDITIONAL DL11'S ASSIGNED BETWEEN ADDRESSES 776500-776676, CAN BE TESTED. TO DRIVE ANY DL11'S BETWEEN AJDRESSES 775610 AND 775600, DEPOSIT 175610 IN DLADR BEFORE STARTING THE DIAGNOSTIC. TO DRIVE ANY DL11'S BETWEEN ADDRESSES 776000 AND 776176(DEPOSIT 176000 IN DLADR BEFORE STARTING THE DIAGNOSTIC. TO TEST A DL11 AT ANY NON-SPANDARD AJDRESS OR TO RE-DEFINE ANY TARMINAL AS THE CONSOLE TERMINAL, SET THE ADDRESS OF THE RECEIVER STATUS REGISTER IN CONADD AND THE RECEIVER INTERRUPT VECTOR IN CONVEC BEFORE STARTING THE DIAGNOSTIC. IF A SLOWER DEVICE IS USED AS THE CONSOLE TERMINAL, ALL LA36'S WILL APPEAR TO BE RUNNING AT THE SHOWER BAUD RATE. TK OVERCOME THIS, DEPOSIT THE ADDRESS OF THE RECEIVER STATUS REGISTER OF ANY AVAILABLE DL11 IN CONADD AND THE RECEIVER INTERRUPT VECTOR OF THE SAME DL11 IN CONVEC. (REFER TO THE SYMBOL DEFINITIONS IN THE LISTING FOR THE ABOVE MANTIONED LOCATIONS.)

2.2 STORAGE

THE DIAGNOSTIC PROGRAM USES ALL OF 4K OF MEMORY WITH EXCEPTION OF THE AREA USED BY THE ABSOLUTE LOADER.

2.3 PRELIMINARY PROGRAMS

ANY APPLICABLE PDP-11 DIAGNOSTICS SHOULD BE RUN ON THE PROCESSOR. IF ANY ERRORS ARE ENCOUNTERED DURING THE INTERFACE CHECK, REFER TK THE APPROPRIATE IJTERFACE DIAGNOSTIC FOR BURTHER HELP IN LOCATING THE PROBLEM IF NEEDED.

3.0 LOADING PROCEDURE AND INITIALIZATION

HOAD THE LA36 DIAGNOSTIC PROGRAM TAPE FOLLOWING NORMAL PROCEDURES. BEFORE STARTING THE PROGRAM, REFER TK THE DESCRIPTION OF THE ROUTINE "DLY". TIME DELAYS USED BY THE PROGRAM ARE A FUNCTION OF THE CPU MODEL AND MEMORY TYPE AND SHOULD BE SET-UP BEFORE RUNNING THE DIAGNOSTIC. THE ROUTINE IS PRESET FOR A PDP-11/05 WITH CORE MEMORY. REFER TO SECTION 2.1 FOR NON-SPANDARD TERMINAL ADDRESSES AND FOR TESTING MULTIPLE DL11 INTERFACES.

IF A HARDWARE SWITCH REGISTER DOES NOT EXIST, THE PROGRAM WILL USE THE CONTENTS OF LOCATION 172 AS THE VALUE OF THE SWITCHES. THEREFORE, BE SURE TO LOAD LOCATION 176 WITH THE SWITCH VALUE BEFORE STARTING THE PROGRAM WHEN NOT USING HARDWARE SWITCHES.

4.0 STARTING PROCEDURE

4.1 STARTING ADDRESSES

```
200(8) = RUN WITH SWITCH REGISTER CONTROL
         - PERFORM CONSOLE TERMINAL I/O TESTS.
204(8) = RUN WITH SWITCH REGISTER CONTROL
         - SKIP CONSOLE TERMINAL I/O TESTS.
210(8) = RUN WITH KAYBOARD CONTROL
         - PERFORM CONSOLE TERMINAL I/O TESTS.
214(8) = RUN WITH KEYBOARD CONTROL
         - SKIP CONSOLE TERMINAL I/O TESTS.
```

4.2 SWITCH REGISTER CONTROL WITH I/O TESTS

A. SET THE SWITCH REGISTER TO 200(8) AND PRESS THE LOAD ADDRESS SWITCH

B. SET THE SWITCH REGISTER BITS 7-0 EQUAL TO THE PAPER WIDTH IN TERMS OF THE NUMBER OF COLUMNS (OCTAL). REFER TO SECTION 5.1.7

C. SET THE SWITCH REGISTER BIT 8 EQUAL TO 1 OR 0 AND PRESS THE START SWITCH. A MESSAGE WILL BE PRINTED INDICATING THE NUMBER OB DL11'S BEING TESTED. RAFER TO SECTION 5.1.5

D. IF BIT 8 WERE ZERO WHEN STARTING, THE PRINTER TESTS ARE EXECUTED SEQUENTIALLY, AFTER THE ENTIRE SARIES OF I/O TESTS ARE EXECUTED.

E. IF BIT 8 WAS SET WHEN THE START SWITCH WAS PRESSED, THE ENTIRE SERIES OF I/O TESTS WILL BE EXECUTED AND THE PROGRAM WILL THEN BE WAITING FOR CONTROL VIA THE SWITCH REGISTER. THE CPU WILL HALT AT LOCATION SELHLT.

4.3 SWITCH REGISTER CONTROL - WITHOUT I/O TESTS

SAME AS SECTION 4.2 EXCEPT IN STEP A, SET THE SWITCH REGISTER TO 204(8).

4.4 KEYBOARD CONTROL - WITH I/O TESTS

A. SET THE SWITCH REGISTER TO 210(8) AND PRESS THE LOAD ADDRESS SWITCH.

B. SET THE SWITCH REGISTER BITS 7-0 EQUAL TO THE PAPER WIDTH IN TERMS OF THE NUMBER OB COLUMNS (OCTAL). REFER TO SECTION 5.1.7

C. SET SWITCH 8 AND PRESS THE START SWITCH. A MESSAGE WILL BE PRINTED INDICATING THE NUMBER OF DL11'S BEING TESTED. REFER TO SECTION 5.1.5

D. IF BIT 8 WAS ZERO WHEN SPARTING, THE PRINTER TESTS ARE EXECUTED SEQUENTIALLY AFTER THE ENTIRE SERIES OF I/O TESTS ARE EXECUTED.

E. IF BIT 8 WERE SET WHEN THE START SWITCH WAS PRESSED, THE ENTIRE SERIES OF I/O TESTS WILL BE EXECUTED FOLLOWED BY THE SELECT TEST MESSAGE. THE PROGRAM WILL THEN BE WAITING FOR A TEST SELECTION VIA ANY TARMINAL KEYBOARD. RAFER TO SECTION 5.2

4.5 KEYBOARD CONTROL - WITHOUT I/O TESTS

SAME AS SECTION 4.4 EXCEPT IJ STEP A, SET THE SWITCH REGISTER TO 214 (8).

5.0 OPERATING PROCEDURE

THE PROGRAM CAN BE CONTROLLED IN EITHER OF TWO METHODS: BY THE CONSOLE SWITCH REGISTER OR FROM THE KEYBOARD OF THE TERMINAL(S) UNDER TEST.

5.1 SWITCH REGISTER CONTROL

THE VARIOUS SWITCHES AND THEIR FUNCTIONS ARE LISTED BELOW. SWITCHES MAY BE CHANGED AND SAT AS DESIRED EXCEPT AS NOTED IN THE SPECIFIC SWITCH DESCRIPTIONS. REFER TO THE DETAILED SWITCH DESCRIPTIONS FOR BURTHER, MORE COMPLETE INFORMATION.

SWITCH NUMBER	DESCRIPTION
15	1(UP) = HALT AT END OF TEST 0(DOWN) = CONTINUE TEST SEQUENCE
14	1(UP) = CONTINUE ON ERROR 0(DOWN) = HALT ON ERROR
13	1(UP) = DRIVE ONLY CONSOLE TERMINAL 0(DOWN) = DRIVE ALL TERMINALS
11	1(UP) = LOOP ON INDIVIDUAL TEST 0(DOWN) = NORMAL TEST SEQUENCE
8	1(UP) = RUN TEST ONCE AND HALT 0(DOWN) = LOOP OJ TEST SEQUENCE
5-0	TEST NUMBER SELECTION
7-0	NUMBER OF COLUMNS AT START-UP

5.1.1 SWITCH 15

WITH SWITCH 15 IN THE UP POSITION, THE PROGRAM WILL HALT AT THE END OB THE CURRENT TEST. REPLACING SWITCH 15 TO THE DOWN POSITION AND PRESSING CONTINUE WILL CONTINUE THE NORMAL TEST OPERATION. DURING THE HALT, ANY OF THE CKNTROL SWITCHES MAY BE CHANGED OR SET AS DESIRED.

5.1.2 SWITCH 14

PLACING SWITCH 14 IN THE UP POSITION WILL CAUSE THE PROGRAM TO CKNTINUE ON ERRORS DURING ANY OF THE I-O TESTS ONLY. WITH SWITCH 14 DOWN, THE PROGRAM WILL HALT (AT ERRHLT) ON ANY ERRORS DURING THE I-O TESTS WITH THE LOCATION OF THE ERROR IN R0. PRESSING CONTINUE WILL CAUSE THE PROGRAM TO CONTINUE IF SWITCH 14 IS DOWN. WITH SWITCH 14 UP, PRESSING CONTINUE WILL CAUSE THE PROGRAM TO LOOP ON THE ERROR.

NOTE

ERRORS CAN OCCUR ONLY DURING THE I/O TESTS. THE TERMINAL IS CONNECTED TO A SERIAL LINE AND THERE IS NO INFORMATION RETURNING TO THE CPU FROM THE TERMINAL. THE PROGRAM CONNOT DETECT ERRORS IN THE TERMINAL. ONLY ERRORS IN THE INTERFACE CAN BE DETECTED.

5.1.3 SWITCH 13

PLACING SWITCH 13 IN THE DOWN POSITION WILL CAUSE THE DRIVING OF ALL MULTIPLE TERMINALS DURING THE PRINTER TESTS ONLY. IF SWITCH 13 IS UP, ONLY THE CONSOLE TERMINAL IS DRIVEN.

** NOTE: SWITCH 13 SHOULD ONLY BE CHANGED WHEN THE PROGRAM IS WAITING FOR A TEST SELECTION.

5.1.4 SWITCH 11

PLACING SWITCH 11 UP AT ANY TIME WILL CAUSE THE PROGRAM TO LOOP OJ THE CURRENT TEST AS LONG AS SWITCH 11 REMAINS UP. REPLACING SWITCH 2 DOWN WILL CAUSE THE PROGRAM TO RESUME NORMAL OPERATION AT THE COMPLETION OF THE TEST.

5.1.5 SWITCH 8

WITH SWITCH 8 IN THE DOWN POSITION THE PROGRAM WILL CKNTINUE TO LOOP THROUGH THE PRESENT TEST SEQUENCE. PLACING SWITCH 8 UP WILL CAUSE THE PROGRAM TO HALT (AT SELHLT) AT THE COMPLETION OF THE CURRENT TEST. AFTER THE HALT SET THE CONTROL SWITCHES AS DESIRED AND SET SWITCHES 5 TO 0 TO THE NEXT DESIRED TEST NUMBER THEN PRESS CONTINUE TO START THE TEST.

WHEN STARTING THE DIAGNOSTIC THE OPERATOR CAN SELECT A SPECIFIC TEST RATHER THAN AUTOMATICALLY STARTING THE PRINTING TEST SEQUENCE BY SETTING SWITCH 8 UP BEFORE STARTING THE DIAGNOSTIC. UPON COMPLETION OF THE I/O TEST SEQUENCE (IF BEING RUN) THE PROGRAM WILL EITHER HALT AT SELHLT WAITING FOR A TEST SELECTION VIA THE SWITCH REGISTER OR PRINT THE SELECT TEST MESSAGE AND WAIT FOR A TEST SELECTION FROM ANY KEYBOARD. REFER TO SECTION 4 FOR FURTHER INFORMATION.

5.1.6 SWITCHES 5 TO 0

SWITCHES 5 TO 0 ARE USED TO SELECT SPECIFIC TESTS WHEN UNDER SWITCH REGISTER CONTROL. TEST NUMBERS ARE ALWAYS IN OCTAL.

5.1.7 SWITCHES 7 TO 0 (AT START-UP ONLY)

AT START-UP ONLY, SWITCHES 7 TO 0 ARE USED TO SET THE DESIRED MAXIMUM NUMBER OF COLUMNS THE DIAGNOSTIC IS TO TEST. IF THE NUMBER SET IS GREATER THAN 132(10) OR LESS THAN 30(10), THE PROGRAM WILL DEFAULT TO 132(10). THE VALUE SET MUST BE IN OCTAL FORM. THUS, FOR NORMAL OPERATION - TESTING THE FULL 132 (10) COLUMNS - LEAVE SWITCHES 7 - 0 DOWN.

5.2 KEYBOARD CONTROL

THE PROGRAM WILL BE UNDER KEYBOARD CONTROL WHENEVER THE DIAGNOSTIC IS STARTED AT LOCATION 210 OR 214. SWITCHES ON THE CONSOLE SWITCH REGISTER WILL HAVE NO EFFECT WHEN UNDER TERMINAL CONTROL EXCEPT FOR SSITCH 15. THE I/O TESTS CANNOT BE SELECTED WHEN UNDER KEYBOARD CONTROL.

TO STOP A TEST AT ANY TIME, TYPE THE "RUBOUT" OR "DELETE" KEY ON ANY KEYBOARD. ANY TERMINAL MAY STOP THE TEST AND SELECT THE NEXT TEST IF SWITCH 13 IS DOWN. WHEN A TEST IS STOPPED BY TYPING A "RUBOUT" OR DELETE, THE TEST WILL TERMINATE AND THE FOLLOWING MESSAGE WILL BE TYPED:

SELECT TEST NUMBER

AT THIS TIME, TYPE THE DESIRED TEST NUMBER FOLLOWED BY ANY ONE OF THE BOLLOWING CONTROL CHARACTERS:

. (PERIOD) = RUN THE SELECTED TEST ONCE AND RETURN FOR ANOTHER TEST SELECTION.

L = LOOP ON THE SELECTED TEST UNTIL A "RUBOUT" IS TYPED.

S = START THE TEST SEQUENCE WITH THE SELECTED TEST. CONTINUE TO LOOP ON THE PRINTING TEST SEQUENCE UNTIL A "RUBOUT" IS TYPED.

THE "L" OR "S" MAY BE EITHER UPPER OR LOWER CASE, BUT THE TEST NUMBER MUST ALWAYS BE A 2 DIGIT OCTAL NUMBER. THE TEST NUMBER AND TERMINATOR ARE ECHOED BY THE PROGRAM, THUS EACH CHARACTER WILL BE PRINTED TWICE IF THE TERMINAL IS IN HALF DUPLEX. FOR ALL ECHO TESTS, THE "L" AND "S" WILL ONLY RUN THE TEST ONCE (THE SAME AS IF TYPING A PERIOD). FOR ALL OPTION TESTS, THE "S" WILL ONLY RUN THE TEST ONCE (THE SAME AS IF TYPING A PERIOD), HOWEVER, TYPING AN "L" WILL CAUSE THE PROGRAM TO LOOP ON THE SELECTED TEST. IF AN ERROR IS DETECTED IN THE TEST SELECTION (ILLEGAL TEST NUMBER OR CONTROL CHARACTER) A QUESTION MARK IS PRINTED AND THE MESSAGE WILL BE REPEATED.

6.0 TEST DESCRIPTIONS

6.1 PRINTING TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE PRINTING MECHANISM AND THE ASSOCIATED CONTROL LOGIC. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED AND, IF THE TEST IS A FUNCTION OF THE NUMBER OF COLUMNS, THE NUMBER OF COLUMNS BEING TESTED WILL BE INDICATED. A DETAILED DESCRIPTION AND SAMPLE PATTERNS FOR EACH PRINTING TEST FOLLOWS:

6.1.1 TEST 0 - DATA PATH TEST

THIS TEST IS USED TO TEST THE DATA LINES TO AND THROUGH THE INTERFACE AND TO THE TERMINAL. AN ALTERNATING BIT PATTERN IS SENT WHICH WILL PRINT ALTERNATING *'S AND U'S IN A CHECKERBOARD PATTERN TO THE MAXIMUM COLUMN WIDTH. THE STARTING CHARACTER FOR EACH LINE IS ALTERNATED AND A TOTAL OF FOUR LINES ARE PRINTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U
U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*
*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U
U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*U*
```

6.1.2 TEST 1 - PRINTABLE CHARACTER TEST

THIS TEST PRODUCES A CHECK OF ALL 94(10) PRINTABLE CHARACTERS. THE CHARACTERS ARE PRINTED IN GROUPS OF THREE WITH THREE GROUPS PER LINE, SEPARATED BY THREE SPACES BETWEEN GROUPS. THE FIRST COLUMN WILL CONTAIN ALL ASCII CODES FROM 040 TO 077. COLUMN TWO WILL CONTAIN ALL ASCII CODES FROM 100 TO 137 - PRIMARILY THE CAPITAL LETTER SET. THE LAST COLUMN WILL CONTAIN ALL ASCII CODES FROM 140 TO 176 - PRIMARILY THE SMALL LETTER SET.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
      @@@   '''
!!!   AAA   AAA
"""   BBB   BBB
###   CCC   CCC
$$$   DDD   DDD
%%%   EEE   EEE
&&&   FFF   FFF
'''   GGG   GGG
(((   HHH   HHH
)))   III   III
***   JJJ   JJJ
+++   KKK   KKK
,,,   LLL   LLL
---   MMM   MMM
...   NNN   NNN
///   OOO   OOO
000   PPP   PPP
111   QQQ   QQQ
222   RRR   RRR
333   SSS   SSS
444   TTT   TTT
555   UUU   UUU
666   VVV   VVV
777   WWW   WWW
888   XXX   XXX
999   YYY   YYY
:::   ZZZ   ZZZ
;;;   [[[
<<<   \\\
===   ]]]
>>>
???
```

6.1.3 TEST 2 - NON-PRINTABLE CHARACTER TEST

THIS TEST CHECKS ALL NON-PRINTABLE CHARACTERS THAT HAVE NO CONTROL FUNCTION IN THE LA36 TERMINAL OR THE LA36 OPTIONS (SUCH AS CR, LF, BS, & BEL). FIRST THE ASCII CODE WILL BE PRINTED FOLLOWED BY THE MNEMONIC AFTER A FEW SEPARATING SPACES. FOLLOWING THE MNEMONIC, THE ACTUAL CONTROL CHARACTER WILL BE SENT THREE TIMES AND NOTHING SHOULD HAPPEN AT THE PRINTER. THIS PATTERN IS REPEATED, THREE TIMES ON A LINE, UNTIL ALL OF THE NON-PRINTING CHARACTERS HAVE BEEN TESTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
000  NUL  001  SOH  002  STX
006  ACK  020  DLE  021  DC1
022  DC2  023  DC3  024  DC4
025  NAK  026  SYN  027  ETB
030  CAN  031  EM   032  SUB
034  FS   035  GS   036  RS
037  US   177  DEL
```

6.1.4 TEST 3 - CARRIAGE RETURN TEST

THIS TEST CHECKS THE CARRIAGE RETURN FROM ALL EVEN NUMBERED COLUMNS AND THE SPACING OF THE SOLENOID HEAD FROM THE LEFT MARGIN. IT IS ALSO A GOOD CHECK FOR PROPER OPERATION OF THE POSITION DECODER.

THE TEST PRINTS A FULL LINE OF ALTERNATING 0'S AND SPACES, STARTING WITH A 0. AT THE END OF THE LINE THE PRINT HEAD IS RETURNED TO THE LEFT MARGIN WITH A CARRIAGE RETURN. THE SPACES ARE THEN FILLED IN BY SPACING THE PRINT HEAD OUT FROM THE LEFT MARGIN TO THE FIRST SPACE, PRINTING AN "X", AND EXECUTING A CARRIAGE RETURN. THIS PATTERN IS REPEATED UNTIL THE LINE IS COMPLETED. CHECK TO SEE THAT ALL X'S ARE IN THE MIDDLE OF THE SPACE BETWEEN THE TWO ZEROES ON EITHER SIDE OF IT.

EXAMPLE:

```
0X0X0X0X0X0X0X0X0X0X0X0X0X0X0X
```

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THIS TEST WILL PRINT A LINE OF 0'S AND SPACES, THEN PRINT A DIAGINAL LINE OF X'S. TO CORRECTLY CHECK THE ENCODER, THE AUTO LINE FEED OPTION SHOULD BE DISABLED.

EXAMPLE:

```
0 0 0 0 0 0 0 0 0
 X
  X
   X
    X
     X
      X
       X
        X
```

6.1.5 TEST 4 - MULTIPLE LINE FEED TEST

THIS TEST CHECKS THE LINE FEED CAPABILITY OF THE PRINTER BY SENDING VARIOUS GROUPS OF LINE FEEDS INTERSPACED WITH REFERENCE LINES. THE NUMBER PRINTED AS THE REFERENCE LINE INDICATES THE NUMBER OF LINE FEEDS THAT FOLLOW. THE FIRST AND LAST LINES ALSO CONTAIN A STRING OF DASHES AS REFERENCE POINTS FOR MEASURING, THE TOTAL DISTANCE IS 63(10) LINES BETWEEN THE TWO DASHED LINES.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY CARRIAGE RETURN, THE NUMBER PRINTED WILL INDICATE ONE LESS THAN THE NUMBER OF LINE FEEDS (THE NUMBER OF BLANK LINES) THAT FOLLOW. THE TOTAL DISTANCE BETWEEN THE TWO DASHED LINES WILL THEN BE 69 LINES.

EXAMPLE:

```
        01--------------------------------------------------
        02

        04



        08







        16
          \
           15 BLANK LINES
          /
        32
          \
           31 BLANK LINES
          /
        00--------------------------------------------------
```

6.1.6 TEST 5 - SINGLE LINE FEED TEST

THIS TEST IS DESIGNED TO CHECK THE TIMING OF SINGLE LINE FEEDS AND THE CAPABILITY OF DOING LINE FEEDS IN ALL COLUMNS. TWO REFERENCE LINES ARE USED BY THIS TEST (AND TEST 6) WHICH ALSO CAN BE USED TO EASILY CHECK THE NUMBER OF COLUMNS THE PRINTER IS PRINTING.

THE FIRST REFERENCE LINE CONTAINS 130(10) ZEROES FOLLOWED BY TWO 2'S IF TESTING 132(10) COLUMNS. IF LESS THAN 132 COLUMNS, THE LINE WILL CONTAIN 0'S FOR TWO LESS THAN THE MAXIMUM NUMBER OF COLUMNS FOLLOWED BY THE TWO 2'S. THIS REFERENCE LINE IS A QUICK CHECK FOR 132(10) COLUMNS IF TESTING THE FULL 132(10) COLUMNS. THE SECOND REFERENCE LINE PRINTS A STRING OF NUMBERS (1 TO 9 & 0) REPEATED TO THE MAXIMUM COLUMN. THIS LINE, AGAIN, CAN BE USED AS A QUICK CHECK OF THE NUMBER OF COLUMNS.

THE LINE FEED TEST IS ACCOMPLISHED BY: PRINTING THE FIRST REFERENCE LINE OF 0'S AND TWO 2'S; THEN EITHER SENDING 60(10) 3'S, IF TESTING 132(10) COLUMNS, OR WAITING 1.8 SECONDS FOR AN LCV, IF TESTING LESS THAN 132(10) COLUMNS. IF TESTING 132(10) COLUMNS, NOTHING SHOULD HAPPEN, EXCEPT FOR AN LCV, AT THE END OF THE LINE. THE 3'S SHOULD BE LOST AND NEVER PRINTED. AFTER THE LCV, WITH THE PRINT HEAD AT THE EXTREME RIGHT, A CARRIAGE RETURN - LINE FEED WILL BE SENT FOLLOWED BY REPEATED BACKSLASHES "\" AND LINEFEEDS TO PRINT A DIAGONAL LINE DOWN THE PAPER. WHEN A BACKSLASH IS PRINTED IN THE MAXIMUM COLUMN, A CARRIAGE RETURN WILL BE SENT IMMEDIATELY AFTER THE LINE FEED AND THE SECOND REFERENCE LINE OF SEQUENTIAL NUMBERS WILL BE PRINTED. AFTER COMPLETING THE LINE, A CARRIAGE RETURN - LINE FEED WILL BE SENT AND THE PROGRAM WILL WAIT ONE SECOND FOR THE CARRIAGE RETURN FUNCTION TO COMPLETE. AFTER THE DELAY, THE REFERENCE LINE WILL BE REPEATED, THE LAST LINE BEING GUARANTEED TO BE CORRECT. ANY TIMING PROBLEMS DURING THE LINE FEEDS WILL SHOW AS MISS PRINTS OR MISSING CHARACTERS DURING THE FIRST 16(10) CHARACTERS OF THE MIDDLE REFERENCE LINE. ALSO, ANY PAPER FEED PROBLEMS WILL CAUSE MISS-ALIGNMENT OF THE SLASHES FORMING THE DIAGONAL LINE.

EXAMPLE:

```
0000000000000000000000000000022

123456789012345678901234567890
123456789012345678901234567890
```

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE EVERY PLACE A CARRIAGE RETURN IS EXECUTED.

EXAMPLE:

```
0000000022

\
 \
  \
   \
    \
     \
      \
       \
        \
         \

1234567890

1234567890
```

6.1.7 TEST 6 - BACKSPACE TEST

THIS TEST IS DESIGNED TO TEST THE PRINT TIMING AS IN TEST 5 AS WELL AS THE BACKWARD AND FORWARD MOVEMENT OF THE PRINT SOLENOID HEAD.

THE TEST CONSISTS OF THE SAME FIRST REFERENCE LINE AS IN TEST 5 THEN A CARRIAGE RETURN-LINE FEED. A FULL LINE IS THEN PRINTED USING THE FOLLOWING PATTERN:

```
FORWARD SLASH   "/"
BACKSPACE
BACK SLASH      "\"
```

THIS PATTERN PRODUCES A LINE OF ALL X'S. THE TWO SLASHES SHOULD CROSS EXACTLY AT THE MIDDLE, PRODUCING THE X CHARACTER. WHEN THE LINE IS COMPLETED A CARRIAGE RETURN-LINE FEED IS SENT AND THE LAST TWO REFERENCE LINES ARE PRINTED AS IN TEST 5. ANY TIMING PROBLEMS WILL SHOW IN THE FIRST 16(10) CHARACTERS OF THE MIDDLE REFERENCE LINE; AGAIN AS IN TEST 5.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
000000000000000000000000000022
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
123456789012345678901234567890
123456789012345678901234567890
```

6.1.8 TEST 7 - OVERPRINT TEST

THIS TEST IS DESIGNED TO CHECK THE SPACING AND REPEATABLE PRINTING CHARACTERISTICS OF THE PRINTER. THREE ROWS OF CHARACTERS ARE EACH OVERPRINTED TWO TIMES. THE ROWS CONSIST OF THE FOLLOWING CHARACTERS ALTERNATED ACROSS THE LINE:

```
ROW 1          M-SP
ROW 2          SP-@
ROW 3          &-SP
```

THE RESULTING PATTERN WILL BE A CHECKERBOARD PATTERN AND THE OVERPRINTED CHARACTERS SHOULD BE ALIGNED PROPERLY WITH THE INITIAL CHARACTERS.

EXAMPLE:

```
M M M M M M M M M M M M M M M
 @ @ @ @ @ @ @ @ @ @ @ @ @ @
& & & & & & & & & & & & & & &
```

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THE LINES WILL NOT BE OVERPRINTED. THERE WILL BE THREE LINES OF EACH CHARACTER WITH A BLANK LINE BETWEEN EACH GROUP OF CHARACTERS. THE CHARACTERS IN EACH GROUP SHOULD BE IN THE SAME COLUMNS.

EXAMPLE:

```
M M M M M M M M M M
M M M M M M M M M M
M M M M M M M M M M

 @ @ @ @ @ @ @ @ @
 @ @ @ @ @ @ @ @ @
 @ @ @ @ @ @ @ @ @

& & & & & & & & & &
& & & & & & & & & &
& & & & & & & & & &
```

6.1.9 TEST 10 - PRINTING FREQUENCY SWEEP TEST

THIS TEST PRINTS THE CHARACTER "H" REPEATEDLY, 30(10) CHARACTERS PER LINE FOR FOUR LINES. DURING THE FIRST TWO LINES, THE TIME INTERVAL BETWEEN CHARACTERS IS INCREASED FROM 30(10) MILLISECONDS TO 1.8 SECONDS USING THE FOLLOWING FORMULA TO CREATE A LOGRITHMIC INCREASE:

NEW DELAY = OLD DELAY + OLD DELAY/16 + OLD DELAY/128

THE LAST TWO LINES DO JUST THE REVERSE. THE TIME INTERVAL BETWEEN CHARACTERS IS DECREASED FROM 1.8 SECONDS TO 30(10) MILLISECONDS USING THE FOLLOWING FORMULA TO AGAIN CREATE A LOGRITHMIC DECREASE:

NEW DELAY = OLD DELAY - OLD DELAY/16 - OLD DELAY/128

LOOK FOR POSSIBLE MISS-ALIGNMENT OF THE CHARACTERS OR SPACES BETWEEN CHARACTERS AS AN INDICATION OF TIMING PROBLEMS.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
```

6.1.10 TEST 11 - RIBBON FEED TEST

THIS TEST CHECKS THE RIBBON FEED MECHANISM BY PRINTING A SINGLE COLUMN OF 24 LINES OF X'S DOWN THE LEFT HAND MARGIN OF THE PAGE. VISUALLY CHECK FOR PROPER OPERATION OF THE RIBBON FEED MECHANISM DURING THIS TEST.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
X
X
X
X
X
X
X
```

6.1.11 TEST 12 - PRINTER BELL TEST

THIS TEST CHECKS THE PRINTER BELL BUFFER TO INSURE THAT EIGHT BELLS ARE DISTINCTLY HEARD, EVEN WHEN SENT AT THE MAXIMUM TRANSFER RATE. THE PROGRAM SENDS 8 BELL CODES AT THE MAXIMUM RATE TO THE PRINTER THEN WAITS 2.5 SECONDS TO ALLOW THE OPERATOR TO HEAR THE BELLS.

6.1.12 TEST 17 - LIFE TEST

THIS TEST RUNS CONTINUOUSLY AND IS RUN AS AN INDIVIDUAL, SPECIAL TEST, IT IS NOT PART OF THE STANDARD PRINTING TEST SEQUENCE.

THIS TEST PRINTS 2 LINES OF EACH PRINTABLE CHARACTER AND THEN REPEATS CONTINUOUSLY. THE SECOND LINE OF EACH CHARACTER IS OVERPRINTED 4 TIMES TO CONSERVE PAPER. AT THE END OF EACH COMPLETE PASS THROUGH THE CHARACTER SET A MESSAGE IS PRINTED INDICATING THE NUMBER OF PASSES EXECUTED. IF ANY CHARACTER (EXCEPT "RUBOUT") IS TYPED ON THE KEYBOARD DURING THIS TEST, THE PATTERN WILL CHANGE AND RESTART WITH THE TYPED CHARACTER. THIS WILL ONLY HAPPEN IF KEYBOARD CONTROL IS IN USE.

EXAMPLE:

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
```

IF THE AUTO LINE FEED OPTION IS SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THE TEST WILL PRINT SIX LINES OF EACH CHARACTER WITH A BLANK LINE BETWEEN THE FIRST AND SECOND LINES AS WELL AS BETWEEN EACH GROUP OF CHARACTERS.

EXAMPLE:

```
AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAA

BBBBBBBBBBBBBBBB

BBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBB
```

6.2 ECHO TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE KEYBOARD AND AN AID IN ISOLATING TROUBLES WITHIN THE TERMINAL. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED. TYPING A "RUBOUT" OR "DELETE" AT ANY TIME, WHETHER IN KEYBOARD CONTROL OR NOT, WILL EXIT THE CURRENT ECHO TEST AND PRINT A TEST TERMINATION MESSAGE. IF IN KEYBOARD CONTROL, THE SELECT TEST MESSAGE WILL BE PRINTED AND THE PROGRAM WILL AWAIT A TEST SELECTION AS USUAL. IN SWITCH REGISTER CONTROL, THE PROGRAM WILL HALT (AT SELLHLT) WAITING FOR CONTROL VIA THE SWITCH REGISTER. A DETAILED DESCRIPTION OF EACH TEST FOLLOWS:

6.2.1 TEST 20 - CHARACTER ECHO TEST

THIS TEST IS DESIGNED TO OPERATE THE TERMINAL IN A SIMULATED LOCAL MODE. ANY CHARACTER TYPED ON THE KEYBOARD (EXCEPT A "RUBOUT") WILL BE ECHOED TO THE PRINTER.

IF THE LA36 TERMINAL IS IN HALF DUPLEX WITH THE AUTO LINE FEED OPTION AVAILABLE, TYPING A CARRIAGE RETURN MAY CAUSE A GARBLED RESPONSE ON THE TERMINAL DURING THIS TEST.

6.2.2 TEST 21 - LINE ECHO TEST, FAST RATE

THIS TEST CONTINUALLY SENDS FULL LINES OF ANY CHARACTER UP TO THE MAXIMUM COLUMN WIDTH. THE TEST PRINTS A "0" CHARACTER WHEN STARTED UNTIL A KEY IS TYPED ON THE KEYBOARD. THE PROGRAM WILL THEN SEND THE TYPED CHARACTER UNTIL ANOTHER CHARACTER IS TYPED OR THE TEST IS TERMINATED BY TYPING A "RUBOUT". THE CHARACTERS ARE TRANSMITTED AT THE MAXIMUM RATE WITH A CARRIAGE RETURN-LINE FEED INSERTED AFTER EVERY 132(10) PRINTABLE CHARACTERS.

IF THE LA36 IS IN HALF DUPLEX WHEN RUNNING THIS TEST, CHARACTERS MAY BE LOST OR GARBLED WHENEVER A CHARACTER IS TYPED ON THE KEYBOARD.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

6.2.3 TEST 22 - LINE ECHO TEST, SLOW RATE

THIS TEST IS IDENTICAL TO TEST 21 EXCEPT A DELAY OF 1.8 SECONDS IS INSERTED BETWEEN EACH CHARACTER TO ALLOW THE PRINT HEAD TO PERFORM AN LCV BETWEEN CHARACTERS.

6.2.4 TEST 23 - CHARACTER/CODE ECHO TEST

THIS TEST WILL PRINT THE OCTAL CODE RECEIVED BY THE PROCESSOR FOLLOWED BY THE CHARACTER OR THE MNEMONIC OF THE CHARACTER EVERY TIME A KEY IS PRESSED ON THE KEYBOARD. THE PARITY OF THE RECEIVED CODE WILL BE INDICATED AS EITHER ODD OR EVEN. ALLOW SUFFICIENT TIME BETWEEN CHARACTERS FOR THE LINE TO BE PRINTED.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BETWEEN EACH PRINTED LINE.

EXAMPLE:

```
301   A    ODD
263   3    ODD
215   CR   EVEN
240   SP   EVEN
```

6.2.5 TEST 24 - SELECTED PATTERN ECHO TEST

THIS TEST IS DESIGNED TO GIVE MAINTENANCE THE FLEXIBILITY TO CHOOSE THEIR OWN PATTERNS FOR ISOLATING ANY SPECIFIC PROBLEMS WHICH MAY ARISE IN THE FIELD.

TYPE ANY CHARACTERS (EXCEPT CONTROL-C AND RUBOUT) AND EACH CHARACTER WILL BE ECHOED AS TYPED. A MAXIMUM OF 256(10) CHARACTERS MAY BE INPUTTED. NO CARRIAGE RETURNS OR LINE FEEDS ARE INSERTED BY THE PROGRAM. ALL CHARACTERS MUST BE INPUTTED BY THE OPERATOR. TO TERMINATE THE INPUT STRING TYPE A CONTROL-C, THE PROGRAM WILL THEN CONTINUALLY ECHO THE INPUTTED PATTERN. TO STOP THE PRINTING, TYPE CONTROL-C. THE PROGRAM WILL STOP PRINTING THE PATTERN AND WILL WAIT FOR EITHER ANOTHER PATTERN INPUT TERMINATED BY A CONTROL-C, OR THE SAME PATTERN MAY BE USED AGAIN BY TYPING CONTROL-C. TO EXIT THE TEST AT ANY TIME, TYPE A "RUBOUT".

WHEN ANY OPTIONS ARE AVAILABLE, BE CAREFUL WHAT CHARACTERS OR CHARACTER SEQUENCES ARE SELECTED.

6.2.6 TEST 25 - BELL ECHO TEST

THIS TEST IS DESIGNED TO TEST THE BELL ON COLUMN 64 IF TYPING HAS OCCURED ON THE LINE. THE TEST PRINTS A MESSAGE:

TYPE ANY PRINTABLE CHARACTER AND LISTEN FOR BELL ....

AFTER THE TEST MESSAGE IS PRINTED, TYPE ANY PRINTABLE CHARACTER ON THE KEYBOARD. THE CHARACTER WILL BE ECHOED AND THE BELL SHOULD RING. THE MESSAGE WILL THEN BE TYPED AGAIN. TYPE THE "RUBOUT" KEY TO TERMINATE THE TEST AT ANY TIME.

6.3 OPTION TESTS

THESE TESTS ARE DESIGNED AS A TEST OF THE VARIOUS OPTIONS IN WHATEVER COMBINATIONS THEY ARE AVAILABLE IN THE LA36. AT THE BEGINNING OF EACH TEST, THE TEST NUMBER WILL BE PRINTED INDICATING WHICH TEST IS BEING EXECUTED. TYPING A "RUBOUT" OR "DELETE" AT ANY TIME, WHETHER IN KEYBOARD CNTROL OR NOT, WILL EXIT THE CURRENT OPTION TEST. A DETAILED DESCRIPTION OF EACH TEST FOLLOWS:

6.3.1 TEST 30 - SECONDARY CHARACTER SET OPTION

THIS TEST IS DESIGNED TO TEST THE SECONDARY CHARACTER SET OPTION, TESTING THE ABILITY TO SELECT EITHER CHARACTER SET UNDER SOFTWARE CONTROL FROM THE CPU AND PRINTING THE CORRECT CHARACTERS WITHIN EACH CHARACTER SET.

A NUMBER IS PRINTED AT THE LEFT MARGIN INDICATING WHICH CHARACTER SET IS BEING PRINTED, #1 INDICATES THE PRIMARY SET AND #2 INDICATES THE SECONDARY SET (APL). AFTER THE NUMBER, THE APPROPRIATE SHIFT IN (SI) OF SHIFT OUT (SO) WILL BE SENT FOLLOWED BY THE ENTIRE PRINTABLE CHARACTER SET. IF LESS THAN 98 COLUMNS ARE BEING TESTED, A CARRIAGE RETURN - LINE FEED WILL BE INSERTED IN THE APPROPRIATE PLACES. THIS WILL BE REPEATED, ALTERNATING BETWEEN PRIMARY AND SECONDARY SETS UNTIL 16 LINES HAVE BEEN PRINTED (IF USING 98 OR MORE COLUMNS). THERE WILL BE A BLANK LINE BETWEEN EACH PAIR OF LINES TO SEPARATE EACH GROUPING. CHANGE LOCATION "T30SC" AT THE END OF TEST30 TO 377 (8) IF USING 8 BIT SELECTION CODE RATHER THAN THE SI AND SO TO SELECT CHARACTER SETS. THE TEST WILL THEN SET OR CLEAR BIT 8 INSTEAD OF SENDING THE SI OR SO TO SELECT CHARACTER SETS.

WITH THE AUTO LINE FEED OPTION SET TO PRODUCE AN AUTOMATIC LINE FEED AFTER EACH RECEIVED CARRIAGE RETURN, THERE WILL BE EXTRA BLANK LINES EVERY PLACE A CARRIAGE RETURN IS SENT.

EXAMPLE:

```
          #1= !"#$%&'()........PRIMARY CHARACTER SET..........
          #2= !"#$%&'().......SECONDARY CHARACTER SET.........

          #1= !"#$%&'()........PRIMARY CHARACTER SET..........
          #2= !"#$%&'().......SECONDARY CHARACTER SET.........
```

6.3.2 TEST 31 - SELECTIVE ADDRESSING OPTION

THIS TEST IS DESIGNED TO TEST THE VARIOUS FUNCTIONS OF THE SELECTIVE ADDRESSING OPTION. THE TEST FIRST SENDS AN "EOT" <004> TO DISABLE ALL TERMINALS AND TRIES TO PRINT AN ERROR MESSAGE. THE ERROR MESSAGE SHOULD NOT BE PRINTED ON ANY TERMINAL WITH THE SELECTIVE ADDRESSING OPTION. THEN A "BEL" <007> AND "STX" <002> ARE SENT TO SELECT ALL TERMINALS. AT THIS POINT THE TEST NUMBER IS PRINTED ON ALL TERMINALS. THUS, IF AN ERROR MESSAGE IS PRINTED BEFORE THE TEST NUMBER, THE EOT DID NOT DE-SELECT THE TERMINAL WHERE THE MESSAGE WAS PRINTED.

THE TEST NEXT SENDS AN EOT DIRECTLY FOLLOWED BY A STX, WITH NO SELECT CHARAACTER. AGAIN, THE ERROR MESSAGE IS SENT TO ALL TERMINALS, WHICH SHOULD NOW BE ALL DE-SELECTED. THE ERROR MESSAGE SHOULD NOT BE PRINTED ON ANY TERMINAL WITH THE SELECTIVE ADDRESSING OPTION.

THE NEXT SERIES OF CHECKS ARE MADE ON THE GROUP SELECT CHARACTER. A TABLE LOCATED AT THE END OF TEST 31 IN THE LISTING IS USED TO TEST VARIOUS GROUP SELECT CHARACTERS. THE FIRST ZERO ENCOUNTERED IN THE TABLE WILL INDICATE THE END OF THE TABLE AND THE TEST WILL GO TO THE NEXT SERIES OF CHECKS ON THE OPTION. THE TABLE IS PRESET WITH A SINGLE GROUP SELECT CHARACTER, THE LETTER "G", BUT ALLOWS ROOM TO TEST UP TO 8 DIFFERENT SELECT CODES. THIS TABLE SHOULD BE CHANGED TO CONTAIN THE VARIOUS GROUP SELECT CHARACTERS DESIRED TO TEST WITH ONE ASCII CODE PER LOCATION. THE TEST WILL THEN USE THE VARIOUS GROUP SELECT CHARACTERS TO SELECT TERMINALS AND PRINT A MESSAGE ON EACH SELECTED TERMINAL INDICATING THE GROUP SELECT CHARACTER USED. CHECK THAT THE CORRECT GROUP SELECT CHARACTER HAS ENABLED EACH TERMINAL. ALSO, IT MAY BE HELPFUL TO PLACE ;UNUSED SELECT CHARACTERS IN THE TABLE TO CHECK THAT THEY DO NOT SELECT TERMINALS. IF AN ERROR MESSAGE WAS PRINTED BETWEEN THE TEST NUMBER AND THE GROUP SELECT MESSAGE, THE TERMINAL WHERE THE MESSAGE WAS PRINTED WAS SELECTED BY AN EOT AND STX WITH NO SELECT CHARACTER BETWEEN THEM.

THE LAST SERIES OF CHECKS ARE MADE ON THE UNIQUE SELECT CHARACTER. A TABLE LOCATED AT THE END OF TEST 31 IN THE LISING IS USED TO TEST VARIOUS UNIQUE SELECT CHARACTERS. THE FIRST ZERO ENCOUNTERED IN THE TABLE WILL INDICATE THE END OF THE TABLE. THE PROGRAM WILL SELECT ALL TERMINALS USING THE BEL CODE BEFORE EXITING THE TEST. THE TABLE IS PRESET WITH A SINGLE UNIQUE SELECT CHARACTER, THE LETTER "U", BUT ALLOWS ROOM TO TEST UP TO 16 DIFFERENT UNIQUE SELECT CODES. THIS TABLE SHOULD BE CHANGED TO CONTAIN THE VARIOUS UNIQUE SELECT CHARACTERS DESIRED TO TEST, WITH ONE ASCII CODE PER LOCATION. MAKE SURE THAT EACH CHARACTER IN THE TABLE IS A VALID UNIQUE SELECT CODE OR THE DIAGNOSTIC WILL HANG DURING THIS PORTION OF THE TEST. USING EACH UNIQUE SELECT CHARACTER IN TURN, THE TEST WILL PERFORM THE REMAINING CHECKS OF THE SELECTIVE ADDRESSING OPTION.

THE TEST WILL SEND AN EOT FOLLOWED BY THE CURRENT UNIQUE SELECT CHARACTER. BEFORE THE STX IS SENT, THE TEST WILL TRY TO PRINT THE ERROR MESSAGE ON ALL TERMINALS. THEN THE STX WILL BE SENT AND A MESSAGE WILL BE PRINTED TO INDICATE THE UNIQUE SELECT CHARACTER USED. CHECK THAT THE CORRECT UNIQUE SELECT CHARACTER HAS ENABLED EACH TERMINAL. IF AN ERROR MESSAGE IS PRINTED BEFORE THE UNIQUE SELECT MESSAGE, THE TERMINAL WHERE THE MESSAGE WAS PRINTED WAS ENABLED BEFORE THE STX WAS RECEIVED. A MESSAGE WILL THEN BE PRINTED TELLING THE OPERATOR TO TYPE ANY PRINTABLE CHARACTER TO CHECK THAT THE KEYBOARD IS ENABLED. WHATEVER CHARACTER IS TYPED WILL BE ECHOED TO THE TERMINAL.

THE FINAL SECTION OF THE TEST WILL USE A DUMMY SELECT CHARACTER. THE ASCII CODE FOR THIS SELECT CHARACTER IS LOCATED BETWEEN THE TWO SELECT CHARACTER TABLE AT THE END OF THE TEST. THIS LOCATION SHOULD CONTAIN THE ASCII CODE OF ANY UNUSED SELECT CHARACTER. THE TEST WILL SEND AN EOT FOLLOWED BY THE DUMMY SELECT CHARACTER AND AN STX. THE ERROR MESSAGE WILL BE LOADED TO ALL TERMINALS AND SHOULD NOT BE PRINTED ON ANY TERMINALS SINCE ALL SHOULD BE DE-SELECTED. NEXT AN ETX <003> FOLLOWED BY THE CURRENT UNIQYUE SELECT CHARACTER AND AN STX WILL BE SENT AND A PRINTED MESSAGE WILL INDICATE THE SELECT CHARACTER USED. ANOTHER EXT WILL BE SENT, FOLLOWED BY THE DUMMY SELECT CHARACTER AND AN STX THIS TIME. A MESSAGE WILL AGAIN BE PRINTED INDICATING THE CURRENT UNIQUE SELECT CHARACTER. ALL SELECTED TERMINALS SHOULD REMAIN SELECTED AND NO OTHER TERMINALS SHOULD GET SELECTED.

6.3.3 TEST 32 - ANSWER BACK OPTION

THIS TEST IS DESIGNED TO TEST THAT THE ANSWER BACK OPTION SENDS THE CORRECT MESSAGE UPON RECEIPT OF AN ENQ (005) OR UPON TYPING CONTROL-E OR THE HERE IS KEY ON THE KEYBOARD. THE TEST WILL SEND AN ENQ (005), READ THE MESSAGE, AND THEN PRINT OUT THE MESSAGE ON THE LA36. THE TEST WILL THEN ASK THE OPERATOR TO DEPRESS THE HERE IS KEY, READ THE MESSAGE, AND THEN PRINT OUT THE MESSAGE. FINALLY, THE TEST WILL TELL THE OPERATOR TO DEPRESS THE CONTROL-E KEY, READ THE MESSAGE, AND PRINT OUT THE MESSAGE. IF THE SELECTIVE ADDRESSING OPTION IS AVAILABLE, THE AUTO ANSWER BACK OPTION WILL NOT RESPOND TO ANOTHER ENQ AFTER THE FIRST ONE RECEIVED. THUS, YOU MAY HAVE TO DEPRESS THE RUBOUT KEY TO EXIT THE TEST.

6.3.4 TEST 33 - TOP OF FORM OPTION

THIS TEST IS DESIGNED TO TEST THE FORM FEED CAPIBILITY OF THE TOP OF FORM OPTION. A SET OF INSTRUCTIONS IS PRINTED FOR THE OPERATOR TO REMIND HIM TO DEPRESS THE TOP OF FORM RESET SWITCH AFTER MAKING EACH SWITCH SETTING. UPON COMPLETION OF EACH SETTING, AFTER DEPRESSING THE RESET SWITCH, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD TO TEST THAT SWITCH SETTING. THE REFERENCE LINES PRINTED WILL INDICATE THE LENGTH FORM FEED JUST EXECUTED AND THE NEXT SWITCH SETTING TO MAKE. THE 3 INCH FORM FEED IS TESTED TWICE BEFORE TESTING THE REMAINING POSITIONS. THE FIRST TIME, 16 OR 17 LINE FEEDS ARE EXECUTED BEFORE DOING THE FORM FEED, DEPENDING ON HOW THE AUTO LINE FEED OPTION IS SET UP. THE DIAGNOSTIC WILL THEN TEST EACH POSITION IN SEQUENCE FROM 3 TO 14 INCHES. THE SINGLE STEP POSITION IS NOT CHECKED.

6.3.5 TEST 34 - HORIZONTAL TAB OPTION

THIS TEST CHECKS THE ABILITY TO SET A TAB IN EVERY COLUMN AND AT PREDETERMINED INTERVALS, AS WELL AS THE ABILITY TO CLEAR ALL TABS. THE PROGRAM SETS A TAB IN THE PREDETERMINED COLUMN, DOES A BACKSPACE, AND PRINTS AN "O". AFTER THE LINE IS PRINTED AND THE TABS ARE SET, A CARRIAGE RETURN IS SENT AND THEN THE PRINT HEAD IS POSITIONED USING TABS AND X'S ARE PRINTED OVER THE O'S. SINCE THE FIRST LINE OF THE TEST SETS A TAB IN EVERY COLUMN, THE PRINT HEAD IS TABED ACROSS THE PAGE TWICE TO TEST ALL TABS. THE FIRST PASS CHECKS THE EVEN NUMBERED COLUMNS WHILE THE SECOND PASS CHECKS THE ODD NUMBERED COLUMNS. THE TEST SETS TABS IN EVERY COLUMN, EVERY OTHER COLUMN, AND EVERY 4, 8, 16, 32, 64, 128, & 132 COLUMNS. ALL HORIZONTAL TABS WILL BE CLEARED AT THE END OF THE TEST IF THE TEST IS RUN TO COMPLETION. IF A RUBOUT IS USED TO EXIT THE TEST BEFORE COMPLETION, THE TABS WILL STILL BE SET.

EXAMPLE:

```
          0000000000
           0 0 0 0 0
             0   0
                 0
```

WHEN THE AUTO LINE FEED OPTION IS SET UP TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN, THERE WILL BE A BLANK LINE BEFORE EACH REFERENCE LINE OF O'S AND THE X'S WIL BE PRINTED ON THE NEXT LINE UNDER THE O'S. THE FIRST LINE OF O'S WILL HAVE 2 LINES OF X'S UNDER IT, THE FIRST HAVING X'S IN ALL EVEN NUMBERED COLUMNS AND THE SECOND HAVING X'S IN ALL ODD NUMBERED COLUMNS.

EXAMPLE:

```
          000000000000
           X X X X X X
          X X X X X X

           0 0 0 0 0 0
           X X X X X X

             0   0   0
             X   X   X

                 0
                 X
```

6.3.6 TEST 35 - VERTICAL TAB OPTION

THIS TEST CHECKS THE VERTICAL TAB OPTION BY TESTING THE ABILITY TO SET TABS IN VARIOUS POSITIONS OF A 14 INCH FORM. AN INSTRUCTION IS PRINTED TELLING THE OPERATOR TO SET A 14 INCH FORM LENGTH AND DEPRESS THE TOP OF FORM RESET SWITCH. WHEN READY, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD TO CONTINUE. THE TEST WILL SEND LINE FEEDS, SET TABS, AND PRINT REFERENCE LINES WHEREVER A TAB IS SET. AT THE END OF THE FORM, A MESSAGE WILL INDICATE TO EITHER REMOVE THE REFERENCE PAGE (WITHOUT TOUCHING THE KEYBOARD) OR RESET THE FIRST REFERENCE LINE. TO RESET THE REFERENCE PAGE IN THE PRINTER, OPEN THE PAPER TRACTORS AND PLACE THE FIRST REFERENCE LINE INFRONT OF THE PRINT HEAD. WHEN READY TO CONTINUE, TYPE ANY CHARACTER (EXCEPT RUBOUT) ON THE KEYBOARD. THE TEST WILL THEN REPRINT THE REFERENCE LINES, USING THE TABS INSTEAD OF LINE FEEDS TO ADVANCE THE PAPER. IF THE FIRST REFERENCE PAGE WAS REMOVED, HOLD IT AGAINST THE SECOND REFERENCE PAGE TO CHECK FOR PROPER PAPER ADVANCING USING TABS. IF THE REFERENCE PAGE WAS RESET IN THE PRINTER, THE SECOND SET OF REFERENCE LINES SHOULD HAVE PRINTED DIRECTLY OVER THE FIRST SET EXCEPT ON THE FIRST LINE WHERE THEY SHOULD BE SIDE-BY-SIDE. ALLOW FOR A SLIGHT VARIANCE IN PAPER POSITION WHEN CHECKING THAT THE REFERENCE LINES ARE CORRECT. LOOK FOR FULL LINE DIFFERENCES. THE TEST PRODUCES 0,1,2,3,4,5,6,7,8,9, & 10 BLANK LINES BETWEEN THE REFERNECES LINES, IN THAT ORDER.

EXAMPLE:

```
        --------<<<<<<<<    --------<<<<<<<<
        --------<<<<<<<<      !
                              !
        --------<<<<<<<<      !
                              !----- FIRST REF.  LINE WITH TABS

        --------<<<<<<<<


        --------<<<<<<<<



        --------<<<<<<<<
```

THE REFERENCE PATTERN PRODUCED WILL BE DIFFERENT IF THE AUTO LINE FEED OPTION IS SET UP TO PRODUCE AN AUTOMATIC LINE FEED AFTER EVERY RECEIVED CARRIAGE RETURN. THERE WILL, IN FACT, BE ONE EXTRA LINE BETWEEN EACH REFERENCE LINE DUE TO THE AUTO LINE FEED.

EXAMPLE:

```
        --------<<<<<<<<    --------<<<<<<<<

        --------<<<<<<<<


        --------<<<<<<<<



        --------<<<<<<<<




        --------<<<<<<<<
```

6.4 STANDARD I-O TESTS

THESE TESTS ARE DESIGNED AS A BRIEF CHECK OF THE CONSOLE TERMINAL INTERFACE LOGIC. EACH CHECK IS STRUCTURED AS AN INDEPENDENT TEST AND THE SWITCH REGISTER CONTROLS MAY BE USED. A DESCRIPTION OF EACH TEST IS GIVEN IN THE PROGRAM LISTING. ANY ERRORS ENCOUNTERED DURING THE I/O TESTS WILL CAUSE A HALT AT LOCATION "ERRHLT" IF SWITCH 14 IS DOWN.

```
			.TITLE	MAINDEC-11-DZLAC-C
;
;LA36 DIAGNOSTIC (DL11 & KL11 INTERFACE)
;
;
;AUTHOR:   ROBERT W. BAKER
;COPYRIGHT 1974,1975 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754



			.SBTTL	SWITCH REGISTER OPTIONS
;
;SWITCH		POSITION	FUNCTION
;
;  15		UP (1)		HALT AT COMPLETION OF CURRENT TEST
;		DOWN (0)	CONTINUE NORMAL TET SEQUENCE
;
;  14		UP (1)		CONTINUE ON ERROR
;				HALT ON ERROR
;
;  13		UP (1)		DRIVE ONLY CONSOLE TERMINAL
;		DOWN (0)	DRIVE ALL TERMINALS
;
;  11		UP (1)		LOOP ON INDIVIDUAL TEST
;		DOWN (0)	NORMAL TEST SEQUENCE
;
;  08		UP (1)		HALT AND SELECT TEST AT COMPLETION OF CURRENT TEST
;		DOWN (0)	LOOP ON TEST SEQUENCE
;
; 05-00				TEST # SELECTION
;
; 07-00				# OF COLUMNS AT START-UP
```

```
                                        .SBTTL  SPECIAL OPERTIONAL INFORMATION

                               ;1.--    THE STANDARD CONSOLE TERMINAL INTERRUPT VECTOR AND REGISTER
                               ;        ADDRESSES ARE USED. TO REDEFINE THE LOCATION OF THE CONSOLE
                               ;        TERMINAL THE SYMBOLIC LOCATIONS "CONADD" AND "CONVEC" SHOULD
                               ;        BE CHANGED BEFORE START UP.
                               ;
                               ;2.--    BEFORE START UP REFER TO THE DESCRIPTION OF THE ROUTINE "DLY"
                               ;        TIMING IS A FUNCTION OF THE PDP11 MODEL AND MEMORY TYPE AND
                               ;        SHOULD BE SET UP BEFORE RUNNING THE DIAGNOSTIC.




                                        .SBTTL  SYSTEM EQUATES
                               ;
                               ;
                               ;        REGISTER EQUATES
                               ;
        000000                          R0=%0
        000001                          R1=%1
        000002                          R2=%2
        000003                          R3=%3
        000004                          R4=%4
        000005                          R5=%5
        000006                          SP=%6
        000007                          PC=%7
        177776                          PSW=177776
```

```
                        ;       SYSTEM EQUATES
                        ;
        000001                  BIT0=1
        000002                  BIT1=2
        000004                  BIT2=4
        000010                  BIT3=10
        000020                  BIT4=20
        000040                  BIT5=40
        000100                  BIT6=100
        000200                  BIT7=200
        000400                  BIT8=400
        001000                  BIT9=1000
        002000                  BIT10=2000
        004000                  BIT11=4000
        010000                  BIT12=10000
        020000                  BIT13=20000
        040000                  BIT14=40000
        100000                  BIT15=100000
        000000                  OPEN=0
        040000                  SCOPSW=BIT14            ;SCOPE SWITCH
        004000                  NITRSW=BIT11            ;TEST LOOP SWITCH
        005726                  POPSP=5726              ;POP STACK ONCE
        022626                  POPSP2=22626            ;POP STACK TWICE
        000340                  PRTY7=340               ;PRIORITY LEVEL DEFINITIONS
        000200                  PRTY4=200
        000200                  ACRLF=200

                        ;
                        ;       PROGRAM TRAP EQUATES
                        ;
        104000                  TYPE=EMT+0
        104001                  ERROR=EMT+1
        104002                  EHALT=EMT+2
        104003                  STRDRV=EMT+3
        104004                  STPCHV=EMT+4
        104005                  CHAIN=EMT+5
        104006                  CHALT=EMT+6
        104007                  TYPEM=EMT+7
        104010                  DELAY=EMT+10
        104011                  TTYCTL=EMT+11
        104012                  CRLF=EMT+12
        104013                  SCRLF=EMT+13
        104014                  LF=EMT+14
        104015                  PRINTC=EMT+15
        104016                  PRTHDR=EMT+16
        104017                  PRNT=EMT+17
        104020                  READ=EMT+20
        104021                  AREAD=EMT+21
        104022                  CR=EMT+22
        104023                  BTOASC=EMT+23
        104024                  FORWD=EMT+24
        104025                  READC=EMT+25
```

```
                                         .SBTTL  TRAP CATCHER & STARTING ADDRESSES
                                 ;
                                         .ENABL  ABS,AMA


        000000                           .=0

000000  000002                           .+2                     ;UNASSIGNED TRAP
000002  000000                           HALT
000004  000006                   MACHER: .+2                     ;SP OVERFLOW, BUS ERROR TRAP
000006  000000                           HALT
000010  000012                           .+2                     ;RESERVED INSTRUCTION TRAP
000012  000000                           HALT
000014  000016                           .+2                     ;TRACE TRAP
000016  000000                           HALT
000020  000022                           .+2                     ;TRAP TO CALL IOX
000022  000000                           HALT
000024  000026                           .+2                     ;POWER FAIL TRAP
000026  000000                           HALT
000030  003176                           EMTINT                  ;EMT TRAP
000032  000340                           PRTY7

        000042                           .=42

000042  000000                           0

        000046                           .=46

000046  011426                           LOGICAL

        000052                           .=52

000052  010000                           010000

        000174                           .=174

000174  000000                   DISPREG:        .WORD   0       ;SOFTWARE DISPLAY
000176  000000                   SWREG:          .WORD   0       ;SOFTWARE SWITCH REGISTER

000200  000137  001304                   JMP     START           ;START UP WITH I/O TEST RUNNING
000204  000137  001232                   JMP     START1          ;START UP, SKIP ALL I/O TEST
000210  000137  001250                   JMP     START2          ;START UP TERMINAL CONTROL WITH I/O TEST
000214  000137  001266                   JMP     START3          ;START UP TERM CNTL WITHOUT I/O TESTS


        001100                           .=1100

001100  000000                   SPBOT:  0                       ;BOTTOM OF STACK
```

```
                                .SBTTL  SYMBOL DEFINITIONS
                        ;
                        ;
                        ;
001102  177560          CONADD: 177560          ;ADDR OF CONSOLE RECEIVER STATUS REG
001104  000060          CONVEC: 60              ;CONSOLE TERMINAL INTERRUPT VECTOR
001106  176500          DLADR:  176500          ;ADDRESS OF FIRST DL11, DEFAULT TO DL11-A,B
                                                ;SET TO 175610 FOR FIRST 16 (OF 31) OR
                                                ;SET TO 176000 FOR LAST 16 (OF 31)
                                                ;DL11-C,D,E
001110  177560          TKS:    177560          ;CONSOLE RECEIVER STATUS REG
001112  177562          TKB:    177562          ;CONSOLE RECEIVER BUFFER
001114  177564          TPS:    177564          ;CONSOLE TRANSMITTER STATUS REG
001116  177566          TPB:    177566          ;CONSOLE TRANSMITTER BUFFER
001120  000060          TKVTR:  60              ;C.T. RECEIVER INTERRUPT VECTOR
001122  000200          TKLVL:  PRTY4           ;C.T. RECEIVER PRIORITY LEVEL
001124  000064          TPVTR:  64              ;C.T. TRANSMITTER INTERRUPT VECTOR
001126  000200          TPLVL:  PRTY4           ;C.T. TRANSMITTER PRIORITY LEVEL
001130  000000          FSTDL:  OPEN            ;ADDRESS OF FIRST ACTIVE DL11
001132  000000          CNTLSW: OPEN            ;CONSOLE TERMINAL CONTROL SWITCH
001134  000000          RTNNO:  OPEN            ;CONTAINS CURRENT TEST NUMBER
001136  000000          NXTST:  OPEN            ;CONTAINS ADDRESS OF NEXT TEST
001140  000000          SCOPTR: OPEN            ;CONTAINS ADDRESS OF TEST SCOPE ENTRY
001142  000000          PRGID:  OPEN            ;CONTAINS TEST PROGRAM INDICATORS
001144  000000          CRBUF:  OPEN
001146  000000          CTRA:   OPEN
001150  000000          WIDTH:  OPEN            ;CURRENT PAPER WIDTH, BINARY
001152  000000          LEVEL:  OPEN            ;LEVEL OF EXECUTION
001154  000000          DLCNT:  OPEN            ;# OF MULTIPLE DL11S
001156  000000          ICTR:   OPEN            ;I/O TEST ITERATION COUNT
001160  000000          REPT:   OPEN            ;TEMP STORAGE FOR TESTS E021&E022
001162  000000          BRCTR:  OPEN            ;COUNTER FOR ROUTINE "AREAD"
001164  000000          COUNT3: OPEN            ;COUNTER FOR ROUTINE "PRINTC"
001166  000000          XCSR:   OPEN            ;ADDRESS OF MULTIPLE DL11 STATUS
001170  000251          TIMER:  251             ;1 MSEC COUNTER FOR ROUTINE "DELAY"
001172  000000          SPCNT:  OPEN            ;COUNTER FOR TEST ROUTINE "PT3"
001174  000000          CURTST: OPEN            ;ADDRESS OF CURRENT TEST
001176  000000          TEMPCH: OPEN            ;TEMP STOR FOR ECHO TESTS
001200  000000          PARITY: OPEN            ;PARITY FLAG FOR RECEIVED CHAR
001202  000000          PCHAR:  OPEN            ;CHAR CODE WITH PARITY BIT
001204  000000          LFCNT:  OPEN            ;COUNTER FOR TEST ROUTINE "PT4"
001206  000000          INCHK:  OPEN            ;CHECK FOR INPUT FLAG
001210  000000          TEMP:   OPEN            ;TEMPORARY WORKING STORAGE
001212  177570          SR:     177570          ;SW REG ADDRESS
```

```
                                                .SBTTL  PROGRAM INITIALIZATION & CONTROL

                                        ;********
                                        ;COMMON HALT---WHEN IN SWITCH REGISTER CONTROL THE CPU
                                        ;       WILL BE ADVANCED TO THIS COMMON HALT WHERE
                                        ;       A NEW TEST WILL BE EXPECTED TO BE STARTED
                                        ;********

001214  005737  001152          CHLT:   TST     LEVEL           ;TEST CURRENT LEVEL
001220  001403                          BEQ     SELHLT          ;BRANCH IF 0, DO NOT HALT
001222  011600                          MOV     @SP,R0          ;PUT ADDRESS OF CALLER INTO R0
001224  005740                          TST     -(R0)
001226  000000                          HALT
001230  000002                  SELHLT: RTI                     ;RETURN FROM INTERRUPT
001232  012737  177777  005522  START1: MOV     #177777,ATOX    ;FORCE END OF I/O TESTS
001240  012737  104006  002154          MOV     #CHALT,WAITF    ;FORCE SR CONTROL
001246  000424                          BR      STARTX
001250  012737  104011  002154  START2: MOV     #TTYCTL,WAITF   ;FORCE TERMINAL CONTROL
001256  012737  005552  005522          MOV     #AT1,ATOX       ;FORCE ALL I/O TESTS
001264  000415                          BR      STARTX
001266  012737  104011  002154  START3: MOV     #TTYCTL,WAITF   ;FOECE TERMINAL CONTROL
001274  012737  177777  005522          MOV     #177777,ATOX    ;FORCE END OF I/O TESTS
001302  000406                          BR      STARTX
001304  012737  005552  005522  START:  MOV     #AT1,ATOX       ;FORCE ALL I/O/TESTS
001312  012737  104006  002154          MOV     #CHALT,WAITF    ;FORCE SR CONTROL
001320  012706  001100          STARTX: MOV     #SPBOT,SP       ;SET STACK POINTER
001324  013746  000006                  MOV     6,-(SP)         ;SAVE CURRENT VECTOR
001330  013746  000004                  MOV     4,-(SP)
001334  012737  001350  000004          MOV     #10$,4          ;SET UP TIMEOUT VECTOR
001342  005777  177644                  TST     @SR             ;TRY TO REFERENCE HARDWARE SW REG
001346  000404                          BR      11$             ;BRANCH IF NO TIMEOUT TRAP OCCURS
001350  012737  000176  001212  10$:    MOV     #SWREG,SR       ;POINT TO SOFTWARE SWITCH REGISTER
001356  022626                          CMP     (SP)+,(SP)+     ;RESTORE STACK
001360  012637  000004          11$:    MOV     (SP)+,4         ;RESTORE TIMEOUT VECTOR
001364  012637  000006                  MOV     (SP)+,6
001370  005037  001206                  CLR     INCHK           ;ALLOW INPUT CHECKING
001374  012737  000006  000004          MOV     #6,MACHER       ;CLEAN UP
001402  005037  001142                  CLR     PRGID           ;INITIALIZE PROGRAM FLAGS
001406  005037  001132                  CLR     CNTLSW          ;INITIALIZE TERMINAL CONTROL SWITCH
001412  005037  001152                  CLR     LEVEL           ;INITIALIZE LEVEL
001416  012737  003672  000024          MOV     #PFAIL,24       ;SET ADDR POWER FAIL ROUTINE
001424  004737  004122                  JSR     PC,CONIT        ;SET UP CONSOLE TERMINAL ADDRESS
```

```
                                        ;**********
                                        ;READ THE PAPER WIDTH, NUMBER OF COLUMNS,
                                        ;FROM SWITCH REGISTER POSITIONS 0-7. SAVE AND
                                        ;CONVERT TO 3 ASCII CHARACTERS. A WIDTH GT132
                                        ;OR LT30 COLUMNS (DECIMAL) WILL BE ABORTED TO 132.
                                        ;THE SWITCHES MAY BE CHANGED ONCE THE PROGRAM TITLE OR THE DL11 COUNT
                                        ;MESSAGE HAS STARTED TO PRINT.
                                        ;**********

001430  017701  177556                  MOV     @SR,R1          ;PUT (SR) INT R1
001434  042701  177400                  BIC     #177400,R1      ;SAVE ONLY BITS 0-7
001440  020127  000204                  CMP     R1,#204         ;TEST NO. COLUMN GT132
001444  003003                          BGT     2$              ;COLUMNS GT132, DEFAULT TO 132
001446  020127  000035          1$:     CMP     R1,#35          ;CHECK IF NO. COLUMNS LT 30
001452  101002                          BHI     3$              ;NOT LT 30 NOR GT 132
001454  012701  000204          2$:     MOV     #204,R1         ;COLUMNS LT 30 OR GT 132, DEFAULT
001460  010137  001150          3$:     MOV     R1,WIDTH        ;SAVE NO. COLUMNS IN WIDTH
001464  012700  016715                  MOV     #HDR0,R0        ;ADDR TO STORE ASCII COLUMN VALUE
001470  012702  000003                  MOV     #3,R2           ;DO A 3 CHAR. CONVERSION
001474  104023                          BTOASC                  ;CONVERT NO. COLUMNS TO ASCII
001476  000401                  4$:     BR      5$
001500  000405                          BR      6$
001502  104007                  5$:     TYPEM                   ;TYPE PROGRAM TITLE FIRST TIME RUN
001504  016505                          STARTM
001506  012737  000240  001476          MOV     #NOP,4$
```

```
                                        ;**********
                                        ;THIS NEXT PART CHECKS THE PRESENCE OF DL11-A OR DL11-C
                                        ;STARTING AT 776500. A MESSAGE WILL BE PRINTED INDICATING THE NUMBER
                                        ;PRESENT. THE PRINTER DIAGNOSTIC WILL ADDRESS EACH OF
                                        ;THE MULTIPLE DL11S IN THE SYSTEM IF SWITCH 13 IS DOWN (0)
                                        ;**********

001514  012737  001574  000004  6$:     MOV     #END2A,MACHER     ;INITIALIZE TIME OUT TRAP
001522  013700  001106                  MOV     DLADR,R0          ;ADDRESS OF FIRST DL11 TO R0
001526  012701  000020                  MOV     #20,R1            ;SET DL CHECK COUNT
001532  005037  001154                  CLR     DLCNT             ;INITIALIZE DLCNT
001536  005710                  END3:   TST     (R0)              ;IS DL PRESENT?
001540  012737  001606  000004          MOV     #END2,MACHER      ;YES, RESET TIME OUT TRAP
001546  010037  001130                  MOV     R0,FSTDL          ;STORE ADDRESS OF FIRST DL11
001552  000401                          BR      2$                ;CONTINUE
001554  005710                  1$:     TST     (R0)              ;IS DL11 PRESENT
001556  062700  000010          2$:     ADD     #10,R0            ;POINTER AND DL11 ADDRESS
001562  005237  001154                  INC     DLCNT             ;INCREMENT COUNT OF DL11'S
001566  005301                          DEC     R1                ;DECREMENT DL CHEC COUNT, DONE?
001570  001407                          BEQ     END4              ;BRANCH IF DONE
001572  000770                          BR      1$                ;CHECK PRESENCE OF NEXT DL11
001574  005301                  END2A:  DEC     R1                ;DONE DL CHECK?
001576  001404                          BEQ     END4              ;YES, EXIT
001600  062700  000010                  ADD     #10,R0            ;NO, CHECK NEXT DL
001604  000754                          BR      END3              ;CONTINUE
001606  022626                  END2:   POPSP2                    ;DL11 NOT PRESENT
001610  013701  001154          END4:   MOV     DLCNT,R1          ;GET # DL11'S
001614  012700  016654                  MOV     #DL11S1,R0        ;ADR OF ASCII CHAR STORAGE
001620  012702  000002                  MOV     #2,R2             ;# OF ASCII CHARS
001624  104023                          BTOASC                    ;CONVERT NUMBER
001626  104007                          TYPEM                     ;TYPE MESSAGE
001630  016641                          DL11S

                                        ;**********
                                        ;EXECUTE THE STRING OF CONSOLE TERMINAL I/O TESTS
                                        ;THEN EITHER HALT AT LOCATION 1022 OR CONTINUE WITH
                                        ;PRINTER TESTS AS A FUNCTION OF SR BIT 8.
                                        ;**********

001632  005037  001134                  CLR     RTNNO             ;SET ROUTINE NO = 0
001636  005037  001152                  CLR     LEVEL             ;ST LEVEL = 0
001642  023727  005522  177777          CMP     ATOX,#177777      ;SEE IF I/O IS TO BE SKIPED
001650  001515                          BEQ     SKIP
001652  012737  005520  001136          MOV     #AT0,NXTST        ;ADDRESS OF FIRST I/O TEST
001660  104024                          FORWD                     ;SET UP TEST PARAMETERS
001662  000177  177306                  JMP     @CURTST           ;GO TO I/O TEST ROUTINE
```

```
                                            ;**********
                                            ;CHAINN--    THIS PORTION IS THE COMMON RETURN
                                            ;            FOR ALL THREE CLASSES OF TESTS.
                                            ;
                                            ;        1--IF AN ERROR OCCURED DURING AN I/O TEST THE
                                            ;           OPERATOR CAN CAUSE THAT TEST TO BE LOOPED
                                            ;           WITHOUT ANY FURTHER ERROR HALTS BY
                                            ;           SETTING THE "SCOPE" BIT (#14) ON THE SR=1.
                                            ;           RESETTING SR BIT 14 TO 0 WILL ALLOW THE
                                            ;           ERROR HALT TO OCCUR AGAIN  IF IT STILL EXISTS.
                                            ;
                                            ;        2--IF THE OPERATOR IS IN THE MAINTENANCE
                                            ;           MODE (BIT 8 SET = 1 AT START UP TIME), THE
                                            ;           SELECTED PROGRAM WILL LOOP CONTINOUSLY
                                            ;           IF SR BIT 11 IS SET=1. IF BIT 11 IS = 0
                                            ;           THEN THE PROGRAM WILL BE ADVANCED TO
                                            ;           THE NEXT TEST IN IT'S CLASS IF BIT 8=0.
                                            ;           AS LONG AS BIT 11 AND
                                            ;           BIT 8 ARE 0, THE CLASS OF TESTS SELECTED
                                            ;           WILL BE CONTINOUSLY SEQUENCED THROUGH.
                                            ;           IF BIT 11 IS 0 AND BIT 8=1, THEN THE CPU
                                            ;           WILL HALT AT LOCATION 1022 AND WAIT FOR THE
                                            ;           NEXT TEST NUMBER TO BE SET IN THE
                                            ;           SWITCH REGISTER.
                                            ;**********

001666  032737  000001  001132  CHAINN: BIT     #1,CNTLSW       ;CKECK IF TERMINAL CONTROL
001674  001401                          BEQ     1$              ;BRANCH IF NOT
001676  104011                          TTYCTL                  ;GO TO TERMINAL CONTROL
001700  005737  001142          1$:     TST     PRGID           ;TEST ERROR BIT IN PRGID
001704  100016                          BPL     3$              ;BRANCH IF ERROR BIT NOT SET
001706  032777  040000  177276          BIT     #SCOPSW,@SR     ;ERR, CHECK IF SCOPE OPTION ON
001714  001407                          BEQ     2$              ;BRANCH IF NO SCOPING
001716  022737  177777  001140          CMP     #-1,SCOPTR      ;YES, CHECK IF OK TO SCOPE THIS TEST
001724  001403                          BEQ     2$              ;BRANCH IF NOT OK
001726  017716  177206                  MOV     @SCOPTR,@SP     ;PUT ADDR OF SCOPE ENTRY INTO STACK
001732  000002                          RTI                     ;GO TO SCOPE ENTRY IN TEST
001734  042737  100000  001142  2$:     BIC     #BIT15,PRGID    ;CLEAR ERROR IND. IN PRGID
001742  005737  001152          3$:     TST     LEVEL           ;CHECK LEVEL
001746  001405                          BEQ     4$              ;BRANCH IF LEVEL=0
001750  032777  004000  177234          BIT     #NITRSW,@SR     ;TEST LOOP SWITCH ON (=1)
001756  001405                          BEQ     5$              ;BRANCH IF NO LOOP TEST
001760  000002                          RTI                     ;GO BACK TO TEST
001762  005337  001156          4$:     DEC     ICTR            ;DECREMENT TEST ITERATION COUNT
001766  001407                          BEQ     6$              ;BRANCH IF COUNT=0
001770  000002                          RTI                     ;NOT ZERO, REPEAT TEST
001772  032777  000400  177212  5$:     BIT     #BIT8,@SR       ;TEST IF SEQUENCE TEST (BIT8)
002000  001402                          BEQ     6$              ;BRANCH TO NEXT TEST IF BIT8=0
002002  000137  002154                  JMP     WAITF           ;GO WAIT FOR MORE INPUT
002006  022626                  6$:     POPSP2                  ;POP 2 OFF STACK
002010  000005                  CHAINY: RESET                   ;RESET
002012  005777  177174                  TST     @SR             ;CHECK SR
002016  100003                          BPL     1$              ;BRANCH IF NO HALT WANTED
002020  113700  001134                  MOVB    RTNNO,R0        ;CURRENT TEST NUMBER TO R0
```

```
002024  000000                                 HALT                        ;HALT
002026  005737  001152                 1$:     TST     LEVEL               ;TEST THE CURRENT LEVEL
002032  001420                                 BEQ     3$                  ;BRANCH IF 0
002034  012737  000006  000004                 MOV     #6,MACHER           ;CLEAN UP
002042  012706  001100                         MOV     #SPBOT,SP           ;SET UP STACK POINTER
002046  104024                                 FORWD                       ;SET UP VALUES FOR NEXT TEST
002050  022737  177777  001136                 CMP     #-1,NXTST           ;END OF I/O TESTS (=-1)
002056  001004                                 BNE     2$                  ;BRANCH IF NOT END
002060  012737  005520  001136                 MOV     #AT0,NXTST          ;RESET NXTST TO FIRST I/O TEST
002066  104024                                 FORWD                       ;SET UP VALUES FOR NEXT TEST
002070  000177  177100                 2$:     JMP     @CURTST             ;GO TO TEST
002074  022737  177777  001136         3$:     CMP     #-1,NXTST           ;END OF I/O TESTS (=-1)
002102  001012                                 BNE     NEXT                ;BRANCH IF NOT
002104  032777  000400  177100         SKIP:   BIT     #BIT8,@SR           ;TEST IF WANT TEST SELECTION RIGHT AWAY
002112  001016                                 BNE     NEXT1               ;BRANCH IF NOT
002114  052737  000200  001142                 BIS     #BIT7,PRGID         ;BYPASS SCOPING
002122  012737  007242  001136                 MOV     #PT0,NXTST          ;PROD TESTING, GO TO PRINTER TESTS
002130  012737  000006  000004         NEXT:   MOV     #6,MACHER           ;CLEAN UP
002136  012706  001100                         MOV     #SPBOT,SP           ;SET UP STACK POINTER
002142  104024                                 FORWD                       ;SET UP NEXT TEST PARAMETERS
002144  000177  177024                         JMP     @CURTST             ;GO TO ROUTINE
002150  005237  001152                 NEXT1:  INC     LEVEL

                                       ;**********
                                       ;WAIT FOR FURTHER INSTRUCTIONS:
                                       ;               -LOAD PROGRAM NUMBER INTO BITS 0-5 OF THE SR
                                       ;               -SET SR BIT 11=1 TO LOOP ON SELECTED TEST
                                       ;               -SET SR BIT 11=0 AND BIT 8=0 TO LOOP THROUGH
                                       ;                SEQUENCE OF SELECTED TESTS.
                                       ;               -SET SR BIT 11=0 AND BIT 8=1 TO HALT AGAIN AFTER
                                       ;                EXECUTING TEST ONCE
                                       ;**********

002154  104006                         WAITF:  CHALT                       ;OR TTYCTL IF START WAS AT 206
002156  012737  000006  000004                 MOV     #6,MACHER           ;CLEAN UP
002164  012706  001100                         MOV     #SPBOT,SP           ;SET UP STACK POINTER
002170  017700  177016                         MOV     @SR,R0              ;GET CURRENT SW REG
002174  042700  177700                         BIC     #177700,R0
002200  020027  000037                         CMP     R0,#37              ;TEST IF PROG NO. IS I/O TEST
002204  101403                                 BLOS    1$                  ;BRANCH IF EQ OR LT 37. AN ECHO OR PRINTER
002206  005037  001142                         CLR     PRGID               ;I/O TEST, CLEAR PRGID
002212  000403                                 BR      2$
002214  052737  000200  001142         1$:     BIS     #BIT7,PRGID         ;BYPASS SCOPING
002222  000241                         2$:     CLC                         ;CLEAR C BIT
002224  006100                                 ROL     R0                  ;GET PROGRAM ADDRESS OUT OF
002226  016037  002776  001136                 MOV     PRGTAB(R0),NXTST    ;PROGRAM ADDRESS TABLE
002234  023727  001136  002154                 CMP     NXTST,#WAITF        ;TEST IF LEGAL TEST NO.
002242  001744                                 BEQ     WAITF               ;BRANCH IF ILLEGAL
002244  104024                                 FORWD                       ;SET UP TEST PARAMETERS
002246  000177  176722                         JMP     @CURTST             ;GO TO TEST
```

```
                                ;**********
                                ;TTY1-- THIS SECTION IS USED WHEN THE DIAGNOSTIC IS UNDER CONTROL
                                ;         THE CONSOLE TERMINAL. IT IS EFFECTIVE ONLY WHEN THE DIAGNOSTIC
                                ;       STARTING ADDRESS IS 210 AND SR BIT 8 WAS SET AT START TIME.
                                ;       THE RESPONSE TO THE MESSAGE "SELECT TEST NO." MUST BE THE 2
                                ;       DIGIT OCTAL TEST NUMBER FOLLOWED BY ;
                                ;               "L" TO LOOP ON TEST
                                ;               "S" TO LOOP ON SEQUENCE
                                ;               "." TO EXECUTE TEST ONCE
                                ;       ALL SPACES WILL BE IGNORED. AN ILLEGAL INPUT WILL BE FLAGED BY A "?"
                                ;       AND THE RETYPING OF THE ABOVE MESSAGE.
                                ;**********

002252  022626                  TTY1:   POPSP2                          ;POP 2 FROM STACK
002254  105777  176630                  TSTB    @TKS                    ;TEST IF ANY INPUT
002260  100013                          BPL     1$                      ;BRANCH IF NOT
002262  017705  176624                  MOV     @TKB,R5                 ;GET CHAR
002266  042705  177600                  BIC     #177600,R5              ;MASK BITS
002272  020527  000177                  CMP     R5,#177                 ;CHECK IF RUBOUT
002276  001004                          BNE     1$                      ;BRANCH IF NOT
002300  042737  004400  001132          BIC     #4400,CNTLSW            ;CLEAR LOOP BITS
002306  000413                          BR      TTY1B
002310  032737  004000  001132  1$:     BIT     #NITRSW,CNTLSW          ;CHECK IF LOOP ON TEST
002316  001401                          BEQ     2$                      ;BRANCH IF NO LOOP ON TEST
002320  000002                          RTI                             ;LOOP ON TEST
002322  032737  000400  001132  2$:     BIT     #BIT8,CNTLSW            ;TEST IF LOOP ON SEQUENCE
002330  001402                          BEQ     TTY1B                   ;BRANCH IF NO LOOP ON SEQUENCE
002332  000137  002010                  JMP     CHAINY                  ;CHAIN TO NEXT TEST
002336  012737  177777  001206  TTY1B:  MOV     #-1,INCHK               ;STOP INPUT CHECKING
002344  012700  000036                  MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
002350  104010                          DELAY
002352  104007                          TYPEM
002354  017127                          MESG3                           ;TYPE MESSAGE
002356  005037  001206                  CLR     INCHK                   ;ALLOW INPUT CHECKING AGAIN
002362  104020                  1$:     READ                            ;WAIT FOR INPUT
002364  023727  001176  000040          CMP     TEMPCH,#40              ;TEST IF CHAR IS A SPACE
002372  001773                          BEQ     1$                      ;BRANCH IF YES
002374  012700  000036                  MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
002400  104010                          DELAY
002402  104017                          PRNT                            ;READY?
002404  117777  176502  176504          MOVB    @TKB,@TPB               ;ECHO CHAR
002412  004737  002734                  JSR     PC,TESTC                ;CHECK IF CHAR IS OK
002416  000541                          BR      8$                      ;NO, ERROR
002420  010005                          MOV     R0,R5                   ;OK, PUT CHAR INTO R5
002422  006305                          ASL     R5                      ;SHIFT INTO POSITION 5-3
002424  006305                          ASL     R5
002426  006305                          ASL     R5
002430  104020                  2$:     READ                            ;WAIT FOR NEXT CHAR
002432  023727  001176  000040          CMP     TEMPCH,#40              ;CHECK IF A SPACE
002440  001773                          BEQ     2$                      ;BRANCH IF SPACE
002442  012700  000036                  MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
002446  104010                          DELAY
002450  104017                          PRNT                            ;READY?
002452  117777  176434  176436          MOVB    @TKB,@TPB               ;ECHO CHAR
```

```
002460  004737  002734                   JSR    PC,TESTC         ;CHECK IF CHAR IS OK
002464  000516                           BR     8$               ;ERROR IN CHAR
002466  060005                           ADD    R0,R5            ;OK,R5 NOW = OCTAL TEST NO.
002470  104020                   3$:     READ                    ;WAIT FOR TERMINATION CHARACTER
002472  023727  001176  000040           CMP    TEMPCH,#40       ;CHECK IF SPACE
002500  001773                           BEQ    3$               ;BRANCH IF SPACE
002502  012700  000036                   MOV    #30.,R0          ;DELAY FOR HALF DUPLEX
002506  104010                           DELAY
002510  104017                           PRNT                    ;READY?
002512  117777  176374  176376           MOVB   @TKB,@TPB        ;ECHO CHAR
002520  012737  004001  001132           MOV    #4001,CNTLSW     ;SET BITS 11 & 0
002526  023727  001176  000114           CMP    TEMPCH,#114      ;NO, IS IT AN "L" ?
002534  001427                           BEQ    5$               ;BRANCH IF YES
002536  023727  001176  000154           CMP    TEMPCH,#154      ;CHECK LOWER CASE
002544  001423                           BEQ    5$
002546  023727  001176  000123           CMP    TEMPCH,#123      ;NO, IS IT AN "S"
002554  001414                           BEQ    4$               ;BRANCH IF YES
002556  023727  001176  000163           CMP    TEMPCH,#163      ;CHECK LOWER CASE
002564  001410                           BEQ    4$
002566  023727  001176  000056           CMP    TEMPCH,#56       ;NO, IS IT A "." ?
002574  001052                           BNE    8$               ;NO, ERROR
002576  012737  000001  001132           MOV    #1,CNTLSW        ;YES SET ONLY BIT 0 IN CONTROL WD
002604  000403                           BR     5$
002606  012737  000401  001132   4$:     MOV    #401,CNTLSW      ;SET BITS 8 & 0
002614  012737  000006  000004   5$:     MOV    #6,MACHER        ;CLEAN UP
002622  012706  001100                   MOV    #SPBOT,SP        ;INIT SP
002626  020527  000040                   CMP    R5,#40           ;IS THIS AN I/O TEST
002632  103033                           BHIS   8$               ;BRANCH IF YES
002634  020527  000030                   CMP    R5,#30           ;IS THIS AN OPTION TEST?
002640  103007                           BHIS   6$               ;SKIP IF YES
002642  020527  000020                   CMP    R5,#20           ;IS THIS AN ECHO TEST
002646  103404                           BLO    6$               ;BRANCH IF NOT
002650  012737  000001  001132           MOV    #1,CNTLSW        ;FORCE ECHO TEST TO A SINGLE RUN
002656  000402                           BR     7$               ;LEAVE THIS TERMINAL AS CONSOLE
002660  004737  004122           6$:     JSR    PC,CONIT         ;RESET CONSOLE TERMINAL ADDRESS
002664  052737  000200  001142   7$:     BIS    #BIT7,PRGID      ;BYPASS SCOPING
002672  000241                           CLC                     ;CLEAR C BIT
002674  006105                           ROL    R5
002676  016537  002776  001136           MOV    PRGTAB(R5),NXTST;ADDR OF TEST TO NXTST
002704  023727  001136  002154           CMP    NXTST,#WAITF     ;CHECK IF TEST EXISTS
002712  001403                           BEQ    8$               ;BRANCH IF NOT
002714  104024                           FORWD                   ;SET UP TEST PARAMETERS
002716  000177  176252                   JMP    @CURTST          ;GO TO TEST
002722  104017                   8$:     PRNT                    ;CHECK IF PRINTER IS READY
002724  112777  000077  176164           MOVB   #77,@TPB         ;SEND A "?"
002732  000601                           BR     TTY1B            ;TRY AGAIN
```

```
                                            ;TESTC--CHECKS THAT THE INPUTTED CHARACTER IS BETWEEN A 0 AND A 7 INCLUSIVE

002734  023727  001176  000060  TESTC:  CMP     TEMPCH,#60      ;CHECK IF NUMERIC AND EQ OR GT 0
002742  103001                          BHIS    1$              ;BRANCH IF OK
002744  000207                          RTS     PC              ;ERROR RETURN
002746  023727  001176  000067  1$:     CMP     TEMPCH,#67      ;CHECK IF EQ OR LT 7
002754  101401                          BLOS    2$              ;BRANCH IF OK
002756  000207                          RTS     PC              ;ERROR RETURN
002760  062716  000002          2$:     ADD     #2,@SP          ;SET UP RETURN ADDRESS
002764  013700  001176                  MOV     TEMPCH,R0       ;GET CHAR
002770  042700  177770                  BIC     #177770,R0      ;SAVE ONLY THE DIGIT
002774  000207                          RTS     PC              ;NORMAL RETURN
```

```
002776 007242		PRGTAB: PT0		;DATA PATH TEST
003000 007316			PT1		;PRINTER CHARACTER TEST
003002 007440			PT2		;NON-PRINTING CHARACTER TEST
003004 010070			PT3		;CARRIAGE RETURN TEST
003006 010210			PT4		;MULTIPLE LINE FEED TEST
003010 010366			PT5		;SINGLE LINE FEED TEST
003012 010572			PT6		;BACKSPACE TEST
003014 010760			PT7		;OVERPRINT TEST
003016 011172			PT10		;PRINTING FREQUENCY SWEEP TEST
003020 011330			PT11		;RIBBON FEED TEST
003022 011362			PT12		;PRINTER BELL TEST
003024 002154			WAITF		;SPARE
003026 002154			WAITF		;SPARE
003030 002154			WAITF		;SPARE
003032 002154			WAITF		;SPARE
003034 011452			PT17		;LIFE TEST
003036 011606			E020		;CHARACTER ECHO TEST
003040 011656			E021		;LINE ECHO TEST, FAST RATE
003042 011714			E022		;LINE ECHO TEST, SLOW RATE
003044 012166			E023		;CHARACTER/CODE ECHO TEST
003046 012510			E024		;SELECTIVE PATTERN ECHO TEST
003050 013256			E025		;BELL ECHO TEST
003052 002154			WAITF		;SPARE
003054 002154			WAITF		;SPARE
003056 013366			TEST30		;SECONDARY CHARACTER SET OPTION
003060 013550			TEST31		;SELECTIVE ADDRESS OPTION
003062 014424			TEST32		;AUTO ANSWER BACK OPTION
003064 014702			TEST33		;TOP OF FORM OPTION
003066 015466			TEST34		;HORIZONTAL TAB OPTION
003070 016030			TEST35		;VERTICAL TAB OPTION
003072 002154			WAITF		;SPARE
003074 002154			WAITF		;SPARE
003076 005520			AT0		;I/O TEST NO. 40
003100 005552			AT1		;I/O TEST NO. 41
003102 005604			AT2		;I/O TEST NO. 42
003104 005636			AT3		;I/O TEST NO. 43
003106 005670			AT4		;I/O TEST NO. 44
003110 005760			AT5		;I/O TEST NO. 45
003112 006030			AT6		;I/O TEST NO. 46
003114 006120			AT7		;I/O TEST NO. 47
003116 006170			AT10		;I/O TEST NO. 50
003120 006216			AT11		;I/O TEST NO. 51
003122 006256			AT12		;I/O TEST NO. 52
003124 006330			AT13		;I/O TEST NO. 53
003126 006410			AT14		;I/O TEST NO. 54
003130 006474			AT15		;I/O TEST NO. 55
003132 006574			AT16		;I/O TEST NO. 56
003134 006632			AT17		;I/O TEST NO. 57
003136 006672			AT20		;I/O TEST NO. 60
003140 006754			AT21		;I/O TEST NO. 61
003142 007044			AT22		;I/O TEST NO. 62
003144 007142			AT23		;I/O TEST NO. 63
003146 002154			WAITF		;SPARE
003150 002154			WAITF		;SPARE
```

```
003152  002154                      WAITF                         ;SPARE
003154  002154                      WAITF                         ;SPARE
003156  002154                      WAITF                         ;SPARE
003160  002154                      WAITF                         ;SPARE
003162  002154                      WAITF                         ;SPARE
003164  002154                      WAITF                         ;SPARE
003166  002154                      WAITF                         ;SPARE
003170  002154                      WAITF                         ;SPARE
003172  002154                      WAITF                         ;SPARE
003174  002154                      WAITF                         ;SPARE

                           ;**********
                           ;EMTINT AND TRPINT--SERVICE ROUTINES FOR TRAPS THROUGH
                           ;                  LOC. 30 AND 34.
                           ;**********

003176  011646             EMTINT:  MOV     @SP,-(SP)             ;GET SAVED PC
003200  162716  000002              SUB     #2,@SP
003204  017616  000000              MOV     @(SP),@SP
003210  121627  000025              CMPB    @SP,#25               ;CHECK THAT CALL IS WITHIN LIMITS
003214  101402                      BLOS    2$                    ;BRANCH IF OK
003216  000000             1$:      HALT
003220  000776                      BR      1$
003222  006116             2$:      ROL     @SP                   ;EMT ARGUMENT *2
003224  042716  177001              BIC     #177001,@SP           ;REMOVE 7 MSB
003230  062716  003252              ADD     #EMTTAB,@SP           ;FORM EMT RTN ADDRESS
003234  017616  000000              MOV     @(SP),@SP
003240  005046                      CLR     -(SP)                 ;CLEAR PSW
003242  012746  003250              MOV     #3$,-(SP)
003246  000002                      RTI
003250  000136             3$:      JMP     @(SP)+                ;GO TO EMT ROUTINE

003252  003326             EMTTAB:  TYP                           ;MESSAGE OUTPUT ROUTINE
003254  003532                      ERR                           ;I/O TEST ERROR ROUTINE
003256  003560                      EHLT                          ;UNCONDITIONAL HALT
003260  003570                      STLSRV                        ;KEYBOARD VECTOR/PRIORITY SETUP
003262  003620                      STLSPV                        ;PRINTER VECTOR/PRIORITY SETUP
003264  001666                      CHAINN                        ;COMMON TEST EXIT
003266  001214                      CHLT                          ;SR BIT 15 HALT
003270  003404                      TYPM                          ;MESSAGE OUTPUT ROUTINE, MULTI DEVICES
003272  003650                      DLY                           ;DELAY ROUTINE
003274  002252                      TTY1                          ;CONSOLE TERMINAL CONTROL
003276  003434                      $CRLF                         ;CARRIAGE RETURN-LINE FEED TO ALL DL11'S
003300  003362                      $SCRLF                        ;CARRIAGE RETURN-LINE FEED TO CONSOLE
003302  003436                      $LF                           ;LINE FEED ONLY (TO ALL)
003304  004536                      $PRTC                         ;PRINT CHAR
003306  003456                      $PRHDR                        ;PRINT TEST HEADER
003310  004526                      $PRNT                         ;PRINTER READY
003312  004324                      $READ                         ;READ CHAR
003314  004052                      $AREAD                        ;I/O TEST READ ROUTINE
003316  003446                      $CR                           ;CARRIAGE RETURN ONLY (TO ALL)
003320  004220                      $BTASC                        ;BINARY TO ASCII CONVERSION
003322  003774                      $FORWD                        ;FORWARD ROUTINE ( BETWEEN TESTS )
003324  004416                      $READC                        ;READ CONSOLE KYBD ONLY
```

```
                                          .SBTTL  COMMON ROUTINES USED BY LA36 TESTS

                                   ;**********
                                   ;
                                   ;THIS SECTION CONTAINS MOST ROUTINES CALLED BY
                                   ;THE VARIOUS TESTS EITHER BY TRAPPING THROUGH LOCATIONS
                                   ;30 AND 34 OR BY SUBROUTINE CALLS (JSR PC,***)
                                   ;
                                   ;**********
                                   ;
                                   ;TYPE-- A COMMON ROUTINE USED TO TYPE MESSAGES ON THE
                                   ;       CONSOLE TERMINAL ONLY. THE NULL CHARACTER TERMINATES
                                   ;       THE MESSAGE. CALLED THROUGH AN EMT TRAP.
                                   ;       CALLING SEQUENCE
                                   ;       TYPE
                                   ;       MESG    ;ADDRESS OF MESSAGE
                                   ;
                                   ;**********

003326  011601                     TYP:    MOV     (SP),R1         ;GET POINTER TO ADDR. OF MESG.
003330  062716  000002                     ADD     #2,@SP
003334  011101                             MOV     (R1),R1         ;ADDR. OF MESG TO R1
003336  112100                     1$:     MOVB    (R1)+,R0        ;GET CHAR
003340  100402                             BMI     2$              ;BRANCH IF WANT AUTO CR-LF
003342  001003                             BNE     3$              ;PRINT CHAR IF NOT NULL
003344  000002                             RTI                     ;EXIT IF NULL CHAR
003346  104013                     2$:     SCRLF                   ;YES, SEND CR-LF
003350  000772                             BR      1$              ;GET NEXT CHAR
003352  104017                     3$:     PRNT                    ;PRINTER READY?
003354  110077  175536                     MOVB    R0,@TPB         ;LOAD PRINTER BUFFER WITH CHAR
003360  000766                             BR      1$              ;GO GET NEXT CHAR

003362  104017                     $SCRLF: PRNT                    ;PRINTER READY?
003364  112777  000015  175524             MOVB    #15,@TPB        ;SEND CR
003372  104017                             PRNT                    ;PRINTER READY?
003374  112777  000012  175514             MOVB    #12,@TPB        ;SEND LF
003402  000002                             RTI                     ;RETURN TO CALLER
```

```
                                ;XXXXXXXXXX
                                ;
                                ;TYPM---MULTI TYPE-A COMMON ROUTINE TO OUTPUT
                                ;       A MESSAGE ON ALL DL11S IF THE MULTI TEST
                                ;       SWITCH (BIT 13) IS SET. THIS ROUTINE IS USED BY
                                ;       THE PRINTER TESTS TO TYPE HEADINGS.  IF A UNIT
                                ;       IS NOT READY, THE CHARACTER WILL NOT BE TYPED.
                                ;
                                ;XXXXXXXXXX

003404  011601                  TYPM:   MOV     (SP),R1         ;GET POINTER TO ADDR OF MESG
003406  062716  000002                  ADD     #2,@SP
003412  011101                          MOV     (R1),R1         ;ADDR OF MESG TO R1
003414  112100                  1$:     MOVB    (R1)+,R0        ;GET CHAR
003416  100402                          BMI     2$              ;BRANCH IF WANT AUTO CR-LF
003420  001003                          BNE     3$              ;CONTINUE IF NOT NULL
003422  000002                          RTI                     ;RETURN
003424  104012                  2$:     CRLF                    ;YES, SEND CR-LF
003426  000772                          BR      1$              ;NEXT CHAR
003430  104015                  3$:     PRINTC                  ;PRINT CHAR
003432  000770                          BR      1$              ;GO GET NEXT CHAR.

003434  104022                  $CRLF:  CR                      ;SEND CR
003436  012700  000012          $LF:    MOV     #12,R0          ;SET LF CHAR
003442  104015                          PRINTC                  ;SEND IT
003444  000002                          RTI                     ;RETURN TO CALLER

003446  012700  000015          $CR:    MOV     #15,R0          ;SET CR CHAR
003452  104015                          PRINTC                  ;SEND IT
003454  000002                          RTI                     ;RETURN

                                ;**********
                                ;
                                ;ROUTINE TO PRINT TEST HEADER
                                ;
                                ;**********

003456  104007                  $PRHDR: TYPEM                   ;PRINT MESSAGE
003460  016703                          HDRMSG
003462  013700  001134                  MOV     RTNNO,R0        ;GET TEST NUMBER
003466  006200                          ASR     R0              ;GET FIRST DIGIT
003470  006200                          ASR     R0
003472  006200                          ASR     R0
003474  042700  177770                  BIC     #177770,R0      ;MASK FIRST DIGIT
003500  062700  000060                  ADD     #60,R0          ;MAKE ASCII
003504  104015                          PRINTC                  ;PRINT DIGIT
003506  013700  001134                  MOV     RTNNO,R0        ;GET TEST NUMBER AGAIN
003512  042700  177770                  BIC     #177770,R0      ;MASK LAST DIGIT
003516  062700  000060                  ADD     #60,R0          ;MAKE ASCII
003522  104015                          PRINTC                  ;PRINT DIGIT
003524  104012                          CRLF                    ;CR-LF
003526  104014                          LF                      ;BLANK LINE
003530  000002                          RTI                     ;RETURN
```

```
                                     ;**********
                                     ;
                                     ;ERRA-- COMMON ERROR RETURN FROM I/O TESTS. HALTS
                                     ;       WITH ADDRESS OF ERROR IN R0. TO CONTINUE
                                     ;       ON SAME TEST BUT NOT HALTING ON ERROR,
                                     ;       SET THE SCOPE BIT (14) = 1 AND PRESS CONTINUE
                                     ;
                                     ;**********

003532 032777 040000 175452  ERR:    BIT     #SCOPSW,@SR       ;CHECK SCOPE SWITCH
003540 001404                        BEQ     1$                ;BRANCH IF NO SCOPE
003542 005737 001142                 TST     PRGID             ;SCOPING WANTED, FIRST ERROR?
003546 100001                        BPL     1$                ;BRANCH AND HALT ON FIRST ERROR
003550 000002                        RTI                       ;SCOPE EXIT
003552 052737 100000 001142  1$:     BIS     #BIT15,PRGID      ;SET ERROR INDICATOR
003560 011600                EHLT:   MOV     @SP,R0
003562 005740                        TST     -(R0)             ;ADDRESS OF CALL INTO R0
003564 000000                        HALT
003566 000002                ERRHLT: RTI                       ;RETURN TO TEST FOLLOWING CALL

                                     ;**********
                                     ;
                                     ;STLSRV--- THIS ROUTINE SETS UP KEYBOARD INTERRUPT
                                     ;            VECTOR AND PRIOITY. CALLING SEQUENCE
                                     ;
                                     ;       STRDRV
                                     ;       AT20C    ;LOCATION OF NEW INTERRUPT VECTOR
                                     ;
                                     ;**********

003570 017637 000000 003610  STLSRV: MOV     @(SP),STPRA+2    ;SET RETURN ADR AND VECTOR
003576 062716 000002                 ADD     #2,@SP
003602 013701 001120                 MOV     TKVTR,R1
003606 012721 000000         STPRA:  MOV     #0,(R1)+
003612 013721 001122                 MOV     TKLVL,(R1)+
003616 000002                        RTI

                                     ;**********
                                     ;
                                     ;STLSPV-- THIS ROUTINE SETS UP PRINTER INTERRUPT
                                     ;
                                     ;       VECTOR AND PRIORITY CALLING SEQUENCE
                                     ;       STPCHV
                                     ;       AT35E    ;LOCATION OF NEW INTERRUPT VECTOR
                                     ;
                                     ;**********

003620 017637 000000 003640  STLSPV: MOV     @(SP),STPPA+2    ;SET RETURN ADR AND VECTOR
003626 062716 000002                 ADD     #2,@SP
003632 013701 001124                 MOV     TPVTR,R1
003636 012721 000000         STPPA:  MOV     #0,(R1)+
003642 013721 001126                 MOV     TPLVL,(R1)+
003646 000002                        RTI                      ;RETURN TO CALLER
```

```
                                        ;**********
                                        ;
                                        ;DELAY--A COMMON ROUTINE TO DELAY PROCESSING
                                        ;       A GIVEN NUMBER OF MSEC.
                                        ;       CALLING SEQUENCE;
                                        ;       MOV    #5,R0    ;R0 CONTAINS THE NUMBER OF MSEC DELAY DESIRED
                                        ;       DELAY
                                        ;
                                        ;       THE DELAY IS EFFECTED BY THE EXECUTION OF THE LOOP;
                                        ;          1$:  DEC  R1
                                        ;               BNE  1$
                                        ;
                                        ;       SINCE THE EXECUTION TIMES OF THE PDP11 LINE DOES VARY FROM
                                        ;       MACHINE TO MACHINE, THE VALUE AT SYMBOLIC LOCATION
                                        ;       "TIMER" MUST BE CHANGED TO THE APPROPRIATE VALUE AS SHOWN BELOW
                                        ;       BEFORE STARTING THE DIAGNOSTIC. "TIMER" IS INITIALIZED
                                        ;       FOR AN 11/05,11/10(=251).
                                        ;
                                        ;MACHINE        05&10    35&40   15&20                 11/45 & 11/70
                                        ;                                          BIPOLAR          MOS      CORE
                                        ;
                                        ;LOOP: DEC R1   3.4      .99     2.3          .30            .51      .90
                                        ;      BNE LOOP 2.5      1.76    2.6          .60            .98     1.13
                                        ;  TIME=        5.9USEC  2.75    4.9         .90USEC        1.49USEC            2.03USEC
                                        ; SET TIMER     251      554     314         2127           1237      755
                                        ;
                                        ;XXXXXXXXXX

 003650  010146                         DLY:    MOV     R1,-(SP)        ;SAVE R1
 003652  013701  001170                 1$:     MOV     TIMER,R1        ;MOV 1 MSEC LOOP CNT TO R1
 003656  005301                         2$:     DEC     R1              ;DECREMENT COUNT
 003660  001376                                 BNE     2$              ;BRANCH IF NOT ZERO
 003662  005300                                 DEC     R0              ;DEC NO. OF MSEC DELAY
 003664  001372                                 BNE     1$              ;DELAY AGAIN IF NOT ZERO
 003666  012601                                 MOV     (SP)+,R1        ;ALL DONE RESTORE R1
 003670  000002                                 RTI
```

```
                                        ;**********
                                        ;
                                        ;PFAIL--POWER FAIL ROUTINE
                                        ;       SAVE ALL REGISTERS AND SET RESTART ADDRESS
                                        ;       INTO LOCATION 24
                                        ;
                                        ;RESTART--POWER FAIL RECOVERY
                                        ;        RESTORE ALL REGISTERS AND GO TO START
                                        ;
                                        ;**********

003672  010046                          PFAIL:  MOV     R0,-(SP)
003674  010146                                  MOV     R1,-(SP)
003676  010246                                  MOV     R2,-(SP)
003700  010346                                  MOV     R3,-(SP)
003702  010446                                  MOV     R4,-(SP)
003704  010546                                  MOV     R5,-(SP)
003706  013746  000024                          MOV     24,-(SP)
003712  010637  003726                          MOV     SP,SAVR6        ;SAVE STACK POSITION
003716  012737  003730  000024                  MOV     #RESTRT,24      ;STORE RESTART ADDRESS
003724  000000                                  HALT
003726  000000                          SAVR6:  .WORD   0
003730  104007                          RESTRT: TYPEM
003732  003764                                  1$
003734  013706  003726                          MOV     SAVR6,SP        ;RESTORE STACK POINTER
003740  012637  000024                          MOV     (SP)+,24        ;RESTORE PFAIL ADDRESS
003744  012605                                  MOV     (SP)+,R5
003746  012604                                  MOV     (SP)+,R4
003750  012603                                  MOV     (SP)+,R3
003752  012602                                  MOV     (SP)+,R2
003754  012601                                  MOV     (SP)+,R1
003756  012600                                  MOV     (SP)+,R0
003760  000137  001304                          JMP     START

003764  050200  053517  051105          1$:     .ASCIZ  <ACRLF>/POWER/<ACRLF>
003772  000200
                                                .EVEN
```

```
                                        ;**********
                                        ;
                                        ;FORWARD--THIS ROUTINE TRANSFERS THE 2 OR 4 ARGUMENTS
                                        ;       FROM THE TEST ROUTINE. THEY ARE;
                                        ;
                                        ;       1- ROUTINE NUMBER
                                        ;       2- ADDRESS OF NEXT TEST
                                        ;       3- ITERATION COUNT (I/O TESTS ONLY)
                                        ;       4- SCOPE ENTRY ADDRESS (I/O TESTS ONLY)
                                        ;
                                        ;**********

003774  013705  001136                  $FORWD: MOV     NXTST,R5        ;ADDR OF NEXT TEST TO R5
004000  012537  001134                          MOV     (R5)+,RTNNO     ;GET NUMBER OF NEXT TEST
004004  012537  001136                          MOV     (R5)+,NXTST     ;GET ADDR OF FOLLOWING TEST
004010  105737  001142                          TSTB    PRGID           ;CHECK IF I/O TEST
004014  100407                                  BMI     FORWDB          ;SKIP THE FETCH OF ITER CNT AND SCOPE
004016  012537  001156                          MOV     (R5)+,ICTR      ;GET ITERATION COUNT
004022  012537  001140                          MOV     (R5)+,SCOPTR    ;GET SCOPE ENTRY POINT
004026  010537  001174                  FORWDA: MOV     R5,CURTST       ;ENTRY POINT TO TEST IN CUR TST
004032  000002                                  RTI                     ;EXIT
004034  012737  177777  001140          FORWDB: MOV     #-1,SCOPTR      ;FORCE NO SCOPE
004042  012737  000001  001156                  MOV     #1,ICTR         ;FORCE INTERATION COUNT OF 1
004050  000766                                  BR      FORWDA
```

```
                                        ;**********
                                        ;
                                        ;AREAD--A ROUTINE WHICH, THROUGH THE FACILITY OF
                                        ;       THE MAINTENANCE BIT, OUTPUTS TO THE
                                        ;       PRINTER BUFFER AND READS THE KEYBOARD
                                        ;       STATUS DONE. IF THE DONE IS NOT SET
                                        ;       WITHIN 200 MSEC, THE CPU WILL HALT WITH
                                        ;       THE LOCATION OF THE ERROR IN RO. PRESS
                                        ;       CONTINUE TO CONTINUE WITH TESTS.
                                        ;
                                        ;**********

004052  012737  000200  001162  $AREAD: MOV     #200,BRCTR      ;SET UP 200 MSEC DELAY
004060  052777  000004  175026          BIS     #4,@TPS         ;SET MAINTENANCE BIT
004066  005077  175024                  CLR     @TPB            ;LOAD PRINTER BUFFER
004072  105777  175012          1$:     TSTB    @TKS            ;CHECK DONE BIT
004076  100410                          BMI     2$              ;BRANCH IF DONE
004100  012700  000001                  MOV     #1,RO           ;ONE TO RO
004104  104010                          DELAY                   ;DELAY 1 MSEC.
004106  005337  001162                  DEC     BRCTR           ;200 MSEC OVER
004112  001367                          BNE     1$              ;BRANCH IF NO
004114  104002                          EHALT
004116  000755                          BR      $AREAD          ;TRY AGAIN
004120  000002                  2$:     RTI                     ;RETURN TO TEST

                                        ;**********
                                        ;
                                        ;CONIT--THIS ROUTINE SETS UP THE DEVICE ADDRESSES
                                        ;       AND INTERRUPT VECTORS FOR THE CONSOLE
                                        ;       TERMINAL.
                                        ;
                                        ;**********

004122  013700  001102          CONIT:  MOV     CONADD,RO       ;CONSOLE KEYBOARD STATUS ADDR TO RO
004126  010037  001110          CONSET: MOV     RO,TKS          ;KEYBOARD STATUS ADDRESS (777560) TO TKS
004132  005720                          TST     (RO)+           ;INCREMENT RO BY TWO
004134  010037  001112                  MOV     RO,TKB          ;KEYBOARD DATA ADDR (777562) TO TKB
004140  005720                          TST     (RO)+           ;INCREMENT RO BY TWO
004142  013737  001114  004214          MOV     TPS,TPSS        ;SAVE TPS OF LAST TERMINAL
004150  010037  001114                  MOV     RO,TPS          ;PRINTER STATUS ADDR(777564) TO TPS
004154  005720                          TST     (RO)+           ;INCREMENT RO BY TWO
004156  013737  001116  004216          MOV     TPB,TPBS        ;SAVE TPB OF LAST TERMINAL
004164  010037  001116                  MOV     RO,TPB          ;PRINTER DATA ADDR (777566) TO TPB
004170  013737  001104  001120          MOV     CONVEC,TKVTR    ;KEYBOARD INTERRUPT VECTOR (60) TO TKVTR
004176  013737  001104  001124          MOV     CONVEC,TPVTR
004204  062737  000004  001124          ADD     #4,TPVTR        ;PRINTER INTERRUPT VECTOR (64) TO TPVTR
004212  000207                          RTS     PC

004214  000000                  TPSS:   .WORD   0               ;LAST TERM STATUS REG ADR
004216  000000                  TPBS:   .WORD   0               ;LAST TERM BUFFER REG ADR
```

```
                                        ;**********
                                        ;
                                        ;BINARY TO ASC11 CONVERSION (1 TO 5 ASC11 CHARACTERS)
                                        ;CALLING SEQUENCE
                                        ;       MOV     ADDRESS OF LOC TO STORE FIRST ASC11 CHAR. INTO R0
                                        ;       MOV     BINARY NUMBER TO BE CONVERTED INTO R1
                                        ;       MOV     NUMBER TO BE CONVERTED AS A POWER OF TEN INTO R2
                                        ;       BTOASC
                                        ;
                                        ;**********

004220  010237  004304          $BTASC: MOV     R2,CNVCTR       ;SAVE TEN POWER
004224  006302                          ASL     R2              ;R2*2
004226  062702  004312                  ADD     #ADTENP,R2      ;CALCULATE ADDRESS OF
                                                                ;STARTING TEN POWER
004232  014237  004310          1$:     MOV     -(R2),TENPWR    ;POWER OF TEN VALUE TO TEN PWR
004236  005037  004306                  CLR     DIGIT           ;CLEAR CURRENT DIGIT
004242  163701  004310          2$:     SUB     TENPWR,R1       ;SUBTRACT TEN POWER FROM BINARY VALUE
004246  103403                          BCS     3$              ;BRANCH IF END
004250  005237  004306                  INC     DIGIT
004254  000772                          BR      2$
004256  063701  004310          3$:     ADD     TENPWR,R1       ;RESTORE SUBTRACTED VALUE
004262  062737  000060  004306          ADD     #60,DIGIT       ;CONVERT (DIGIT) TO ASCII
004270  113720  004306                  MOVB    DIGIT,(R0)+     ;PUT ASCII CHAR INTO USER BUFFER
004274  005337  004304                  DEC     CNVCTR          ;FINISHED ALL CHARS. CALLED FOR
004300  001354                          BNE     1$              ;BRANCH IF NOT FINISHED
004302  000002                          RTI                     ;YES, EXIT
004304  000000                  CNVCTR: .WORD   0               ;CONVERSION CHARACTER COUNT
004306  000000                  DIGIT:  .WORD   0               ;CONVERTED CHARACTER
004310  000000                  TENPWR: .WORD   0               ;CURRENT TEN POWER
004312  000001  000012  000144  ADTENP: .WORD   1.,10.,100.,1000.,10000.
004320  001750  023420
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;READ-- A COMMON ROUTINE WHICH CHECKS THE KEYBOARD
                                        ;       DONE FLAG & SETS A FLAG INDICATING CHAR PARITY
                                        ;
                                        ;XXXXXXXXXX

004324  004737  004122                  $READ:  JSR     PC,CONIT                ;RESET CONSOLE ADR AND VECTORS
004330  005737  001154                          TST     DLCNT                   ;CHECK IF MULTI DL11'S AVAILABLE
004334  001430                                  BEQ     $READC                  ;NONE, WAIT FOR CONSOLE INPUT
004336  013737  001154  001164          1$:     MOV     DLCNT,COUNT3            ;SET DL11 COUNT
004344  013737  001130  001166                  MOV     FSTDL,XCSR              ;ADDRESS OF FIRST DL11 INTO XCSR
004352  105777  174610                  2$:     TSTB    @XCSR                   ;TEST IF ANY INPUT
004356  100005                                  BPL     3$                      ;CONTINUE IF NO INPUT
004360  013700  001166                          MOV     XCSR,R0                 ;SET THIS DL11 AS CONSOLE
004364  004737  004126                          JSR     PC,CONSET
004370  000415                                  BR      READ1                   ;READ CHAR AND RETURN
004372  005337  001164                  3$:     DEC     COUNT3                  ;DECREMENT DL11 COUNT
004376  001404                                  BEQ     4$                      ;TEST CONSOLE WHEN DONE DL11'S
004400  062737  000010  001166                  ADD     #10,XCSR                ;NEXT DL11 ADDRESS
004406  000761                                  BR      2$                      ;CONTINUE
004410  105777  174474                  4$:     TSTB    @TKS                    ;CHECK CONSOLE
004414  100350                                  BPL     1$                      ;WAIT, NO INPUT
004416  105777  174466                  $READC: TSTB    @TKS                    ;CHECK KEYBOARD DONE FLAG
004422  100375                                  BPL     $READC                  ;BRANCH IF NOT SET
004424  117737  174462  001176          READ1:  MOVB    @TKB,TEMPCH             ;SAVE CHARACTER
004432  113737  001176  001202                  MOVB    TEMPCH,PCHAR            ;SAVE CODE WITH PARITY BIT
004440  042737  177400  001202                  BIC     #177400,PCHAR           ;MASK UNWANTED BITS
004446  113737  001176  001201                  MOVB    TEMPCH,PARITY+1         ;SAVE CHAR WITH PARITY BIT
004454  042737  177600  001176                  BIC     #177600,TEMPCH          ;MAKE IT 7 BIT ASCII
004462  023727  001176  000004                  CMP     TEMPCH,#4               ;DISREGARD EOT
004470  001715                                  BEQ     $READ
004472  012700  000011                          MOV     #11,R0                  ;SET SHIFT COUNT
004476  042737  000377  001200                  BIC     #377,PARITY             ;CLEAR PARITY FLAG
004504  005300                          1$:     DEC     R0                      ;DECREMENT SHIFT COUNT
004506  001406                                  BEQ     2$                      ;EXIT IF DONE
004510  106337  001201                          ASLB    PARITY+1                ;SHIFT CODE
004514  103373                                  BCC     1$                      ;CONTINUE IF BIT WAS ZERO
004516  105137  001200                          COMB    PARITY                  ;CHANGE PARITY FLAG IF BIT WAS ONE
004522  000770                                  BR      1$                      ;CONTINUE
004524  000002                          2$:     RTI                             ;SET, RET. TO CALLER

                                        ;XXXXXXXXXX
                                        ;
                                        ;PRINT-- A COMMON ROUTINE TO CHECK THE PRINTER READY FLAG
                                        ;
                                        ;XXXXXXXXXX

004526  105777  174362                  $PRNT:  TSTB    @TPS                    ;CHECK PRINTER READY FLAG
004532  100375                                  BPL     $PRNT                   ;BRANCH IF NOT SET
004534  000002                                  RTI                             ;SET, RETURN
```

```
                                        ;**********
                                        ;
                                        ;PRINTC--SENDS A CHARACTER AT A TIME FIRST TO THE
                                        ;       CONSOLE  DL11 THEN TO ALL MULTIPL DL11S IF
                                        ;       SR BIT 13 IS = 0. IF THE REFERENCED PRINTER
                                        ;       READY BIT IS NOT SET, THE CHARACTER WILL NOT BE
                                        ;       SENT TO THAT PRINTER. ENTER WITH CHARACTER IN R0
                                        ;       CALL:   PRINTC
                                        ;
                                        ;********

004536  013737  001102  001210  $PRTC:  MOV     CONADD,TEMP         ;SET CONSOLE ADR
004544  062737  000004  001210          ADD     #4,TEMP
004552  105777  174432          1$:     TSTB    @TEMP
004556  100375                          BPL     1$                  ;WAIT FOR CONSOLE READY
004560  062737  000002  001210          ADD     #2,TEMP             ;SET ADR
004566  010077  174416                  MOV     R0,@TEMP            ;LOAD CONSOLE PRINT BUFFER
004572  032777  020000  174412          BIT     #BIT13,@SR          ;CHECK SW 13
004600  001003                          BNE     2$                  ;SEND ALL TERMS IF SW13 DOWN
004602  005737  001154                  TST     DLCNT               ;CHECK IF MULTIPLE DL11'S
004606  001002                          BNE     3$                  ;CHECK FOR INPUT IF THERE
004610  000137  005256          2$:     JMP     18$
004614  013737  001154  001164  3$:     MOV     DLCNT,COUNT3        ;PUT NO. DL11'S INTO COUNT3
004622  013737  001130  001166          MOV     FSTDL,XCSR          ;ADDR OF FIRST DL INTO XCSR
004630  023727  001134  000032  4$:     CMP     RTNNO,#32           ;CHECK IF TEST 32
004636  001543                          BEQ     13$                 ;DON'T CHECK FOR INPUT IF TEST 32
004640  005737  001206                  TST     INCHK               ;CHECK FOR INPUT?
004644  001140                          BNE     13$
004646  023727  001134  000020          CMP     RTNNO,#20           ;PRINTING TEST?
004654  002004                          BGE     5$                  ;BRANCH IF NOT
004656  022737  104011  002154          CMP     #TTYCTL,WAITF       ;KEYBOARD CONTROL?
004664  001130                          BNE     13$                 ;SKIP INPUT CHECK IF NOT
004666  105777  174274          5$:     TSTB    @XCSR               ;TEST IF ANY INPUT
004672  100125                          BPL     13$                 ;CONTINUE IF NO INPUT
004674  062737  000002  001166          ADD     #2,XCSR             ;SET BUFFER ADDRESS
004702  017737  174260  001176          MOV     @XCSR,TEMPCH
004710  042737  177600  001176          BIC     #177600,TEMPCH
004716  023727  001176  000003          CMP     TEMPCH,#3           ;CHECK IF CONTROL-C
004724  001006                          BNE     6$                  ;CONTINUE IF NOT
004726  023727  001134  000024          CMP     RTNNO,#24           ;CHECK IF TEST 24
004734  001002                          BNE     6$                  ;CONTINUE IF NOT CONTROL-C
004736  000137  005372                  JMP     20$
004742  023727  001176  000177  6$:     CMP     TEMPCH,#177         ;CHECK IF RUBOUT
004750  001427                          BEQ     9$                  ;YES, CHECK TEST NUMBER
004752  023727  001134  000017          CMP     RTNNO,#17           ;TEST 17?
004760  001003                          BNE     7$                  ;BRANCH IF NOT
004762  013703  001176                  MOV     TEMPCH,R3           ;SAVE CHAR
004766  000461                          BR      12$                 ;CONTINUE
004770  023727  001134  000021  7$:     CMP     RTNNO,#21           ;TEST 21?
004776  001004                          BNE     8$                  ;BRANCH IF NOT
005000  013737  001176  001160          MOV     TEMPCH,REPT         ;SAVE CHAR
005006  000451                          BR      12$                 ;CONTINUE
005010  023727  001134  000022  8$:     CMP     RTNNO,#22           ;TEST 22?
005016  001056                          BNE     14$                 ;CONTINUE IF NOT
```

```
005020  013737  001176  001160          MOV     TEMPCH,REPT     ;SAVE CHAR
005026  000441                          BR      12$             ;CONTINUE
005030  023727  001134  000021  9$:     CMP     RTNNO,#21       ;CHECK IF TEST 21
005036  001011                          BNE     10$             ;NO, CHECK IF TEST 22
005040  022626                          POPSP2                  ;ASJUST STACK
005042  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
005046  104010                          DELAY
005050  104007                          TYPEM                   ;YES, TEST 21
005052  017060                          ECOEND                  ;PRINT TERMINATION MESSAGE
005054  104005                          CHAIN                   ;CHAIN TO NEXT TEST
005056  000137  011664                  JMP     E021A           ;REPEAT TEST IF LOOP ON TEST SW SET
005062  023727  001134  000022  10$:    CMP     RTNNO,#22       ;CHECK IF TEST 22
005070  001011                          BNE     11$             ;NO, CHECK IF TEST 24
005072  022626                          POPSP2                  ;ADJUST STACK
005074  012700  000036                  MOV     #30.,R0         ;@ELAY FOR HALF DUPLEX
005100  104010                          DELAY
005102  104007                          TYPEM                   ;YES, PRINT TERMINATION MESSAGE
005104  017060                          ECOEND
005106  104005                          CHAIN                   ;CHAIN TO NEXT TEST
005110  000137  011722                  JMP     E022A           ;REPEAT TEST IF LOOP ON TEST SW SET
005114  023727  001134  000024  11$:    CMP     RTNNO,#24       ;TEST 24?
005122  001137                          BNE     22$             ;WAIT FOR NEXT TEST IF NOT TEST 24
005124  022626                          POPSP2                  ;RESET STACK
005126  000137  012644                  JMP     TERM            ;TARMINATE TEST
005132  012700  000036          12$:    MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
005136  104010                          DELAY
005140  013700  001176                  MOV     TEMPCH,R0       ;SET NEW CHAR
005144  000403                          BR      14$             ;CONTINUE
005146  062737  000002  001166  13$:    ADD     #2,XCSR         ;SET STATUS ADDRESS IN XCSR
005154  062737  000002  001166  14$:    ADD     #2,XCSR
005162  013737  001102  001210          MOV     CONADD,TEMP     ;CHECK IF CONSOLE TERM
005170  062737  000004  001210          ADD     #4,TEMP         ;IS THIS DL
005176  023737  001210  001166          CMP     TEMP,XCSR
005204  001420                          BEQ     17$
005206  105777  173754          15$:    TSTB    @XCSR           ;TEST PRINTER READY
005212  100375                          BPL     15$             ;WAIT FOR READY
005214  062737  000002  001166          ADD     #2,XCSR         ;SET XCSR TO PRINTER BUFFER
005222  010077  173740                  MOV     R0,@XCSR        ;LOAD CHARACTER INTO BUFFER
005226  005337  001164          16$:    DEC     COUNT3          ;DECREASE COUNT OF DL11'S
005232  001411                          BEQ     18$             ;ALL DONE,EXIT
005234  062737  000002  001166          ADD     #2,XCSR         ;SET XCSR TO NEXT DL11 PRINTER STATUS
005242  000137  004630                  JMP     4$              ;GO TEST NEXT DL11 READY FLAG
005246  062737  000002  001166  17$:    ADD     #2,XCSR         ;SET XCSR TO PRINTER BUFFER
005254  000764                          BR      16$             ;DO NOT LOAD BUFFER
005256  023727  001134  000032  18$:    CMP     RTNNO,#32       ;CHECK IF TEST 32
005264  001514                          BEQ     26$             ;DON'T CHECK FOR INPUT IF TEST 32
005266  005737  001206                  TST     INCHK           ;WANT INPUT CHECK?
005272  001111                          BNE     26$             ;NO, BRANCH
005274  023727  001134  000020          CMP     RTNNO,#20       ;PRINTING TEST?
005302  002004                          BGE     19$             ;BRANCH IF NOT
005304  022737  104011  002154          CMP     #TTYCTL,WAITF   ;KEYBOARD CONTROL?
005312  001101                          BNE     26$             ;SKIP INPUT CHECK IF NOT
005314  105777  173562          19$:    TSTB    @CONADD         ;TEST IF ANY INPUT
005320  100076                          BPL     26$             ;BRANCH IF NONE
```

```
005322  013737  001102  001210          MOV     CONADD,TEMP         ;SET ADR
005330  062737  000002  001210          ADD     #2,TEMP
005336  117737  173646  001176          MOVB    @TEMP,TEMPCH
005344  042737  177600  001176          BIC     #177600,TEMPCH      ;MASK UNWANTED BITS
005352  023727  001176  000003          CMP     TEMPCH,#3           ;CHAR = CONTROL-C?
005360  001013                          BNE     21$                 ;CONTINUE IF NOT
005362  023727  001134  000024          CMP     RTNNO,#24           ;TEST 24?
005370  001007                          BNE     21$                 ;CONTINUE IF NOT
005372  012700  000036          20$:    MOV     #30.,R0             ;DELAY FOR HALF DUPLEX
005376  104010                          DELAY
005400  104012                          CRLF                        ;SEND CR-LF
005402  022626                          POPSP2                      ;RESET STACK
005404  000137  012516                  JMP     E0248               ;RETURN TO TEST
005410  023727  001176  000177  21$:    CMP     TEMPCH,#177         ;CHECK IF RUBOUT
005416  001006                          BNE     23$                 ;BRANCH IF NO
005420  000603                          BR      9$
005422  012737  000001  001132  22$:    MOV     #1,CNTLSW           ;CLEAR LOOP AND SEQUENCE BITS
005430  000137  002336                  JMP     TTY18               ;GO WAIT FOR NEXT TEST
005434  010046                  23$:    MOV     R0,-(SP)            ;SAVE R0
005436  012700  000036                  MOV     #30.,R0             ;DELAY FOR HALF DUPLEX
005442  104010                          DELAY
005444  012600                          MOV     (SP)+,R0            ;RESTORE R0
005446  023727  001134  000017          CMP     RTNNO,#17           ;CHECK IF TEST 17
005454  001002                          BNE     24$                 ;BRANCH IF NOT TEST 17
005456  013703  001176                  MOV     TEMPCH,R3           ;STORE INPUTTED CHAR
005462  023727  001134  000021  24$:    CMP     RTNNO,#21           ;CHECK IF TEST 21
005470  001003                          BNE     25$                 ;BRANCH IF NOT TEST 21
005472  013737  001176  001160          MOV     TEMPCH,REPT         ;STORE INPUTTED CHAR
005500  023727  001134  000022  25$:    CMP     RTNNO,#22           ;CHECK IF TEST 22
005506  001003                          BNE     26$                 ;BRANCH IF NOT TEST 22
005510  013737  001176  001160          MOV     TEMPCH,REPT         ;STORE INPUTTED CHAR
005516  000002                  26$:    RTI                         ;RETURN TO TEST
```



```
                                        .SBTTL  I/O LOGIC TESTS
                                ;
                                ;**********
                                ;
                                ;ONLY THE CONSOLE TERMINAL IS TESTED.
                                ;UPON COMPLETION, THE CPU WILL EITHER HALT IF SR
                                ;BIT8 IS = 1 AND AWAIT FUTHER INSTRUCTIONS OR CONTINUE
                                ;AND EXECUTE THE PRINTER TESTS CONTINOUSLY
                                ;IF AN I/O TEST FAILS, THE CPU WILL HALT AT ERRHLT
                                ;WITH THE ADDRESS OF THE ERROR IN R0 (LOC 777700). PRESSING
                                ;THE CONTINUE SWITCH WILL CAUSE THE I/O TEST TO
                                ;CONTINUE WITH THE NEXT TEST. HOWEVER IF SWITCH 14
                                ;WERE SET, OR IS SET BEFORE THE CONTINUE SWITCH IS
                                ;PRESSED, THE FAILED TEST WILL LOOP ON ITSELF
                                ;WITHOUT FUTHER HALTS
                                ;
                                ;**********
                                ;AT0--  TEST #40--TESTS THE ABILITY TO REFERENCE THE
                                ;       RECEIVER STATUS WORD (TKS) WITHOUT TRAPPING.
                                ;**********

005520  000040                  AT0:    40                          ;TEST NUMBER
005522  005552                  AT0X:   AT1                         ;NEXT TEST
005524  000012                          10.                         ;ITERATION COUNT
```

```
005526  005536                          1$                  ;SCOPE ENTRY
005530  012737  005546  000004          MOV     #3 ,MACHER  ;SET UP MACHINE ERROR TRAP
005536  005777  173346          1$:     TST     @TKS        ;REFERENCE RECEIVER STATUS WORD
005542  104005                  2$:     CHAIN               ;CHAIN TO NEXT TEST
005544  000774                          BR      1$          ;REPEAT TEST
005546  104001                  3$:     ERROR               ;ERROR TRAPPED WHEN REFERENCING
005550  000774                          BR      2$          ;RECEIVER STATUS WORD (TKS)

                                ;**********
                                ;AT1--TEST #41--TESTS THE ABILITY TO  REFERENCE THE
                                ;       RECEIVER BUFFER (TKB) WITHOUT TRAPPING.
                                ;**********

005552  000041                  AT1:    41                  ;TEST NUMBER
005554  005604                          AT2                 ;NEXT TEST
005556  000012                          10.                 ;ITERATION COUNT
005560  005570                          1$                  ;SCOPE ENTRY
005562  012737  005600  000004          MOV     #3$,MACHER  ;SET UP MACHINE ERROR TRAP
005570  005777  173316          1$:     TST     @TKB        ;REFERENCE RECEIVER BUFFER
005574  104005                  2$:     CHAIN               ;CHAIN TO NEXT TEST
005576  000774                          BR      1$          ;REPEAT TEST
005600  104001                  3$:     ERROR               ;TRAPPED WHEN REFERENCING
005602  000774                          BR      2$          ;RECEIVER BUFFER (TKB)
```

```
                                    ;**********
                                    ;AT2--TEST #42--TESTS THE ABILITY TO REFERENCE THE
                                    ;            TRANSMITTER STATUS WORD (TPS) WITHOUT TRAPPING.
                                    ;**********

005604  000042                      AT2:    42                              ;TEST NUMBER
005606  005636                              AT3                             ;NEXT TEST
005610  000012                              10.                             ;ITERATION COUNT
005612  005622                              1$                              ;SCOPE ENTRY
005614  012737  005632  000004              MOV     #3 ,MACHER              ;SET UP MACHINE ERROR TRAP
005622  005777  173266              1$:     TST     @TPS                    ;REFERENCE TRANSMITTER STATUS
005626  104005                      2 :     CHAIN                           ;CHAIN TO NEXT TEST
005630  000774                              BR      1$                      ;REPEAT TEST
005632  104001                      3$:     ERROR                           ;TRAPPED WHEN REFERENCING
005634  000774                              BR      2$                      ;TRANSMITTER STATUS WORD

                                    ;**********
                                    ;AT3--  TEST #43--TESTS THE ABILITY TO REFERENCE THE
                                    ;            TRANSMITTER BUFFER (TPB) WITHOUT TRAPPING.
                                    ;**********

005636  000043                      AT3:    43                              ;TEST NUMBER
005640  005670                              AT4                             ;JEXT TEST
005642  000012                              10.                             ;ITERATION COUNT
005644  005654                              1$                              ;SCOPE ENTRY
005646  012737  005664  000004              MOV     #3$(MACHER              ;SET UP ERROR TRAP
005654  005777  173236              1$:     TST     @TPB                    ;REFERENCE TRANSMITTER BUFFER
005660  104005                      2$:     CHAIN                           ;CHAIN TO NEXT TEST
005662  000774                              BR      1$                      ;REPEAT TEST
005664  104001                      3$:     ERROR                           ;TRAPPED WHEN REFERENCING
005666  000774                              BR      2$                      ;TRANSMITTER BUFFER.
```

```
                                        ;**********
                                        ;AT4--   TEST #44--TESTS THE ABILITY TO SET AND CLEAR THE
                                        ;        RECEIVER INTERRUPT ENABLE BIT.
                                        ;
                                        ;**********

005670  000044                          AT4:    44                      ;TEST NUMBER
005672  005760                                  AT5                     ;JEXT TEST
005674  000012                                  10.                     ;ITERATION COUNT
005676  005712                                  1$                      ;SCOPE ENTRY
005700  012746  000340                          MOV     #PRTY7,-(SP)    ;SET PRIORITY 7
005704  012746  005712                          MOV     #1$,-(SP)
005710  000002                                  RTI
005712  052777  000100  173170          1$:     BIS     #BIT6,@TKS      ;SET INTERRUPT ENABLE BIT
005720  032777  000100  173162                  BIT     #BIT6,@TKS      ;CHECK IS BIT IS SET
005726  001002                                  BNE     3$              ;BRANCH IF SET
005730  104001                          2$:     ERROR                   ;NOT SET, ERROR
005732  000410                                  BR      5$              ;TRY AGAIN
005734  042777  000100  173146          3$:     BIC     #BIT6,@TKS      ;CLEAR INTERRUPT ENABLE BIT
005742  032777  000100  173140                  BIT     #BIT6,@TKS      ;CHECK IF BIT IS CLEARED
005750  001401                                  BEQ     5$              ;BRANCH IF CLEARED
005752  104001                          4$:     ERROR                   ;NOT CLEARED, ERROR
005754  104005                          5$:     CHAIN                   ;CHAIN TO NEXT TEST
005756  000755                                  BR      1$              ;DO TEST AGAIN

                                        ;**********
                                        ;AT5--   TEST #45--CHECKS THAT THE RECEIVER INTERRUPT
                                        ;        ENABLE BIT CAN BE CLEARED WITH RESET INSTRUCTION.
                                        ;**********
                                        ;
005760  000045                          AT5:    45                      ;TEST NUMBER
005762  006030                                  AT6                     ;JEXT TEST
005764  000012                                  10.                     ;ITERATION COUNT
005766  006002                                  1$                      ;SCOPE ENTRY
005770  012746  000340                          MOV     #PRTY7,-(SP)    ;SET PRIORITY TO 7
005774  012746  006002                          MOV     #1$,-(SP)
006000  000002                                  RTI
006002  052777  000100  173100          1$:     BIS     #BIT6,@TKS      ;SET INTERRUPT ENABLE BIT
006010  000005                                  RESET                   ;RESET
006012  032777  000100  173070                  BIT     #BIT6,@TKS      ;TEST INTERRUPT ENABLE BIT
006020  001401                                  BEQ     2$              ;BRANCH IF CLEARED
006022  104001                                  ERROR                   ;STILL SET,ERROR
006024  104005                          2$:     CHAIN                   ;CHAIN TO NEXT ROUTINE
006026  000765                                  BR      1$              ;REPEAT TEST
```

```
                                        ;**********
                                        ;AT6--   TEST#46--TESTS THE ABILITY TO SET AND CLEAR
                                        ;        TRANSMITTER INTERRUPT ENABLE BIT.
                                        ;**********

006030  000046                          AT6:    46                              ;TEST NUMBER
006032  006120                                  AT7                             ;JEXT TEST
006034  000012                                  10.                             ;ITERATION COUNT
006036  006052                                  1$                              ;SCOPE ENTRY
006040  012746  000340                          MOV     #PRTY7,-(SP)            ;SET PRIORITY TO 7
006044  012746  006052                          MOV     #1$,-(SP)
006050  000002                                  RTI
006052  052777  000100  173034          1$:     BIS     #BIT6,@TPS              ;SET INTERRUPT ENABLE BIT
006060  032777  000100  173026                  BIT     #BIT6,@TPS              ;CHECK THAT BIT IS SET
006066  001002                                  BNE     2$                      ;BRANCH IF SET
006070  104001                                  ERROR                           ;NOT SET, ERROR
006072  000410                                  BR      3$                      ;TRY AGAIN
006074  042777  000100  173012          2 :     BIC     #BIT6,@TPS              ;CLEAR INTERRUPT ENABLE BIT
006102  032777  000100  173004                  BIT     #BIT6,@TPS              ;CHECK IF BIT IS CLEARED
006110  001401                                  BEQ     3$                      ;BRANCH IF CLEARED
006112  104001                                  ERROR                           ;NOT CLEARED, ERROR
006114  104005                          3$:     CHAIN                           ;CHAIN TO NEXT TEST
006116  000755                                  BR      1$                      ;DO AGAIN

                                        ;**********
                                        ;AT7--   TEST #47--TESTS THE ABILITY TO CLEAR TRANSMITTER
                                        ;        INTERRUPT ENABLE BIT WITH RESET INSTRUCTION.
                                        ;**********

006120  000047                          AT7:    47                              ;TEST NUMBER
006122  006170                                  AT10                            ;NEXT TEST
006124  000012                                  10.                             ;ITERATION COUNT
006126  006142                                  1$                              ;SCOPE ENTRY
006130  012746  000340                          MOV     #PRTY7,-(SP)            ;SET PRIORITY TO 7
006134  012746  006142                          MOV     #1$,-(SP)
006140  000002                                  RTI
006142  052777  000100  172744          1$:     BIS     #BIT6,@TPS              ;SET INTERRUPT BIT
006150  000005                                  RESET                           ;RESET
006152  032777  000100  172734                  BIT     #BIT6,@TPS              ;CHECK IF BIT IS CLEARED
006160  001401                                  BEQ     2$                      ;BRANCH IF CLEARED
006162  104001                                  ERROR                           ;ERROR, RESET DID NOT CLEAR BIT
006164  104005                          2$:     CHAIN                           ;CHAIN TO NEXT ROUTINE
006166  000765                                  BR      1$                      ;REPEAT TEST
```

```
                                  ;**********
                                  ;AT10-- TEST #50--CHECKS THAT RESET SETS THE TRANSMITTER
                                  ;       READY BIT AND THAT THE READY BIT CAN BE READ RELIABLY.
                                  ;
                                  ;**********

006170  000050                    AT10:   50                      ;TEST NUMBER
006172  006216                            AT11                    ;NEXT TEST
006174  000012                            10.                     ;ITERATION COUNT
006176  006200                            1$                      ;SCOPE ENTRY
006200  000005                    1$:     RESET                   ;RESET
006202  105777  172706                    TSTB    @TPS            ;CHECK TRANSMIT READY BIT
006206  100401                            BMI     2$              ;BRANCH IF SET
006210  104001                            ERROR                   ;ERROR, RESET DID NOT SET READY BIT
006212  104005                    2$:     CHAIN                   ;CHAIN TO NEXT TEST
006214  000771                            BR      1$              ;DO AGAIN

                                  ;**********
                                  ;AT11-- TEST #51--TESTS THAT THE TRANSMITTER READY RESETS
                                  ;       BY LOADING THE TRANSMITTER BUFFER.
                                  ;
                                  ;**********

006216  000051                    AT11:   51                      ;TEST NUMBER
006220  006256                            AT12                    ;NEXT TEST
006222  000012                            10.                     ;ITERATION COUNT
006224  006226                            1$                      ;SCOPE ENTRY
006226  012700  000226            1 :     MOV     #226,R0
006232  104010                            DELAY                   ;DELAY 150 MSEC.
006234  000005                            RESET                   ;RESET
006236  005077  172654                    CLR     @TPB            ;LOAD TRANSMITTER BUFFER
006242  105777  172646                    TSTB    @TPS            ;CHECK TRANSMIT READY BIT
006246  100001                            BPL     2$              ;BRANCH IF CLEARED
006250  104001                            ERROR                   ;NOT CLEARED, ERROR
006252  104005                    2$:     CHAIN                   ;CHAIN TO NEXT TEST
006254  000764                            BR      1$              ;REPEAT TEST
```

```
                                        ;**********
                                        ;AT12-- TEST #52--CHECKS THAT THE TRANSMIT READY BIT CAN
                                        ;       CAUSE AN INTERRUPT
                                        ;**********

006256  000052                          AT12:   52                      ;TEST NUMBER
006260  006330                                  AT13                    ;JEXT TEST
006262  000012                                  10.                     ;ITERATION COUNT
006264  006272                                  1$                      ;SCOPE ENTRY
006266  104004                                  STPCHV                  ;SET UP TRANSMITTER INTERRUPT VECTOR
006270  006324                                  4$                      ;TO 3$
006272  005077  172616                  1 :     CLR     @TPS            ;DISABLE TRANSMIT INTERRUPT
006276  005046                                  CLR     -(SP)           ;SET PRIORITY TO ZERO
006300  012746  006306                          MOV     #2$,-(SP)
006304  000002                                  RTI
006306  052777  000100  172600          2$:     BIS     #BIT6,@TPS      ;ENABLE TRANSMIT INTERRUPT
006314  000240                                  NOP
006316  104001                                  ERROR                   ;TRANSMIT READY DID NOT CAUSE INTERRUPT
006320  104005                          3$:     CHAIN                   ;CHAN TO NEXT TEST
006322  000763                                  BR      1$              ;REPEAT TEST
006324  022626                          4$:     POPSP2                  ;INTERRUPT OCCURED, CLEAN STACK
006326  000774                                  BR      3$              ;CHAIN TO NEXT TEST

                                        ;**********
                                        ;AT13-- TEST#35--TESTS THAT THE TRANSMIT READY DOES NOT CAUSE AN
                                        ;       INTERRUPT WHEN THE PROCESSOR IS AT THE SAME
                                        ;**********

006330  000035                          AT13:   35                      ;TEST NUMBER
006332  006410                                  AT14                    ;NEXT TEST
006334  000012                                  10.                     ;ITERATION COUNT
006336  006344                                  1$                      ;SCOPE ENTRY
006340  104004                                  STPCHV                  ;SET UP TRANSMIT INTERRUPT
006342  006402                                  4$                      ;VECTOR TO 3$
006344  013746  001126                  1 :     MOV     TPLVL,-(SP)     ;SET PROCESSOR TO SAME LEVEL AS XMITTER
006350  012746  006356                          MOV     #2$,-(SP)
006354  000002                                  RTI
006356  005077  172532                  2$:     CLR     @TPS            ;DISABLE TRANSMITTER INTERRUPTS
006362  052777  000100  172524                  BIS     #BIT6,@TPS      ;ENABLE TRANSMITTER INTERRUPTS
006370  000240                                  NOP
006372  005077  172516                  3$:     CLR     @TPS            ;OK, NO INTERRUPT OCCURED
006376  104005                                  CHAIN                   ;CHAIN TO NEXT TEST
006400  000761                                  BR      1$              ;REPEAT TEST
006402  022626                          4$:     POPSP2                  ;INTERRUPT OCURRED,ERROR,CLEAN
006404  104001                                  ERROR                   ;UP STACK
006406  000771                                  BR      3$              ;CHAIN TO NEXT TEST
```

```
                                        ;**********
                                        ;AT14--  TEST#54--TESTS THAT THE TRANSMIT READY DOES CAUSE AN
                                        ;        INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY LEVEL
                                        ;        ONE LOWER THAN THE TRANSMIT INTERRUPT REQUEST LEVEL
                                        ;**********

006410  000054                          AT14:   54                      ;TEST NUMBER
006412  006474                                  AT15                    ;NEXT TEST
006414  000012                                  10.                     ;ITERATION COUNT
006416  006424                                  1$                      ;SCOPE ENTRY
006420  104004                                  STPCHV                  ;SET UP TRANSMIT INTERRUPT
006422  006462                                  3$                      ;VECTOR TO 2$
006424  005077  172464                  1$:     CLR     @TPS            ;DISABLE TRANSMIT INTERRUPTS
006430  013746  001126                          MOV     TPLVL,-(SP)     ;SET PROCESSOR PRIORITY ONE
006434  162716  000040                          SUB     #40,(SP)        ;LEVEL LOWER THAN TRANSMITTER
006440  012746  006446                          MOV     #2$,-(SP)
006444  000002                                  RTI
006446  052777  000100  172440          2$:     BIS     #BIT6,@TPS      ;ENABLE TRANSMITTER INTERRUPTS
006454  000240                                  NOP
006456  104001                                  ERROR                   ;NO INTERRUPT, ERROR
006460  000401                                  BR      4$              ;CHAIN TO NEXT TEST
006462  022626                          3$:     POPSP2                  ;INTERRUPT OCCURED, OK, CLEAN STACK
006464  005077  172424                  4$:     CLR     @TPS            ;DISABLE TRANSMITTER INTERRUPTS
006470  104005                                  CHAIN                   ;CHAIN TO NEXT TEST
006472  000754                                  BR      1$              ;REPEAT TEST
```

```
                                        ;**********
                                        ;AT15-- TEST#55--TESTS THAT THE TRANSMIT READY DOES NOT
                                        ;       REINTERRUPT AFTER AN RTI WHEN THE READY BIT HAS
                                        ;       NOT BEEN RESET.
                                        ;**********

006474  000055                          AT15:   55                      ;TEST NUMBER
006476  006574                                  AT16                    ;NEXT TEST
006500  000012                                  10.                     ;ITERATION COUNT
006502  006504                                  1$                      ;SCOPE ENTRY
006504  104004                          1$:     STPCHV                  ;SET TRANSMIT INTERRUPT VECTOR
006506  006546                                  4$                      ;TO 3$
006510  005077  172400                          CLR     @TPS            ;DISABLE TRANSMITTER INTERRUPTS
006514  005046                                  CLR     -(SP)           ;SET PROCESSOR PRIORITY TO ZERO
006516  012746  006524                          MOV     #2$,-(SP)
006522  000002                                  RTI
006524  052777  000100  172362          2$:     BIS     #BIT6,@TPS      ;ENABLE TRANSMITTER INTERRUPTS
006532  000240                                  NOP
006534  104001                                  ERROR                   ;ERROR1, TRANSMITTER FAILED TO INTERRUPT
006536  005077  172352                  3$:     CLR     @TPS            ;DISABLE TRANSMITTER INTERRUPTS
006542  104005                                  CHAIN                   ;CHAIN TO NEXT TEST
006544  000757                                  BR      1$              ;REPEAT TEST
006546  012777  006566  172350          4$:     MOV     #6$,@TPVTR      ;INTERRUPT OCCURED, CHANGE INTERRUPT
006554  012716  006562                          MOV     #5$,@SP         ;VECTOR TO 5$ AND RETURN TO 4$
006560  000002                                  RTI                     ;RETURN FROM INTERRUPT
006562  000240                          5$:     NOP
006564  000764                                  BR      3$              ;CHAIN TO NEXT TEST
006566  022626                          6 :     POPSP2                  ;ERROR2, TRANSMITTER REINTERRUPTED
006570  104001                                  ERROR                   ;AFTER RTI WITH READY BIT LEFT ON.
006572  000761                                  BR      3$              ;CLEAN STACK, CHAIN TO NEXT TEST.

                                        ;**********
                                        ;AT16--TEST#56--CHECKS THAT RESET CLEARS THE RECEIVER DONE BIT
                                        ;**********

006574  000056                          AT16:   56                      ;TEST NUMBER
006576  006632                                  AT17                    ;NEXT TEST
006600  000012                                  10.                     ;ITERATION COUNT
006602  006604                                  1$                      ;SCOPE ENTRY
006604  012700  000226                  1$:     MOV     #226,R0
006610  104010                                  DELAY                   ;DELAY 150 MSEC.
006612  104021                          2$:     AREAD                   ;ENABLE RECEIVER
006614  000005                                  RESET                   ;RESET
006616  105777  172266                          TSTB    @TKS            ;TEST DONE BIT
006622  100001                                  BPL     3$              ;BRANCH IF DONE IS CLEARED
006624  104001                                  ERROR                   ;NOT CLEARED, ERROR
006626  104005                          3$:     CHAIN                   ;CHAIN TO NEXT TEST
006630  000770                                  BR      2$              ;REPEAT TEST
```

```
                                        ;**********
                                        ;AT17-- TEST#57--CHECKS THAT REFERENCING THE RECEIVER BUFFER
                                        ;       CLEARS THE DONE BIT.
                                        ;**********

006632  000057                          AT17:   57                              ;TEST NUMBER
006634  006672                                  AT20                            ;NEXT TEST
006636  000012                                  10.                             ;ITERATION COUNT
006640  006642                                  1$                              ;SCOPE ENTRY
006642  012700  000226                  1$:     MOV     #226,R0
006646  104010                                  DELAY                           ;DELAY 150 MSEC.
006650  104021                          2$:     AREAD                           ;ENABLE RECEIVER
006652  105777  172234                          TSTB    @TKB                    ;REFERENCE RECEIVER BUFFER
006656  105777  172226                          TSTB    @TKS                    ;TEST DONE BIT
006662  100001                                  BPL     3$                      ;BRANCH IF NOT SET
006664  104001                                  ERROR                           ;DONE BIT IS SET, ERROR
006666  104005                          3$:     CHAIN                           ;CHAIN TO NEXT TEST
006670  000767                                  BR      2$                      ;REPEAT TEST

                                        ;**********
                                        ;AT20-- TEST#60--CHECK THAT THE RECEIVER DONE BIT IS ABLE TO
                                        ;       CAUSE AN INTERRUPT.
                                        ;**********

006672  000060                          AT20:   60                              ;TEST NUMBER
006674  006754                                  AT21                            ;JEXT TEST
006676  000012                                  10.                             ;ITERATION COUNT
006700  006716                                  2$                              ;SCOPE ENTRY
006702  104003                                  STRDRV                          ;SET UP RECEIVER INTERRUPT
006704  006746                                  4$                              ;VECTOR TO 3$
006706  012700  000226                  1$:     MOV     #226,R0
006712  104010                                  DELAY                           ;DELAY 150 MSEC
006714  104021                                  AREAD                           ;ENABLE RECEIVER
006716  005077  172166                  2$:     CLR     @TKS                    ;DISABLE RECEIVER INTERRUPTS
006722  005046                                  CLR     -(SP)                   ;SET PROCESS STATUS TO ZERO
006724  012746  006732                          MOV     #3$,-(SP)
006730  000002                                  RTI
006732  052777  000100  172150          3$:     BIS     #BIT6,@TKS              ;ENABLE RECEIVER INTERRUPT
006740  000240                                  NOP
006742  104001                                  ERROR                           ;ERR,RECEIVER FAILED TO INTERRUPT
006744  000401                                  BR      5$                      ;CHAIN TO NEXT TEST
006746  022626                          4$:     POPSP2                          ;OK, CLEAN STACK
006750  104005                          5$:     CHAIN                           ;CHAIN TO NEXT TEST
006752  000761                                  BR      2$                      ;REPEAT TEST
```

```
                                          ;**********
                                          ;AT21--  TEST#61--TESTS THAT THE RECEIVER DONE DOES NOT CAUSE AN
                                          ;        INTERRUPT WHEN THE PROCESSOR IS AT THE SAME LEVEL AS
                                          ;        THE RECEIVER'S INTERRUPT REQUEST LEVEL.
                                          ;**********

006754  000061                            AT21:   61                              ;TEST NUMBER
006756  007044                                    AT22                            ;NEXT TEST
006760  000012                                    10.                             ;ITERATION COUNT
006762  007000                                    2$                              ;SCOPE ENTRY
006764  104003                                    STRDRV                          ;SET RECEIVER VECTOR TO 4$
006766  007036                                    5$
006770  012700  000226                    1$:     MOV     #226,R0
006774  104010                                    DELAY                           ;DELAY 150 MSEC
006776  104021                                    AREAD                           ;ENABLE RECEIVER
007000  005077  172104                    2$:     CLR     @TKS                    ;DISABLE RECEIVER INTERRUPTS
007004  013746  001122                            MOV     TKLVL,-(SP)             ;SET PROCESSOR TO SAME LEVEL AS RECEIVER
007010  012746  007016                            MOV     #3$,-(SP)
007014  000002                                    RTI
007016  052777  000100  172064            3$:     BIS     #BIT6,@TKS              ;ENABLE RECEIVER INTERRUPTS
007024  000240                                    NOP
007026  005077  172056                    4$:     CLR     @TKS                    ;OK, NO INTERRUPT OCCURED
007032  104005                                    CHAIN                           ;CHAIN TO NEXT TEST
007034  000770                                    BR      3$                      ;REPEAT TEST
007036  022626                            5$:     POPSP2                          ;ERROR, RECEIVER INTERRUPTED, CLEAN STACK
007040  104001                                    ERROR
007042  000771                                    BR      4$                      ;BRANCH 3$
```

```
                                        ;**********
                                        ;AT22-- TEST#62--TESTS THAT THE RECEIVER DONE DOES CAUSE AN
                                        ;       INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE
                                        ;       LEVEL LOWER THAN THE RECEIVER'S INTERRUPT
                                        ;       REQUEST LEVEL
                                        ;**********

007044  000062                          AT22:   62                              ;TEST NUMBER
007046  007142                                  AT23                            ;NEXT TEST
007050  000012                                  10.                             ;ITERATION COUNT
007052  007070                                  1$                              ;SCOPE ENTRY
007054  104003                                  STRDRV                          ;SET RECEIVER INTERRUPT
007056  007130                                  3$                              ;VECTOR TO 2$
007060  012700  000226                          MOV     #226,R0
007064  104010                                  DELAY                           ;DELAY 150 MSEC
007066  104021                                  AREAD                           ;ENABLE RECEIVER
007070  005077  172014                  1$:     CLR     @TKS                    ;DISABLE READER INTERRUPTS
007074  013746  001122                          MOV     TKLVL,-(SP)             ;SET PROCESSOR ONE LEVEL
007100  012746  007106                          MOV     #2$,-(SP)
007104  000002                                  RTI
007106  162737  000040  177776          2$:     SUB     #40,PSW                 ;LOWER THAN READERS
007114  052777  000100  171766                  BIS     #BIT6,@TKS              ;ENABLE INTERRUPTS
007122  000240                                  NOP
007124  104001                                  ERROR                           ;FAILED TO INTERRUPT
007126  000401                                  BR      4$                      ;CHAIN TO NEXT TEST
007130  022626                          3$:     POPSP2                          ;OK, CLEAN STACK
007132  005077  171752                  4$:     CLR     @TKS                    ;DISABLE RECEIVER INTERRUPTS
007136  104005                                  CHAIN                           ;CHAIN TO NEXT TEST
007140  000753                                  BR      1$                      ;REPEAT TEST
```

```
                                        ;**********
                                        ;AT23--  TEST#63--CHECKS THAT THE RECEIVER DONE DOES NOT
                                        ;        REINTERRUPT AFTER RTI INSTRUCTION WHEN DONE
                                        ;        BIT IS LEFT SET.
                                        ;**********

007142  000063                          AT23:   63                      ;TEST NUMBER
007144  177777                                  -1                      ;NEXT TEST (LAST TEST)
007146  000012                                  10.                     ;ITERATION COUNT
007150  007162                                  1$                      ;SCOPE ENTRY
007152  012700  000226                          MOV     #226,R0
007156  104010                                  DELAY                   ;DELAY 150 MSEC
007160  104021                                  AREAD                   ;ENABLE RECEIVER
007162  104003                          1$:     STRDRV                  ;SET RECEIVER INTERRUPT
007164  007214                                  3$                      ;VECTOR TO 3$
007166  005077  171716                          CLR     @TKS            ;DISABLE RECEIVER INTERRUPTS
007172  052777  000100  171710                  BIS     #BIT6,@TKS      ;ENABLE RECEIVER INTERRUPT
007200  000240                                  NOP                     ;
007202  104001                                  ERROR                   ;NO INTERRUPT, ERROR
007204  005077  171700                  2$:     CLR     @TKS            ;DISABLE RECEIVER INTERRUPTS
007210  104005                                  CHAIN                   ;CHAIN TO NEXT TEST
007212  000763                                  BR      1$              ;REPEAT TEST
007214  012777  007234  171676          3$:     MOV     #5$,@TKVTR      ;INTERRUPT, OK, CHANGE VECTOR TO 5$
007222  012716  007230                          MOV     #4$,@SP         ;CHANGE RET ADDR. TO 4$
007226  000002                                  RTI                     ;RETURN
007230  000240                          4$:     NOP
007232  000764                                  BR      2$              ;OK, NO ADDITIONAL INTERRUPTS
007234  022626                          5$:     POPSP2                  ;ERROR, ADDITIONAL INTERRUPT
007236  104001                                  ERROR
007240  000761                                  BR      2$              ;CHAIN TO NEXT TEST
```

```
                                        .SBTTL  LA36 PRINTER TESTS
                                ;
                                ;THE LA36 PRINTER TESTS WILL BE EXECUTED IN A
                                ;CONTINOUS LOOP OUTPUTTING TO ALL MULTIPLE DL11'S
                                ;IB SR BITS8 IS SET TO ZERO AT START UP TIME.  IF
                                ;BIT8 IS SET TO 1 TO 1 AT START UP THEY MAY BE EXECUTED
                                ;INDIVIDUALLY ONCE OR CONTINUALLY LOOPED, OR
                                ;BECOME THE FIRST OF THE ENTIRE SEQUENCE OF PRINTER
                                ;TESTS.  REFERENCE INTRUCTIONS IN THE INTRODUCTION
                                ;FOR PROPER MODE OF OPERATION.
                                ;
                                ;
                                ;XXXXXXXXXX
                                ;
                                ;PT0 -- DATA PATH TEST---FOUR LINES OF ALTERNATING
                                ;       "*" AND "U" ARE PRINTED, OUT TO THE GIVEN PAPER
                                ;       WIDTH THE PATTERN WILL APPEAR AS FOLOWS.
                                ;
                                ;       *U*U*U*U*U*U
                                ;       U*U*U*U*U*U*
                                ;       *U*U*U*U*U*U
                                ;       U*U*U*U*U*U*
                                ;
                                ;XXXXXXXXXX

007242  000000                  PT0:    0                       ;TEST NUMBER
007244  007316                          PT1                     ;NEXT TEST
007246  104016                          PRTHDR                  ;
007250  104007                          TYPEM                   ;PRINT COLUMN # MESG
007252  016715                          HDR0
007254  012703  025125          1$:     MOV     #"U*,R3         ;SET FIRST CHAR PAIR
007260  012702  000020                  MOV     #16.,R2         ;SET LINE COUNT
007264  010300                  2$:     MOV     R3,R0           ;SET CHAR PAIR
007266  013701  001150                  MOV     WIDTH,R1        ;SET COLUMN COUNT
007272  104015                  3 :     PRINTC                  ;PRINT CHAR
007274  000300                          SWAB    R0              ;SET NEXT CHAR
007276  005301                          DEC     R1              ;DEC COLUMN COUNT
007300  001374                          BNE     3$              ;FINISH LINE
007302  000303                          SWAB    R3              ;SET NEXT LINE START CHAR
007304  104012                          CRLF                    ;SEND CR-LF
007306  005302                          DEC     R2              ;DEC LINE COUNT
007310  001365                          BNE     2$              ;FINISH TEST
007312  104005                          CHAIN                   ;ALL DONE, EXIT
007314  000757                          BR      1$              ;REPEAT TEST
```

```
                                ;XXXXXXXXXXX
                                ;
                                ;PT1 -- PRINTER CHARACTER TEST --- PRINTS ALL PRINTABLE CHARACTERS
                                ;
                                ;XXXXXXXXXXX

007316  000001                  PT1:    1                       ;TEST NUMBER
007320  007440                          PT2                     ;NEXT TEST
007322  104016                          PRTHDR
007324  012701  000040          1$:     MOV     #40,R1          ;SPACE TO R1
007330  012702  000100                  MOV     #100,R2         ;@ TO R2
007334  012703  000140                  MOV     #140,R3         ;\ TO R3
007340  110100                  2$:     MOVB    R1,R0           ;CHAR TO R0
007342  004737  007410                  JSR     PC,SPSP         ;SEND TWO SPACES
007346  110200                          MOVB    R2,R0           ;JEXT CHAR TO R0
007350  004737  007410                  JSR     PC,SPSP         ;SEND TWO SPACES
007354  012704  000003                  MOV     #3,R4           ;PRINT COUNT TO R4
007360  110300                          MOVB    R3,R0           ;THIRD CHAR TO R0
007362  104015                  3$:     PRINTC                  ;PRINT THE CHAR
007364  005304                          DEC     R4              ;THREE TIMES ?
007366  001375                          BNE     3$              ;BRANCH IF NOT
007370  104012                          CRLF                    ;CARRIAGE RETURN LINE FEED
007372  122122                          CMPB    (R1)+,(R2)+     ;NEXT CHARACTERS
007374  105723                          TSTB    (R3)+
007376  020327  000200                  CMP     R3,#200         ;CHECK IF ALL DONE
007402  103756                          BLO     2$              ;BRANCH IF NOT
007404  104005                          CHAIN                   ;EXIT TO NEXT TEST
007406  000746                          BR      1$              ;REPEAT TEST
007410  012704  000003          SPSP:   MOV     #3,R4           ;PRINT COUNT TO R4
007414  104015                  1$:     PRINTC                  ;PRINT CHAR
007416  005304                          DEC     R4              ;THREE TIMES?
007420  001375                          BNE     1$              ;BRANCH IF NOT
007422  012700  000040          SP2:    MOV     #40,R0          ;SPACE TO R0
007426  104015                          PRINTC                  ;SEND A SPACE
007430  012700  000040          SPC:    MOV     #40,R0          ;SPACE TO R0
007434  104015                          PRINTC                  ;SEND ANOTHER
007436  000207                          RTS     PC              ;RETURN
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT2 -- NON-PRINTING CHARACTER TEST.  THIS TEST
                                        ;       PRINTS THE OCTAL CODE FOLLOWED BY THE MNEMONIC
                                        ;       OF ALL NON-PRINTING CHARACTERS.  FOLLOWING EACH
                                        ;       MNEMONIC, THE PRINTER IS DRIVEN BY THE NON-PRINTING
                                        ;       CODE (000 THROUGH 037 PLUS 177)
                                        ;       ALL CONTROL CHARACTERS (INCLUDING THOSE FOR OPTIONS
                                        ;       WILL BE SKIPPED, REFER TO THE DOCUMENT FOR A LIST OF THOSE
                                        ;       PRINTED.
                                        ;
                                        ;XXXXXXXXXX
                                        ;
 007440 000002                          PT2:    2                       ;TEST NUMBER
 007442 010070                                  PT3                     ;NEXT TEST
 007444 104016                                  PRTHDR
 007446 005003                          1$:     CHR     R3              ;NON-PRINTABLE CHAR..NULL IS FIRST
 007450 012701 007626                           MOV     #LINE2,R1       ;ADDR OF CHAR STRING TO R1
 007454 012702 000003                   2$:     MOV     #3,R2           ;NO. OF CHAR GROUPS PER LINE
 007460 012704 000010                   3$:     MOV     #10,R4          ;NO. OF CHARS PER GROUP
 007464 112100                          4$:     MOVB    (R1)+,R0        ;CHAR INTO R0
 007466 104015                                  PRINTC                  ;PRINT CHAR
 007470 005304                                  DEC     R4              ;8 CHARS. PRINTED?
 007472 001374                                  BNE     4$              ;BRANCH IF NOT
 007474 022703 000002                           CMP     #2,R3           ;CHAR = STX?
 007500 001420                                  BEQ     7$              ;YES, SET NEXT CHAR
 007502 022703 000004                           CMP     #4,R3           ;CHAR = EOT?
 007506 001414                                  BEQ     6$              ;YES, SET NEXT CHAR
 007510 022703 000033                           CMP     #33,R3          ;CHAR = ESC?
 007514 001412                                  BEQ     7$              ;SET NEXT CHAR
 007516 022703 000007                           CMP     #7,R3           ;YES, IS NON-PRINTABLE CHAR A BELL?
 007522 001002                                  BNE     5$              ;BRANCH IF NOT A BELL
 007524 012703 000020                           MOV     #20,R3          ;SET NEXT CHAR
 007530 022703 000040                   5$:     CMP     #40,R3          ;NO, IS IT THE LAST
 007534 001003                                  BNE     8$              ;BRANCH IF NO
 007536 000421                                  BR      11$             ;YES, OUTPUT LAST CHAR (177)
 007540 005203                          6$:     INC     R3              ;SKIP CHAR
 007542 005203                          7$:     INC     R3              ;SKIP CHAR
 007544 010300                          8$:     MOV     R3,R0           ;NON-PRINTABLE CHAR TO R0
 007546 012704 000003                           MOV     #3,R4           ;A COUNT OF 3 TO R4
 007552 104015                          9$:     PRINTC                  ;DRIVE PRINTER WITH NON-PRINTABLE CHAR
 007554 005304                                  DEC     R4              ;DECREMENT COUNTER
 007556 001375                                  BNE     9$              ;BRANCH IF NOT ZERO (3 TIMES)
 007560 005203                                  INC     R3              ;INCREMENT CHAR.CODE
 007562 005302                                  DEC     R2              ;DEC. GROUPS PER LINE COUNTER (3)
 007564 001404                                  BEQ     10$             ;BRANCH IF ZERO
 007566 004737 007422                           JSR     PC,SP2          ;SEND 3 SPACES
 007572 104015                                  PRINTC
 007574 000731                                  BR      3$              ;CONTINUE
 007576 104012                          10$:    CRLF                    ;SEND A CR,LF
 007600 000725                                  BR      2$              ;GO DO NEXT LINE
 007602 012704 000003                   11$:    MOV     #3,R4           ;A 3 COUNT TO R4
 007606 012700 000177                           MOV     #177,R0         ;A DEL TO R0
 007612 104015                          12$:    PRINTC                  ;PRINT CHAR
```

```
007614  005304                                DEC     R4                              ;DECREMENT COUNTER
007616  001375                                BNE     12$                             ;BRANCH IF NOT ZERO
007620  104012                                CRLF                                    ;SEND A CR, LF
007622  104005                                CHAIN                                   ;CHAIN TO NEXT TEST
007624  000710                                BR      1$                              ;REPEAT TEST



007626  030060  020060  047040  LINE2:  .ASCII  /000  NUL001  SOH002  STX/
007634  046125  030060  020061
007642  051440  044117  030060
007650  020062  051440  054124
007656  030060  020066  040440          *ASCII  /006  ACK020  DLE021  DC1/
007664  045503  031060  020060
007672  042040  042514  031060
007700  020061  042040  030503
007706  031060  020062  042040          .ASCII  /022  DC2023  DC3024  DC4/
007714  031103  031060  020063
007722  042040  031503  031060
007730  020064  042040  032103
007736  031060  020065  047040          .ASCII  /025  NAK026  SYN027  ETB/
007744  045501  031060  020066
007752  051440  047131  031060
007760  020067  042440  041124
007766  031460  020060  041440          .ASCII  /030  CAN031  EM 032  SUB/
007774  047101  031460  020061
010002  042440  020115  031460
010010  020062  051440  041125
010016  031460  020064  043040          .ASCII  /034  FS 035  GS 036  RS /
010024  020123  031460  020065
010032  043440  020123  031460
010040  020066  051040  020123
010046  031460  020067  052440          *ASCII  /037  US 177  DEL /
010054  020123  033461  020067
010062  042040  046105     040
        010070                          .EVEN
```

```
                                                ;XXXXXXXXXX
                                                ;
                                                ;PT3 -- CARRIAGE RETURN TEST
                                                ;
                                                ;       THE LINE CONSISTS OF A STRING OF O'S AND
                                                ;       X'S.  FIRST, THE O'S ARE PRINTED OUT TO THE LAST
                                                ;       COLUMN WITH A SPACE SEPARATING EACH.  THEN THE
                                                ;       CARRIAGE IS SPACED TO THE FIRST BLANK SPACE, AND X
                                                ;       PRINTED AND THE RETURNED TO THE MARGIN.  THIS
                                                ;       PROCESS IS CONTINUE UNTIL ALL SPACES BETWEEN
                                                ;       THE ZEROES HAVE BEEN FILLED.
                                                ;
                                                ;XXXXXXXXXX

010070  000003                          PT3:    3                       ;TEST NUMBER
010072  010210                                  PT4                     ;JEXT TEST
010074  104016                                  PRTHDR                  ;TYPE HEADER
010076  005037  001172                  1$:     CLR     SPCNT           ;CLEAR SPACE COUNTER
010102  013701  001150                          MOV     WIDTH,R1        ;POSITION COUNTER TO R1
010106  012700  000117                  2$:     MOV     #117,R0         ;"O" TO R0
010112  104015                                  PRINTC                  ;PRINT THE "O"
010114  005301                                  DEC     R1              ;DECREMENT POSITION COUNTER
010116  001404                                  BEQ     3$              ;BRANCH IF 0
010120  004737  007430                          JSR     PC,SPC          ;SEND SPACE
010124  005301                                  DEC     R1              ;DECREMENT POSITION COUNTER
010126  001367                                  BNE     2$              ;BRANCH IF NOT ZERO
010130  104022                          3$:     CR                      ;SEND A CR
010132  012737  000001  001172                  MOV     #1,SPCNT        ;SPACE, COUNTER SET TO 1
010140  013701  001172                  4$:     MOV     SPCNT,R1        ;NO. OF SPACES TO R1
010144  004737  007430                  5$:     JSR     PC,SPC          ;SEND SPACE
010150  005301                                  DEC     R1              ;DECREMENT SPACE COUNTER
010152  001374                                  BNE     5$              ;BRANCH IF NOT ZERO
010154  012700  000130                          MOV     #130,R0         ;"X" INTO R0
010160  104015                                  PRINTC                  ;PRINT "X"
010162  104022                                  CR                      ;PRINT CR
010164  062737  000002  001172                  ADD     #2,SPCNT        ;INCREMENT SPACE COUNT BY 2
010172  023737  001172  001150                  CMP     SPCNT,WIDTH     ;COMPARE POSITION COUNTER WITH COLM. COUNT
010200  103757                                  BLO     4$              ;BRANCH IF LOWER
010202  104014                                  LF                      ;SEND LF
010204  104005                                  CHAIN                   ;CHAIN TO NEXT TEST
010206  000733                                  BR      1$              ;REPEAT TEST
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;PT4 -- MULTIPLE LINE FEED TEST -- 63 LINE FEEDS ARE
                                        ;       SENT WITH A REFERENCE LINE AT THE START AND END.
                                        ;       A NUMBER IS PRINTED WHICH INDICATES THE NUMBER OF LINE
                                        ;       FEEDS THAT WILL BE ISSUED BEFORE THE NEXT
                                        ;       NUMBER OR REFERENCE LINE IS PRINTED.
                                        ;
                                        ;XXXXXXXXXX

010210  000004                          PT4:    4                       ;TEST NUMBER
010212  010366                                  PT5                     ;NEXT TEST
010214  104016                                  PRTHDR                  ;TYPE HEADER
010216  012737  000001  001204          1$:     MOV     #1(LFCNT        ;LINE FEED COUNT TO 1
010224  013701  001150                          MOV     WIDTH,R1        ;COLUMN COUNT TO R1
010230  012702  010350                          MOV     #LINE3,R2       ;ADDR OF NUMBER FIELD TO R2
010234  004737  010320                          JSR     PC,REF          ;PRINT REFERENCE LINE
010240  013701  001204                  2$:     MOV     LFCNT,R1        ;LINE FEED COUNT TO R1
010244  104014                          3$:     LF                      ;SEND LF
010246  005301                                  DEC     R1              ;DECREMENT COUNTER
010250  001375                                  BNE     3$              ;BRANCH IF NOT YET 0
010252  006337  001204                          ASL     LFCNT           ;DOUBLE LINE FEED COUNT
010256  022737  000100  001204                  CMP     #BIT6,LFCNT     ;TEST IF COUNT IS 32
010264  001406                                  BEQ     4$              ;BRANCH IF =32, END
010266  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
010270  104015                                  PRINTC                  ;PRINT IT
010272  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
010274  104015                                  PRINTC                  ;PRINT IT
010276  104022                                  CR                      ;PRINT CR
010300  000757                                  BR      2$              ;DRIVE THE LINEFEEDS
010302  013701  001150                  4 :     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
010306  004737  010320                          JSR     PC,REF          ;SEND END REFERENCE LINE
010312  104014                                  LF                      ;ADVANCE PAPER
010314  104005                                  CHAIN
010316  000737                                  BR      1$              ;REPEAT TEST
010320  112200                          REF:    MOVB    (R2)+,R0        ;NUMBER TO R0
010322  104015                                  PRINTC                  ;PRINT IT
010324  112200                                  MOVB    (R2)+,R0        ;NUMBER TO R0
010326  104015                                  PRINTC                  ;PRINT IT
010330  005741                                  TST     -(R1)           ;DECREASE COUNTER BY 2
010332  012700  000137                          MOV     #137,R0         ;DASH (-) TO R0
010336  104015                          1$:     PRINTC                  ;PRINT IT
010340  005301                                  DEC     R1              ;DECREMENT COLUMN COUNTER
010342  001375                                  BNE     1$              ;BRANCH IF NO ZERO
010344  104022                                  CR                      ;PRINT CR
010346  000207                                  RTS     PC              ;RETURN

010350  030460  031060  032060          LINE3:  .ASCII  /01020408163200/
010356  034060  033061  031063
010364  030060
```

```
                                        ;XXXXXXXXXX
                                        ;PT5--  SINGLE LINE FEED TEST --  TESTS THE LINE FEED
                                        ;       CAPABILITY FROM ALL COLUMNS.
                                        ;XXXXXXXXXX

010366  000005                          PT5:    5                               ;TEST NUMBER
010370  010572                                  PT6                             ;NEXT TEST
010372  104016                                  PRTHDR                          ;TYPE HEADER
010374  013701  001150                  1$:     MOV     WIDTH,R1                ;COLUMN COUNT TO R1
010400  005741                                  TST     -(R1)                   ;DECREASE BY 2
010402  012700  000060                          MOV     #60,R0                  ;'0' TO R0
010406  104015                          2$:     PRINTC                          ;SEND 0
010410  005301                                  DEC     R1                      ;DECREMENT COLUMN COUNTER
010412  001375                                  BNE     2$                      ;BRANCH IF NOT ZERO
010414  012700  000062                          MOV     #62,R0                  ;SEND A 2
010420  104015                                  PRINTC
010422  104015                                  PRINTC                          ;SEND A SECOND TWO
010424  023727  001150  000204                  CMP     WIDTH,#132.             ;COMPARE COLUMN COUNT
010432  001404                                  BEQ     3$                      ;BRANCH IF EQ 132
010434  012700  003410                          MOV     #3410,R0                ;DELAY 1.8 SEC
010440  104010                                  DELAY
010442  000407                                  BR      5$
010444  012700  000063                  3$:     MOV     #63,R0                  ;3'S TO R0
010450  012701  000100                          MOV     #100,R1                 ;64 TO COUNTER
010454  104015                          4$:     PRINTC                          ;SEND CHARACTER
010456  005301                                  DEC     R1                      ;DECREMENT COUNT
010460  001375                                  BNE     4$                      ;BRANCH IF NOT ZERO
010462  104012                          5$:     CRLF                            ;SEND A CR,LF
010464  013701  001150                          MOV     WIDTH,R1                ;NO. COLUMNS TO R1
010470  012700  000134                  6$:     MOV     #134,R0                 ;BACKSLASH TO R0
010474  104015                                  PRINTC                          ;SEND IT
010476  104014                                  LF                              ;PRINT LF
010500  005301                                  DEC     R1                      ;DECREMENT COUNTER
010502  001372                                  BNE     6$                      ;BRANCH IF NOT ZERO.
010504  104022                                  CR                              ;SEND CR
010506  004737  010534                          JSR     PC,PT5AL                ;SEND REF LINE #1
010512  104012                                  CRLF                            ;SEND A CR,LF
010514  012700  001750                          MOV     #1750,R0                ;DELAY 1 SEC
010520  104010                                  DELAY
010522  004737  010534                          JSR     PC,PT5AL                ;SEND A SECOND REF. LINE
010526  104012                                  CRLF                            ;SEND A CR,LF
010530  104005                                  CHAIN                           ;CHAIN TO NEXT TEST
010532  000720                                  BR      1$                      ;REPEAT TEST
010534  013701  001150                  PT5AL:  MOV     WIDTH,R1                ;COLUMN COUNT TO R1
010540  012700  000061                          MOV     #61,R0                  ;"1" TO R0
010544  104015                          1$:     PRINTC                          ;PRINT R0
010546  005301                                  DEC     R1                      ;DECREMENT COUNTER
010550  001407                                  BEQ     2$                      ;BRANCH IF=0
010552  005200                                  INC     R0                      ;IJCREMENT CHARACTER
010554  020027  000071                          CMP     R0,#71                  ;COMP CHAR TO "9"
010560  101771                                  BLOS    1$                      ;BRANCH IF LOWER OR SAME
010562  012700  000060                          MOV     #60,R0                  ;RESET CHAR TO "0"
010566  000766                                  BR      1$                      ;CONTINUE
010570  000207                          2$:     RTS     PC                      ;FINISHED, RETURN TO CALLER
```

```
                                ;XXXXXXXXXX
                                ;PT6--  BACKSPACE TEST -- A REFERENCE LINE SUCH AS IN
                                ;       TEST PT5 IS PRINTED.  THE SECOND LINE CONSISTS
                                ;       OF PRINTING A BACKSLASH, BACKSPACE AND FORWARD
                                ;       SLASH COMBINATION OUT TO THE GIVEN COLUMN WIDTH.
                                ;       THIS LINE IS THEN FOLLOWED BY THE SAME TWO REFERENCE
                                ;       LINES AS PRINTED IN TEST PT5.
                                ;XXXXXXXXXX

010572  000006                  PT6:    6                           ;TEST NUMBER
010574  010760                          PT7                         ;NEXT TEST
010576  104016                          PRTHDR                      ;PRINT HEADER
010600  104007                          TYPEM                       ;PRINT COLUMN # MESG
010602  016715                          HDRO
010604  013701  001150          1$:     MOV     WIDTH,R1            ;COLUMN COUNT TO R1
010610  005741                          TST     -(R1)               ;DECREMENT BY 2
010612  012700  000060                  MOV     #60,R0              ;"0" TO R0
010616  104015                  2$:     PRINTC                      ;SEND 0
010620  005301                          DEC     R1                  ;DECREMENT COLUMN COUNTER
010622  001375                          BNE     2$                  ;BRANCH IF NOT ZERO
010624  012700  000062                  MOV     #62,R0              ;"2" TO R0
010630  104015                          PRINTC                      ;SEND A "2"
010632  104015                          PRINTC                      ;SEND A SECOND "2"
010634  023727  001150  000204          CMP     WIDTH,#132.         ;COMPARE COLUMN COUNT
010642  001404                          BEQ     3$
010644  012700  003410                  MOV     #3410,R0            ;DELAY 1.8 SEC
010650  104010                          DELAY
010652  000407                          BR      5$
010654  012700  000063          3$:     MOV     #63,R0              ;3'S TO R0
010660  012701  000100                  MOV     #100,R1             ;64 TO CHAR COUNT
010664  104015                  4$:     PRINTC                      ;SEND CHAR
010666  005301                          DEC     R1                  ;DECREMENT CHAR COUNT
010670  001375                          BNE     4$                  ;CONTINUE IF NOT DONE
010672  104012                  5$:     CRLF                        ;SEND A CR,LF
010674  013701  001150                  MOV     WIDTH,R1            ;COLUMN COUNT TO R1
010700  012700  000134          6 :     MOV     #130,R0             ;BACKSLASH TO R0
010704  104015                          PRINTC                      ;SEND IT
010706  012700  000010                  MOV     #10,R0              ;BACKSPACE TO R0
010712  104015                          PRINTC                      ;SEND IT
010714  012700  000057                  MOV     #57,R0              ;FORWORD SLASH TO R0
010720  104015                          PRINTC                      ;SEND IT
010722  005301                          DEC     R1                  ;END OF PAPER
010724  001365                          BNE     6$                  ;BRANCH IF NO
010726  104014                          LF                          ;SEND LF
010730  104022                          CR                          ;SEND CR
010732  004737  010534                  JSR     PC,PT5AL            ;SEND REF LINE #1
010736  104012                          CRLF                        ;SEND A CR,LF
010740  012700  001750                  MOV     #1750,R0            ;DELAY 1 SEC
010744  104010                          DELAY
010746  004737  010534                  JSR     PC,PT5AL            ;SEND SECOND REF LINE
010752  104012                          CRLF                        ;SEND A CR,LF
010754  104005                          CHAIN                       ;CHAIN TO NEXT TEST
010756  000712                          BR      1$                  ;REPEAT TEST
```

```
                                    ;XXXXXXXXXX
                                    ;
                                    ;PT7--  OVERPRINT TEST-- A ROW OF ALTERNATING M'S AND
                                    ;       SPACES ARE PRINTED, OUT TO THE LAST COLUMN AND OVERPRINT TWICE.
                                    ;       A SECOND LINE OF ALTERNATING SPACES AND "@'S" IS THEN
                                    ;       SENT 3 TIMES AS THE FIRST LINE.  THIS IS FOLLOWED
                                    ;       BY A THIRD AND FINAL LINE OF ALTERNATING '&'
                                    ;       AND SPACES.
                                    ;
                                    ;XXXXXXXXXX

010760  000007                      PT7:    7                       ;TEST NUMBER
010762  011172                              PT10                    ;NEXT TEST
010764  104016                              PRTHDR                  ;PRINT MESSAGE
010766  012703  000002              1$:     MOV     #2,R3           ;2 COUNT TO R3
010772  013701  001150              2$:     MOV     WIDTH,R1        ;NO. OF COLUMNS TO R1
010776  012700  000115              3$:     MOV     #115,R0         ;"M" TO R0
011002  104015                              PRINTC                  ;SEND IF
011004  005301                              DEC     R1              ;END OF LINE
011006  001404                              BEQ     4$              ;BRANCH IF YES
011010  004737  007430                      JSR     PC,SPC          ;SEND SPACE
011014  005301                              DEC     R1              ;END OF LINE?
011016  001367                              BNE     3$              ;BRANCH IF NO
011020  022703  000002              4$:     CMP     #2,R3           ;TEST R3
011024  001003                              BNE     6$              ;BRANCH IF NOT FIRST TIME
011026  104022                      5$:     CR                      ;SEND CR
011030  005303                              DEC     R3              ;DECREASE LINE COUNTER
011032  000757                              BR      2$              ;REPEAT LINE
011034  005703                      6$:     TST     R3              ;THIRD TIME?
011036  001373                              BNE     5$              ;BRANCH IF NOT
011040  104012                              CRLF                    ;NEXT LINE
011042  005723                              TST     (R3)+           ;REPEAT COUNTER TO R3
011044  013701  001150              7$:     MOV     WIDTH,R1        ;COLUMN COUNT TO R1
011050  004737  007430              8$:     JSR     PC,SPC          ;SEND SPACE
011054  005301                              DEC     R1              ;DECREASE COLUMN COUNT
011056  001405                              BEQ     9$              ;BRANCH IF 0, END OF LINE
011060  012700  000100                      MOV     #100,R0         ;"@" TO R0
011064  104015                              PRINTC                  ;SEND IT
011066  005301                              DEC     R1              ;DECREASE COLUMN COUNT
011070  001367                              BNE     8$              ;BRANCH IF NOT 0 (NOT END)
011072  022703  000002              9$:     CMP     #2,R3           ;END OF LINE, FIRST TIME?
011076  001003                              BNE     11$             ;BRANCH IF NOT
011100  104022                      10$:    CR                      ;SEND CR
011102  005303                              DEC     R3              ;DECREASE LINE COUNTER
011104  000757                              BR      7$              ;REPEAT LINE
011106  005703                      11$:    TST     R3              ;TEST IF THIRD REPEAT
011110  001373                              BNE     10$             ;BRANCH IF NOT
011112  104012                              CRLF                    ;DO NEXT LINE
011114  005723                              TST     (R3)+           ;LINE REPEAT COUNTER TO R3
011116  013701  001150              12$:    MOV     WIDTH,R1        ;COLUMN COUNT TO R1
011122  012700  000046              13$:    MOV     #46,R0          ;"&" TO R0
011126  104015                              PRINTC                  ;SEND IT
011130  005301                              DEC     R1              ;DECREASE COLUMN COUNT
011132  001404                              BEQ     14$             ;BRANCH IF END
```

```
011134 004737 007430          JSR     PC,SPC          ;SEND SPACE
011140 005301                 DEC     R1              ;DECREASE COLUMN COUNT
011142 001367                 BNE     13$             ;BRANCH IF NOT END
011144 022703 000002   14$:   CMP     #2,R3           ;TEST IF FIRST TIME
011150 001003                 BNE     16$             ;BRANCH IF =2, FIRST TIME
011152 104022          15$:   CR                      ;SENT CR
011154 005303                 DEC     R3              ;DECREASE REPEAT COUNTER
011156 000757                 BR      12$             ;PRINT LINE AGAIN
011160 005703          16$:   TST     R3              ;TEST IF END, R3=0
011162 001373                 BNE     15$             ;BRANCH IF NOT END
011164 104012                 CRLF                    ;SEND CR,LF
011166 104005                 CHAIN                   ;CHAIN TO NEXT TEST
011170 000676                 BR      1$              ;REPEAT TEST
```

```
                                ;XXXXXXXXXX
                                ;
                                ;PT10--  PRINTING FREQUENCY TEST-- 120 H'S ARE PRINTED ON 4 LINES
                                ;        30 PER LINE. THE TEST IS SUCH THAT BETWEEN THE FIRST AND SECOND
                                ;        "H" A 30 MSEC DELAY IS INTRODUCED. THIS DELAY IS THEN INCREASED
                                ;        BETWEEN CHARACTERS OUT TO 60 CHARACTERS IN AN EXPONENTIAL
                                ;        MANNER. THE DELAY IS THEN DECREASED IN THE SAME MANNER OUT TO THE
                                ;        120TH CHARACTER. THIS DELAY IS CALCULATED AS FOLLOWS;
                                ;
                                ;                NEW DELAY = OLD DELAY [+ OR -] (OLD DELAY/16 + OLD DELAY/128 )
                                ;
                                ;XXXXXXXXXX

011172  000010                  PT10:   10                              ;TEST NUMBER
011174  011330                          PT11                            ;JEXT TEST
011176  104016                          PRTHDR                          ;TYPE MESSAGE
011200  012701  000036          1$:     MOV     #36,R1                  ;SET R1=30
011204  012702  000170                  MOV     #120.,R2                ;SET CHAR COUNT = 120
011210  012737  000036  011226          MOV     #30.,3$+2               ;SET UP DELAY VALUE
011216  012700  000110          2$:     MOV     #110,R0                 ;"H" TO R0
011222  104015                          PRINTC                          ;SEND IT
011224  012700  000036          3$:     MOV     #30.,R0
011230  104010                          DELAY                           ;DELAY
011232  005301                          DEC     R1                      ;DEC. COUNT OF CHARS PER LINE
011234  001426                          BEQ     6$                      ;BRANCH IF 0, END OF LINE
011236  005302                  4$:     DEC     R2                      ;DECREMENT CHAR COUNTER
011240  001430                          BEQ     7$                      ;BRANCH IF END
011242  013704  011226                  MOV     3$+2,R4                 ;GET OLD DELAY
011246  006204                          ASR     R4                      ;CAL 1/16 OF OLD DELAY
011250  006204                          ASR     R4
011252  006204                          ASR     R4
011254  006204                          ASR     R4
011256  010405                          MOV     R4,R5                   ;SAVE 1/16 IN R5
011260  006204                          ASR     R4                      ;CAL 1/128 OF OLD DELAY
011262  006204                          ASR     R4
011264  006204                          ASR     R4
011266  060405                          ADD     R4,R5                   ;1/16 +1/128 TO R5
011270  022702  000074                  CMP     #60.,R2                 ;TEST WHICH HALF OF THE 120 CHARS.
011274  003403                          BLE     5$                      ;BRANCH IF LT OR EQ 60
011276  160537  011226                  SUB     R5,3$+2                 ;GT 51( DECREASE DELAY BY 34 MEC.
011302  000745                          BR      2$                      ;GO PRINT AGAIN
011304  060537  011226          5$:     ADD     R5,3$+2                 ;LT HALF WAY, ADD DELAY OF 34 MEC.
011310  000742                          BR      2$                      ;GO PRINT AGAIN
011312  104012                  6$:     CRLF                            ;SEND CRLF
011314  012701  000036                  MOV     #36,R1                  ;SET R1=30
011320  000746                          BR      4$
011322  104012                  7$:     CRLF                            ;SEND CR,LF
011324  104005                          CHAIN                           ;CHAIN TO NEXT TEST
011326  000724                          BR      1$                      ;REPEAT TEST
```

```
                                ;XXXXXXXXXX
                                ;
                                ;PT11)- RIBBON FEED TEST-- THIS TEST PRINTS A SINGLE COLUMN OF X'S
                                ;       (24 LINES) DOWN THE LEFT MARGIN OF THE PAGE.
                                ;       VISUALLY CHECK THE RIBBON FEED MECHANISM FOR PROPER OPERATION.
                                ;
                                ;XXXXXXXXXX


011330  000011                  PT11:   11                      ;TEST NUMBER
011332  011362                          PT12                    ;NEXT TEST
011334  104016                          PRTHDR                  ;TYPE MESSAGE
011336  012701  000030          1$:     MOV     #30,R1          ;SET R1=24(10), LINE COUNT
011342  012700  000130          2 :     MOV     #130,R0         ;SET CHAR = X
011346  104015                          PRINTC                  ;PRINT X
011350  104012                          CRLF                    ;SEND CR-LF
011352  005301                          DEC     R1              ;DECREMENT LINE COUNT
011354  001372                          BNE     2$              ;CONTINUE IF NOT DONE TEST
011356  104005                          CHAIN                   ;CHAIN TO NEXT TEST
011360  000766                          BR      1$              ;REPEAT TEST



                                ;XXXXXXXXXX
                                ;
                                ;PT12-- PRINTER BELL TEST-- THE LAST TEST IN THE
                                ;       PRINTER TEST SEQUENCE.  THIS TEST OUTPUTS
                                ;       EIGHT BELL SIGNALS TO THE PRINTER
                                ;
                                ;
                                ;XXXXXXXXXX
                                ;
011362  000012                  PT12:   12                      ;THIS TEST
011364  007242                          PT0                     ;NEXT TEST
011366  104016                          PRTHDR                  ;TYPE HEADER
011370  012701  000010          PT12A:  MOV     #10,R1          ;COUNTER TO R1
011374  012700  000007                  MOV     #7,R0           ;BELL TO R0
011400  104015                  1$:     PRINTC                  ;SEND IT
011402  005301                          DEC     R1              ;DECREMENT COUNT
011404  001375                          BNE     1$              ;BRANCH IF NOT ZERO
011406  104014                          LF
011410  012700  003720                  MOV     #3720,R0        ;DELAY 2 SEC BEFORE RESTARTING
011414  104010                          DELAY
011416  013700  000042                  MOV     @#42,R0         ;CHECK IF UNDER ACT11 OR XXDP
011422  001405                          BEQ     HERE            ;CONTINUE TEST SEQUENCE
011424  000005                          RESET
011426  004710                  LOGICAL:JSR     PC,(R0)
011430  000240                          NOP
011432  000240                          NOP
011434  000240                          NOP
011436  104005                  HERE:   CHAIN                   ;CHAIN TO NEXT TEST
011440  000753                          BR      PT12A           ;REPEAT TEST
```

```
                                  ;XXXXXXXXXX
                                  ;
                                  ;PT17-- LIFE TEST
                                  ;       THIS TEST PRINTS 2 FULL LINES OF EACH PRINTABLE
                                  ;       CHARACTER AND OVERPRINTS THE SECOND LINE 4 TIMES.
                                  ;       THIS TEST IS CONTINUOUS RUNNING ONCE INITIATED,
                                  ;       LOOPING AUTOMATICALLY ON ITSELF.
                                  ;       END OB PASS COUNT IS CLEARED WHENEVER TEST IS RESTARTED
                                  ;
                                  ;XXXXXXXXXX

011442  000017                    PT17B:  17                      ;TEST NUMBER
011444  011442                            PT17B                   ;JEXT TEST
011446  000137  011470                    JMP     PT17D           ;CONTINUE
011452  000017                    PT17:   17                      ;TEST NUMBER
011454  011442                            PT17B                   ;NEXT TEST
011456  005037  011604                    CLR     PASCNT          ;CLEAR PASS COUNT
011462  104016                            PRTHDR                  ;PRINT HEADER
011464  104007                            TYPEM                   ;PRINT COLUMN # MESG
011466  016715                            HDRO
011470  012703  000041            PT17D:  MOV     #41,R3          ;SET START CHAR
011474  013701  001150            1$:     MOV     WIDTH,R1        ;SET COLUMN COUNT
011500  010300                    2$:     MOV     R3,R0           ;GET CHAR
011502  104015                            PRINTC                  ;SEND CHAR
011504  005301                            DEC     R1              ;JECREMENT COUNT
011506  001374                            BNE     2$              ;BRANCH IF NOT DONE
011510  104012                            CRLF                    ;SEND CR-LF
011512  012702  000005                    MOV     #5,R2           ;SET OVERPRINT COUNT
011516  013701  001150            3$:     MOV     WIDTH,R1        ;SET COLUMN COUNT
011522  010300                    4$:     MOV     R3,R0           ;GET CHAR
011524  104015                            PRINTC                  ;SEND CHAR
011526  005301                            DEC     R1              ;DECREMENT COUNT
011530  001374                            BNE     4$              ;BRANCH IF NOT DONE
011532  104022                            CR                      ;SEND CR
011534  005302                            DEC     R2              ;DONE OVERPRINTS?
011536  001367                            BNE     3$              ;NO, CONTINUE
011540  104014                            LF                      ;SEND LF
011542  005203                            INC     R3              ;SET NEXT CHAR
011544  022703  000177                    CMP     #177,R3         ;DONE CHAR SET?
011550  001351                            BNE     1$              ;NO, CONTINUE
011552  005237  011604                    INC     PASCNT          ;INC PASS COUNT
011556  012700  016632                    MOV     #PASMES,R0      ;SET MESG ADR
011562  013701  011604                    MOV     PASCNT,R1       ;# TO CONVERT
011566  012702  000004                    MOV     #4,R2           ;# DIGITS
011572  104023                            BTOASC                  ;CONVERT PASS COUNT TO ASCII
011574  104007                            TYPEM                   ;TYPE END OF PASS MESG
011576  016613                            ENDPAS
011600  104005                            CHAIN                   ;REPEAT TEST
011602  000732                            BR      PT17D


011604  000000                    PASCNT: 0
```

```
                                        .SBTTL  LA36 ECHO TESTS
                                ;
                                ;XXXXXXXXXX
                                ;
                                ;E020-- CHARACTER ECHO TEST-- ALL PRINTABLE AND
                                ;       NON-PRINTING CHARACTERS TYPED ON THE KEYBOARD
                                ;       ARE USED TO DRIVE THE PRINTER, ONE CHARACTER AT
                                ;       A TIME.  A "RUBOUT" WILL CAUSE THE TEST TO BE
                                ;       TERMINATED.
                                ;
                                ;XXXXXXXXXXX
                                ;
011606  000020                  E020:   20                      ;TEST NUMBER
011610  011656                          E021                    ;NEXT TEST
011612  104016                          PRTHDR                  ;TYPE HEADER
011614  104020                  1$:     READ                    ;GO WAIT FOR KEYBOARD INPUT
011616  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
011622  104010                          DELAY
011624  022737  000177  001176          CMP     #177,TEMPCH     ;CHECK IF RUBOUT
011632  001405                          BEQ     2$              ;BRANCH IF YES
011634  104017                          PRNT                    ;NO, CHECK PRINTER READY
011636  117777  167250  167252          MOVB    @TKB,@TPB       ;READY, ECHO CHARACTER
011644  000763                          BR      1$
011646  104007                  2$:     TYPEM                   ;PRINT TERMINATION MESSAGE
011650  017060                          ECOEND
011652  104005                          CHAIN                   ;CHAIN TO NEXT TEST
011654  000757                          BR      1$              ;REPEAT TEST
                                ;
                                ;XXXXXXXXXX
                                ;
                                ;E021-- LINE ECHO TEST, FAST RATE-- THIS TEST WILL
                                ;       CAUSE THE CONTINUAL PRINTING OF "0" AT THE MAXIMUM
                                ;       RATE UNTIL EITHER ANOTHER CHARACTER IS SELECTED
                                ;       BY PRESSING A KEY ON THE KEYBOARD OR TERMINATION BY THE
                                ;       RUBOUT.
                                ;
                                ;XXXXXXXXXX
                                ;
011656  000021                  E021:   21                      ;TEST NUMBER
011660  011714                          E022                    ;JEXT TEST
011662  104016                          PRTHDR                  ;TYPE HEADER
011664  012737  000060  001160  E021A:  MOV     #60,REPT        ;CHARACTER TO BE REPEATED (0)
011672  013702  001150          1$:     MOV     WIDTH,R2        ;SET COLUMN COUNT
011676  013700  001160          2$:     MOV     REPT,R0         ;GET CHAR
011702  104015                          PRINTC                  ;PRINT CHAR
011704  005302                          DEC     R2              ;DEC COLUMN COUNT
011706  003373                          BGT     2$              ;FINISH LINE
011710  104012                          CRLF                    ;SEND A CR AND LF
011712  000767                          BR      1$
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;E022-- LINE ECHO TEST, SLOW RATE-- SAME AS E021 EXCEPT
                                        ;       THAT A DELAY IS INTRODUCED BETWEEN CHARACTERS
                                        ;       TO PRODUCE A LCV ACTION
                                        ;
                                        ;XXXXXXXXXX
                                        ;
011714  000022                          E022:   22
011716  012166                                  E023
011720  104016                                  PRTHDR                  ;TYPE HEADER
011722  012737  000060  001160          E022A:  MOV     #60,REPT        ;LOAD 0 AS INITIAL CHARACTER
011730  013702  001150                  1$:     MOV     WIDTH,R2        ;SET COLUMN COUNT
011734  013700  001160                  2$:     MOV     REPT,R0         ;GET CHAR
011740  104015                                  PRINTC                  ;PRINT CHAR
011742  005302                                  DEC     R2              ;DEC COLUMN COUNT
011744  001404                                  BEQ     3$              ;BRANCH IF DONE LINE
011746  012700  003410                          MOV     #3410,R0
011752  104010                                  DELAY                   ;DELAY 1.8 SEC.
011754  000767                                  BR      2$              ;OUTPUT NEW CHAR.
011756  104012                          3$:     CRLF                    ;SEND A CR AND LF
011760  000763                                  BR      1$
```

```
                                        ;**********
                                        ;
                                        ;         THIS FOLLOWING TABLE IS USED BY TEST E023
                                        ;
                                        ;**********

011762  052516  020114                  MONIC:  .ASCII  /NUL /
011766  047523  020110                          .ASCII  /SOH /
011772  052123  020130                          .ASCII  /STX /
011776  052105  020130                          .ASCII  /ETX /
012002  047505  020124                          .ASCII  /EOT /
012006  047105  020121                          .ASCII  /ENQ /
012012  041501  020113                          .ASCII  /ACK /
012016  042502  020114                          .ASCII  /BEL /
012022  051502  020040                          .ASCII  /BS  /
012026  052110  020040                          .ASCII  /HT  /
012032  043114  020040                          .ASCII  /LF  /
012036  052126  020040                          .ASCII  /VT  /
012042  043106  020040                          .ASCII  /FF  /
012046  051103  020040                          .ASCII  /CR  /
012052  047523  020040                          .ASCII  /SO  /
012056  044523  020040                          .ASCII  /SI  /
012062  046104  020105                          .ASCII  /DLE /
012066  041504  020061                          .ASCII  /DC1 /
012072  041504  020062                          .ASCII  /DC2 /
012076  041504  020063                          .ASCII  /DC3 /
012102  041504  020064                          .ASCII  /DC4 /
012106  040516  020113                          .ASCII  /NAK /
012112  054523  020116                          .ASCII  /SYN /
012116  052105  020102                          .ASCII  /ETB /
012122  040503  020116                          .ASCII  /CAN /
012126  046505  020040                          .ASCII  /EM  /
012132  052523  020102                          .ASCII  /SUB /
012136  051505  020103                          .ASCII  /ESC /
012142  051506  020040                          .ASCII  /FS  /
012146  051507  020040                          .ASCII  /GS  /
012152  051522  020040                          .ASCII  /RS  /
012156  051525  020040                          .ASCII  /US  /
012162  050123  020040                          .ASCII  /SP  /

                                                .EVEN
```

```
                                        ;XXXXXXXXXX
                                        ;
                                        ;E023-- CHARACTER CODE TEST-- ANY CHARACTER SELECTED
                                        ;       WILL BE ECHOED ALONG WITH ITS OCTAL CODE.
                                        ;       A MNEMONIC WILL BE PRINTED INSTEAD OF THE CHARACTER
                                        ;       IF IT IS A NON-PRINTING CHARACTER.
                                        ;       THE PARITY OF THE RECEIVED CODE WILL ALSO BE
                                        ;       INDICATED AS EITHER EVEN OR ODD.
                                        ;
                                        ;XXXXXXXXXX
                                        ;
012166  000023                          E023:   23                              ;TEST NUMBER
012170  012510                                  E024                            ;NEXT TEST
012172  104016                                  PRTHDR                          ;TYPE HEADER
012174  104020                          1$:     READ                            ;GO WAIT FOR CHARACTER
012176  012700  000036                          MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
012202  104010                                  DELAY
012204  023727  001176  000041                  CMP     TEMPCH,#41              ;TEST IF CHAR IS PRINTABLE
012212  103015                                  BHIS    3$                      ;BRANCH IF IT IS
012214  004737  012350                          JSR     PC,STRLN                ;STORE CODE INTO MESSAGE
012220  113700  001176                          MOVB    TEMPCH,R0               ;GET CODE AGAIN
012224  006300                                  ASL     R0                      ;MULT BY 2
012226  006300                                  ASL     R0                      ;MULT BY 4
012230  062700  011762                          ADD     #MONIC,R0               ;ADD ADDR OF MNEMONIC TABLE
012234  004737  012426                          JSR     PC,MOVNUM               ;MOV MNEMONIC TO MESSAGE
012240  104000                          2$:     TYPE                            ;TYPE CODE AND MNEMONIC
012242  017107                                  E023M                           ;ADDRESS OF MESSAGE
012244  000753                                  BR      1$                      ;GO WAIT FOR NEXT CHARACTER
012246  023727  001176  000177          3$:     CMP     TEMPCH,#177             ;TEST IF CHAR IS A RUBOUT
012254  001421                                  BEQ     4$                      ;BRANCH IF RUBOUT
012256  012701  012500                          MOV     #MG24,R1
012262  113721  001176                          MOVB    TEMPCH,(R1)+
012266  112721  000040                          MOVB    #40,(R1)+
012272  112721  000040                          MOVB    #40,(R1)+
012276  112721  000040                          MOVB    #40,(R1)+
012302  004737  012350                          JSR     PC,STRLN                ;STORE CODE INTO MESSAGE
012306  012700  012500                          MOV     #MG24,R0                ;ADDR OF CHAR INTO R0
012312  004737  012426                          JSR     PC,MOVNUM               ;MOVE CHAR INTO MESSAGE
012316  000750                                  BR      2$                      ;TYPE MESSAGE
012320  004737  012350                  4$:     JSR     PC,STRLN                ;RUBOUT, CONVERT AND STOR CODE
012324  012700  012504                          MOV     #MG25,R0                ;ADDR. OF DEL INTO R0
012330  004737  012426                          JSR     PC,MOVNUM               ;MOVE DEL INTO MESSAGE
012334  104000                                  TYPE                            ;TYPE MESSAGE
012336  017107                                  E023M                           ;ADDR OF MESSAGE
012340  104007                                  TYPEM
012342  017060                                  ECOEND
012344  104005                                  CHAIN                           ;CHAIN TO NEXT TEST
012346  000712                                  BR      1$                      ;REPEAT TEST
012350  012702  000003                  STRLN:  MOV     #3,R2                   ;COUNT OF 3 TO R2
012354  012701  017111                          MOV     #LINES,R1               ;ADDR OF MESG TO R1
012360  062701  000003                          ADD     #3,R1                   ;POINT TO LAST SPACE IN MESG
012364  013700  001202                  1 :     MOV     PCHAR,R0                ;MOVE OCTAL CODE TO R0
012370  042700  177770                          BIC     #177770,R0              ;SAVE LS OCTAL CHAR.
012374  062700  000060                          ADD     #60,R0                  ;IAKE ASCII
```

```
012400  110041                          MOVB    R0,-(R1)        ;MOVE INTO MESG
012402  005302                          DEC     R2              ;DECREMENT CHAR COUNTER
012404  001407                          BEQ     2$              ;BRANCH IF 3 MOVED
012406  006237  001202                  ASR     PCHAR           ;NOT THREE, SHIFT NEXT OCTAL
012412  006237  001202                  ASR     PCHAR           ;CHARACTER TO THE RIGHT
012416  006237  001202                  ASR     PCHAR           ;
012422  000760                          BR      1$              ;CONVERT AND STORE NEXT CHAR.
012424  000207                  2$:     RTS     PC              ;RETURN TO CALLER
012426  012701  017115          MOVNUM: MOV     #LINESA,R1      ;ADDR OF LINES IN R1
012432  012702  000004                  MOV     #4,R2           ;COUNT OF 4 TO R2
012436  112021                  1$:     MOVB    (R0)+,(R1)+     ;MOV 4 CHARS TO MESG AREA
012440  005302                          DEC     R2              ;DECREMENT COUNTER
012442  001375                          BNE     1$              ;BRANCH IF NOT ALL DONE
012444  105737  001200                  TSTB    PARITY          ;TEST PARITY FLAG
012450  001003                          BNE     2$              ;BRANCH IF ODD PARITY
012452  012700  017157                  MOV     #EVEN,R0        ;SET ADDRESS FOR EVEN PARITY MESG
012456  000402                          BR      3$              ;CONTINUE
012460  012700  017163          2$:     MOV     #ODD,R0         ;SET ADDRESS FOR ODD PARITY MESG
012464  012702  000004          3$:     MOV     #4,R2           ;COUNT OF 4 TO R2
012470  112021                  4$:     MOVB    (R0)+,(R1)+     ;MOVE 4 CHARS TO MESG AREA
012472  005302                          DEC     R2              ;DECREMENT COUNTER
012474  001375                          BNE     4$              ;BRANCH IF NOT DONE
012476  000207                          RTS     PC              ;RETURN

012500  020040  020040          MG24:   .ASCII  /    /          ;SAVE CHARACTER CODE

                                        .EVEN

012504  042504  020114          MG25:   .ASCII  /DEL /          ;MNEMONIC FOR RUBOUT

                                        .EVEN
```

```
                                        ;XXXXXXXXXX
                                        ;E024--  SELECTED PATTERN ECHO TEST-- SELECT 1 TO 256
                                        ;        CHARACTERS.  EACH WILL BE ECHOED
                                        ;        AND STORED UNTIL THE CNTL/C IS SELECTED.
                                        ;        AT THAT TIME ALL CHARACTERS WILL BE PRINTED AS
                                        ;        A CONTINOUS STRING UNTIL EITHER THE RUBOUT IS
                                        ;        SELECTED TO TERMINATE OR THE CNTL/C IS SELECTED
                                        ;        AGAIN.  A TERMINATEING CNTL/C FOLLOWED BY ANOTHER
                                        ;        CNTL/C WILL ALWAYS CAUSE THE LAST INPUTTED STRING TO
                                        ;        BE PRINTED.  A TERMINATING CNTL/C FOLLOWED BY A CHARACTER OTHER THAN A
                                        ;        RUBOUT WILL CAUSE A NEW STRING TO BE INPUTTED.
                                        ;XXXXXXXXXXX

 012510  000024                         E024:   24                      ;TEST NUMBER
 012512  013256                                 E025                    ;NEXT TEST
 012514  104016                                 PRTHDR                  ;TYPE TEST HEADER
 012516  005001                         E024B:  CLR     R1              ;CLEAR CHARACTER COUNT
 012520  012702  012654                         MOV     #BUFR,R2        ;ADDRESS OF BUFFER TO R2
 012524  104020                         1$:     READ                    ;WAIT FOR INPUT
 012526  012700  000036                         MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
 012532  104010                                 DELAY
 012534  022737  000177  001176                 CMP     #177,TEMPCH     ;TEST IF RUBOUT
 012542  001440                                 BEQ     TERM            ;BRANCH IF RUBOUT
 012544  022737  000003  001176                 CMP     #3,TEMPCH       ;TEST IF .CNTL-C
 012552  001413                                 BEQ     OUTPUT          ;BRANCH IF CNTL-C
 012554  020127  000400                         CMP     R1,#256.        ;YES, CHECK IF CHAR CNT IS EQ, GT 256
 012560  103361                                 BHIS    1$              ;BRANCH IF YES, IGNORE CHAR
 012562  113722  001176                         MOVB    TEMPCH,(R2)+    ;STORE CHAR INTO BUFFER
 012566  005201                                 INC     R1              ;INCREMENT CHARACTER COUNT
 012570  104017                                 PRNT                    ;CHECK IF PRINTER READY
 012572  113777  001176  166316                 MOVB    TEMPCH,@TPB     ;ECHO CHAR
 012600  000751                                 BR      1$              ;GO WAIT FOR NEXT CHAR
                                        ;
                                        ;SECTION TO OUTPUT CONTINOUS STRING
                                        ;
 012602  020227  012654                 OUTPUT: CMP     R2,#BUFR        ;CHECK IF POINTER IS AT START OF TABLE
 012606  001403                                 BEQ     1$              ;YES, BRANCH
 012610  113722  001176                         MOVB    TEMPCH,(R2)+    ;NO, STORE ↑C IN TABLE
 012614  104013                                 SCRLF                   ;SEND A CR LF
 012616  012702  012654                 1$:     MOV     #BUFR,R2        ;BUFFER ADDRESS TO R2
 012622  021227  000003                         CMP     (R2),#3         ;CHECK IF FIRST CHAR IS ↑C
 012626  001733                                 BEQ     E024B           ;YES, LOOK FOR INPUT AGAIN
 012630  112200                         2$:     MOVB    (R2)+,R0        ;GET CHARACTER
 012632  020027  000003                         CMP     R0,#3           ;DONE STRING?
 012636  001767                                 BEQ     1$              ;YES, RESTART STRING
 012640  104015                                 PRINTC                  ;PRINT CHAR
 012642  000772                                 BR      2$              ;CONTINUE
 012644  104007                         TERM:   TYPEM                   ;OUTPUT TERMINATION MESSAGE
 012646  017060                                 ECOEND
 012650  104005                                 CHAIN                   ;CHAIN TO NEXT TEST
 012652  000721                                 BR      E024B           ;REPEAT TEST
 012654  000003                         BUFR:   3                       ;INITIALIZE FIRST CHAR AS CNTL-C IN TABLE
 012656  000400                                 .BLKB   256.            ;256 CHARACTER BUFFER
```

```
                                        ;XXXXXXXXXXX

                                        ;E025--  BELL ECHO TEST-- A MESSAGE IS PRINTED AND
                                        ;        THE TEST WAITS FOR SOME PRINTABLE CHARACTER
                                        ;        TO BE SELECTED ON THE KEYBOARD (GT040).  THIS
                                        ;        TEST IS VALID ONLY IF THE PAPER WIDTH IS GT 64
                                        ;        COLUMNS.  IF LT64 COLUMNS AN ILLEGAL BELL TEST
                                        ;        MESSAGE IS PRINTED.
                                        ;
                                        ;XXXXXXXXXXX

013256  000025                          E025:   25                      ;TEST NUMBER
013260  011606                                  E020                    ;NEXT TEST HEADER
013262  104016                                  PRTHDR                  ;PRINT HEADER
013264  023727  001150  000101          1$:     CMP     WIDTH,#101      ;TEST IF COLUMN COUNTIS EQ,GT 64
013272  103427                                  BLO     4$              ;BRANCH IF NOT
013274  104007                                  TYPEM                   ;TYPE TEST MESG
013276  016733                                  E025MA
013300  000402                                  BR      3$              ;WAIT FOR CHAR
013302  104000                          2$:     TYPE                    ;TYPE TEST MESG ON TERM CHAR RCVD ON
013304  016733                                  E025MA
013306  104020                          3$:     READ                    ;WAIT FOR OPERATOR RESPONSE
013310  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
013314  104010                                  DELAY
013316  023727  001176  000040                  CMP     TEMPCH,#40      ;TEST IF PRINTABLE
013324  103770                                  BLO     3$              ;BRANCH IF NON-PRINTABLE
013326  022737  000177  001176                  CMP     #177(TEMPCH     ;CHECK IF CHAR IS RUBOUT
013334  001410                                  BEQ     5$              ;BRANCH IF YES
013336  104017                                  PRNT                    ;CHECK IF PRINTER IS READY
013340  113777  001176  165550                  MOVB    TEMPCH,@TPB     ;PRINT CHAR. (BELL SHOULD SOUND)
013346  104013                                  SCRLF                   ;SEND A CRLF
013350  000754                                  BR      2$              ;REPEAT
013352  104007                          4$:     TYPEM                   ;TYPE ERROR MESSAGE
013354  017033                                  E025MB
013356  104007                          5$:     TYPEM                   ;PRINT TERMINATION
013360  017060                                  ECOEND
013362  104005                                  CHAIN                   ;EXIT TO NEXT TEST
013364  000737                                  BR      1$              ;REPEAT TEST
```

```
                                                .SBTTL  LA36 OPTION TESTS

                                        ;XXXXXXXXXXXX
                                        ;TEST30 -       SECONDARY CHARACTER SET OPTION
                                        ;XXXXXXXXXXXX

        013366  000030                  TEST30: 30
        013370  013374                          1$
        013372  104016                          PRTHDR                          ;PRINT TEST HEADER
        013374  012704  000010          1$:     MOV     #8.,R4                  ;SET PASS COUNT
        013400  104007                  2$:     TYPEM                           ;INDICATE PRIMARY SET & SEND "SI"
        013402  013464                          10$
        013404  012702  000177                  MOV     #177,R2                 ;SET END CHAR
        013410  004737  013476                  JSR     PC,30$                  ;PRINT CHAR SET
        013414  104007                          TYPEM                           ;IJDICATE SECONDARY CHAR SET
        013416  013471                          20$
        013420  013702  013546                  MOV     T30SC,R2                ;SET CHAR SET LIMIT
        013424  020227  000377                  CMP     R2,#377                 ;USING 8 BITS INSTEAD OF SI?
        013430  001403                          BEQ     3$                      ;BRANCH IF YES
        013432  012700  000016                  MOV     #16,R0                  ;SET SO CHAR
        013436  104015                          PRINTC                          ;SEND IT
        013440  004737  013476          3$:     JSR     PC,30$                  ;PRINT CHAR SET
        013444  104012                          CRLF                            ;BLANK LINE
        013446  005304                          DEC     R4                      ;DEC PASS COUNT
        013450  001353                          BNE     2$                      ;FINISH TEST
        013452  012700  000017                  MOV     #17,R0                  ;SET SO CHAR AGAIN
        013456  104015                          PRINTC                          ;MAKE SURE ON PRIMARY CHAR SET
        013460  104005                          CHAIN                           ;NEXT TEST SELECTION OR LOOP
        013462  000744                          BR      1$                      ;LOOP ON TEST

        013464  021417  036461     000  10$:    .ASCIZ  <17>/#1=/
        013471     017  031043  000075  20$:    .ASCIZ  <17>/#2=/

        013476  010201                  30$:    MOV     R2,R1                   ;GET LIMIT CHAR
        013500  042701  177537                  BIC     #177537,R1              ;GET START CHAR
        013504  013703  001150                  MOV     WIDTH,R3                ;GET COLUMN COUNT
        013510  162703  000003                  SUB     #3,R3                   ;SUBTRACT 3
        013514  010100                  31$:    MOV     R1,R0                   ;GET CHAR
        013516  104015                          PRINTC                          ;PRINT IT
        013520  005201                          INC     R1                      ;NEXT CHAR
        013522  020102                          CMP     R1,R2                   ;DONE CHAR SET?
        013524  001406                          BEQ     32$                     ;EXIT IF DONE
        013526  005303                          DEC     R3                      ;DEC COLUMN COUNT
        013530  001371                          BNE     31$                     ;FINISH LINE
        013532  104012                          CRLF                            ;CR-LF WHEN DONE LINE
        013534  013703  001150                  MOV     WIDTH,R3                ;RESET COLUMN COUNT
        013540  000765                          BR      31$                     ;CONTINUE
        013542  104012                  32$:    CRLF                            ;SEND CR-LF
        013544  000207                          RTS     PC                      ;RETURN

        013546  000177                  T30SC:  .WORD   177                     ;CHAR SET LIMIT
                                                                                ;CHANGE TO 377 WHEN USING 8 BIT CHAR SELECTION
```

```
                                ;XXXXXXXXXXXX
                                ;TEST31 -       SELECTIVE ADDRESSING OPTION
                                ;XXXXXXXXXXXX

013550  000031                  TEST31: 31
013552  013550                          TEST31
013554  104012                  7$:     CRLF                    ;SEND CR-LF FIRST
013556  013700  014356                  MOV     34$,R0          ;SET EOT CHAR
013562  104015                          PRINTC                  ;SEND IT
013564  104007                          TYPEM                   ;TRY PRINTING ERROR MESG
013566  014206                          10$
013570  012700  000007                  MOV     #7,R0           ;SET BEL CHAR
013574  104015                          PRINTC                  ;SEND IT
013576  012700  000002                  MOV     #2,R0           ;SET STX CHAR
013602  104015                          PRINTC                  ;SEND IT
013604  104016                          PRTHDR                  ;PRINT TEST HEADER ON ALL TERMINALS
013606  013700  014356                  MOV     34$,R0          ;SET EOT CODE
013612  104015                          PRINTC                  ;SEND IT
013614  012700  000002                  MOV     #2,R0           ;SET STX CODE
013620  104015                          PRINTC                  ;SEND IT
013622  104007                          TYPEM                   ;TRY PRINT ERROR MESG
013624  014206                          10$
013626  012703  014334                  MOV     #30$,R3         ;SET TABLE POINTER
013632  005713                  1 :     TST     (R3)            ;CHECK TABLE ENTRY
013634  001416                          BEQ     2$              ;NEXT PORTION OB TEST IF DONE
013636  013700  014356                  MOV     34$,R0          ;SEND EOT CHAR
013642  104015                          PRINTC
013644  011300                          MOV     (R3),R0         ;SEND GROUP SELECT CHAR
013646  104015                          PRINTC
013650  012700  000002                  MOV     #2,R0           ;SEND STX CHAR
013654  104015                          PRINTC
013656  104007                          TYPEM                   ;TYPE MESG
013660  014252                          14$
013662  012300                          MOV     (R3)+,R0        ;TYPE SELECT CHAR FOR MESG
013664  104015                          PRINTC
013666  104012                          CRLF                    ;CR-LF
013670  000760                          BR      1$              ;CONTINUE
013672  012703  014362          2$:     MOV     #40$,R3         ;SET TABLE ADR
013676  005713                  3$:     TST     (R3)            ;CHECK SELECT CHAR
013700  001526                          BEQ     4$              ;CONTINUE TEST
013702  013700  014356                  MOV     34$,R0          ;SET EOT CODE
013706  104015                          PRINTC                  ;SEND IT - ALL TERMINALS OFF
013710  011300                          MOV     (R3),R0         ;GET UNIQUE SELECT CHAR
013712  104015                          PRINTC                  ;SEND CHAR
013714  104007                          TYPEM                   ;TRY PRINTING ERROR MESG
013716  014206                          10$
013720  012700  000002                  MOV     #2,R0           ;SET STX CHAR
013724  104015                          PRINTC                  ;SEND IT
013726  104007                          TYPEM
013730  014260                          15$                     ;TYPE SELECT CHAR MESG
013732  011300                          MOV     (R3),R0         ;PRINT SELECT CHAR FOR MESG
013734  104015                          PRINTC
013736  104012                          CRLF
013740  104007                          TYPEM                   ;PRINT INSTRUCTIONS
```

```
013742  014277                          20$
013744  104020                          READ                    ;READ CHAR
013746  012700  000036                  MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
013752  104010                          DELAY
013754  022737  000177  001176          CMP     #177,TEMPCH     ;CHECK CHAR
013762  001021                          BNE     5$              ;CONTINUE IF NOT  RUBOUT
013764  013700  014356                  MOV     34$,R0          ;ENABLE ALL LINES
013770  104015                          PRINTC
013772  012700  000007                  MOV     #7,R0
013776  104015                          PRINTC
014000  012700  000002                  MOV     #2,R0
014004  104015                          PRINTC
014006  032737  000001  001132          BIT     #1,CNTLSW       ;WHICH CONTROL?
014014  001402                          BEQ     6$
014016  000137  002336                  JMP     TTY1B           ;GO TO KYBD CONTROL
014022  000137  002154          6$:     JMP     WAITF           ;GO TO SW REG CONTROL
014026  104017                  5$:     PRNT                    ;READY?
014030  013777  001176  165060          MOV     TEMPCH,@TPB     ;ECHO CHAR
014036  104012                          CRLF                    ;SEND CR-LF
014040  013700  014356                  MOV     34$,R0          ;SEND EOT
014044  104015                          PRINTC
014046  013700  014360                  MOV     35$,R0          ;SEND DUMMY SELECT CHAR
014052  104015                          PRINTC
014054  012700  000002                  MOV     #2,R0           ;SEND STX CHAR
014060  104015                          PRINTC
014062  104007                          TYPEM                   ;TRY PRINTING ERROR MESG
014064  014206                          10$
014066  012700  000003                  MOV     #3,R0           ;SEND ETX
014072  104015                          PRINTC
014074  011300                          MOV     (R3),R0         ;SEND UNIQUE SELECT CHAR
014076  104015                          PRINTC
014100  012700  000002                  MOV     #2,R0           ;SEND STX
014104  104015                          PRINTC
014106  104007                          TYPEM                   ;PRINT MESG ON SELECTED TERMINAL
014110  014260                          15$
014112  011300                          MOV     (R3),R0         ;PRINT SELECT CHAR
014114  104015                          PRINTC
014116  104012                          CRLF                    ;SEND CR-LF
014120  012700  000003                  MOV     #3,R0           ;SEND ETX
014124  104015                          PRINTC
014126  013700  014360                  MOV     35$,R0          ;SEND DUMMY SELECT CHAR
014132  104015                          PRINTC
014134  012700  000002                  MOV     #2,R0           ;SEND STX
014140  104015                          PRINTC
014142  104007                          TYPEM                   ;PRINT MESG ON SELECTED TERMINALS
014144  014260                          15$
014146  012300                          MOV     (R3)+,R0        ;PRINT SELECT CHAR
014150  104015                          PRINTC
014152  104012                          CRLF                    ;SEND CR-LF
014154  000650                          BR      3$              ;CONTINUE
014156  013700  014356          4$:     MOV     34$,R0          ;ENABLE ALL LINES
014162  104015                          PRINTC                  ;BEFORE EXITING TEST
014164  012700  000007                  MOV     #7,R0
014170  104015                          PRINTC
```

```
014172  012700  000002                  MOV     #2,R0
014176  104015                          PRINTC
014200  104005                          CHAIN                           ;NEXT TEST SELECTION OR LOOP
014202  000137  013554                  JMP     7$                      ;LOOP ON TEST

014206  051105  047522  026122  10$:    .ASCIZ  /ERROR, ALL TERMINALS SHOULD BE OFF/<ACRLF>
014214  040440  046114  052040
014222  051105  044515  040516
014230  051514  051440  047510
014236  046125  020104  042502
014244  047440  043106  000200
014252  051107  052517  020120  14$:    .ASCII  /GROUP /
014260  042523  042514  052103  15$:    .ASCIZ  /SELECT CHAR = /
014266  041440  040510  020122
014274  020075     000
014277     124  050131  020105  20$:    .ASCIZ  /TYPE ANY PRINTABLE CHAR ... /
014304  047101  020131  051120
014312  047111  040524  046102
014320  020105  044103  051101
014326  027040  027056  000040
                                        .EVEN
014334  000107                  30$:    107                             ;GROUP SELECT CHAR TABLE
014336  000000                          0                               ;FIRST ZERO = END OF TABLE
014340  000000                          0
014342  000000                          0
014344  000000                          0
014346  000000                          0
014350  000000                          0
014352  000000                          0
014354  000000                          0

014356  000004                  34$:    004                             ;DESELECT CHAR = "EOT"
014360  000045                  35$:    45                              ;DUMMY SELECT CODE = %
                                                                        ;IF % IS USED AS A GROUP OR UNIQUE
                                                                        ;SELECT CODE, REPLACE WITH ASCII
                                                                        ;CODE OF ANY UNUSED CHARACTER.

014362  000125                  40$:    125                             ;UNIQUE SELECT CHAR TABLE
014364  000000                          0                               ;FIRST ZERO - END OF TABLE
014366  000000                          0
014370  000000                          0
014372  000000                          0
014374  000000                          0
014376  000000                          0
014400  000000                          0
014402  000000                          0
014404  000000                          0
014406  000000                          0
014410  000000                          0
014412  000000                          0
014414  000000                          0
014416  000000                          0
014420  000000                          0
014422  000000                          0
```

```
                                     ;XXXXXXXXXXXX
                                     ;TEST32 -     AUTO ANSWER BACK OPTION
                                     ;XXXXXXXXXXXX

014424  000032                       TEST32: 32
014426  014432                               1$
014430  104016                               PRTHDR                  ;PRINT TEST HEADER
014432  105777  167556               1$:     TSTB    @TPSS           ;SEND ENQ TO SELECTING TERMINAL
014436  100375                               BPL     1$              ;OR CONSOLE TERMINAL
014440  012777  000005  167550               MOV     #5,@TPBS
014446  004737  014476                       JSR     PC,10$          ;READ AND PRINT MESG
014452  104007                               TYPEM                   ;TYPE INSTRUCTION
014454  014604                               20$
014456  004737  014476                       JSR     PC,10$          ;READ AND PRINT MESG
014462  104007                               TYPEM                   ;TYPE INSTRUCTION
014464  014631                               30$
014466  004737  014476                       JSR     PC,10$          ;READ & ECHO MESG
014472  104005                       2$:     CHAIN                   ;CHAIN TO NEXT TEST
014474  000756                               BR      1$              ;LOOP ON TEST

014476  012702  014655               10$:    MOV     #STORE,R2       ;SET TABLE ADR
014502  104020                               READ                    ;READ FIRST CHAR
014504  023727  001176  000177       16$:    CMP     TEMPCH,#177     ;CHAR = RUBOUT?
014512  001010                               BNE     11$             ;CONTINUE IF NOT RUBOUT
014514  032737  000001  001132               BIT     #1,CNTLSW       ;WHICH CONTROL?
014522  001402                               BEQ     17$
014524  000137  002336                       JMP     TTY1B           ;GO TO KYBD CONTROL
014530  000137  002154               17$:    JMP     WAITF           ;GO TO SW REG CONTROL
014534  113722  001176               11$:    MOVB    TEMPCH,(R2)+    ;STORE CHAR
014540  012700  000310                       MOV     #200.,R0        ;SET DELAY COUNTS FOR 200 MSEC.
014544  013701  001170               12$:    MOV     TIMER,R1
014550  105777  164334               13$:    TSTB    @TKS            ;ANY INPUT?
014554  100411                               BMI     14$             ;READ CHAR IF INPUT
014556  005301                               DEC     R1              ;WAIT FOR CHAR
014560  001373                               BNE     13$
014562  005300                               DEC     R0
014564  001367                               BNE     12$
014566  105012                       15$:    CLRB    (R2)            ;SET NULL AS TERMINATOR IN TABLE
014570  104000                               TYPE                    ;TYPE MESG ON TERMINAL RECEIVED ON
014572  014654                               STORE-1
014574  104013                               SCRLF
014576  000207                               RTS     PC              ;RETURN TO TEST
014600  104025                       14$:    READC                   ;READ CHAR
014602  000740                               BR      16$             ;CHECK CHAR & CONTINUE

014604  042504  051120  051505       20$:    .ASCIZ  /DEPRESS HERE IS KEY/<ACRLF>
014612  020123  042510  042522
014620  044440  020123  042513
014626  100131     000
014631     104  050105  042522       30$:    .ASCIZ  /DEPRESS CONTROL-E/<ACRLF>
014636  051523  041440  047117
014644  051124  046117  042455
014652  000200
014654     200                               .ASCII  <ACRLF>
```

```
 2748 014655 000025          STORE: .BLKB  21.          ;20 CHAR + TERMINATOR BUFFER
 2749                               .EVEN
```

```
                                        ;XXXXXXXXXXXX
                                        ;TEST33 -       TOP OF FORM OPTION
                                        ;XXXXXXXXXXXX

014702  000033                          TEST33: 33
014704  014706                                  1$
014706  104016                          1$:     PRTHDR                  ;PRINT TEST HEADER
014710  012705  015406                          MOV     #50$,R5         ;SET TABLE POINTER
014714  012704  015372                          MOV     #60$,R4         ;SET TABLE POINTER
014720  104007                                  TYPEM                   ;PRINT INSTR
014722  015214                                  40$
014724  012701  000020                          MOV     #16.,R1         ;SET LF COUNT TO 16
014730  104020                                  READ                    ;WAIT FOR KYBD FLAG
014732  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
014736  104010                                  DELAY
014740  022737  000177  001176                  CMP     #177,TEMPCH     ;CHECK FOR RUBOUT
014746  001010                                  BNE     8$              ;EXIT IF RUBOUT
014750  032737  000001  001132                  BIT     #1,CNTLSW       ;WHICH CONTROL?
014756  001402                                  BEQ     11$
014760  000137  002336                          JMP     TTY1B           ;GO TO KYBD CONTROL
014764  000137  002154                  11$:    JMP     WAITF           ;SW REG CONTROL
014770  104022                          8$:     CR                      ;SEND CR
014772  104014                          2$:     LF                      ;SEND LF
014774  005301                                  DEC     R1              ;DEC COUNT
014776  001375                                  BNE     2$              ;CONTINUE
015000  012700  000014                          MOV     #14,R0          ;SET FF
015004  104015                                  PRINTC                  ;SENT IT
015006  005000                                  CLR     R0              ;SET NULL
015010  113701  015372                          MOVB    60$,R1          ;SET FILL COUNT
015014  104015                          3$:     PRINTC                  ;SEND FILL
015016  005301                                  DEC     R1              ;DEC FILL COUNT
015020  001375                                  BNE     3$              ;CONTINUE
015022  104007                                  TYPEM                   ;TYPE MESSAGE
015024  015333                                  45$
015026  010537  015122                  4$:     MOV     R5,6$           ;SET MESG
015032  104020                                  READ                    ;WAIT FOR KYBD FLAG
015034  012700  000036                          MOV     #30.,R0         ;DELAY FOR HALF DUPLEX
015040  104010                                  DELAY
015042  022737  000177  001176                  CMP     #177,TEMPCH     ;CHECK CHAR
015050  001010                                  BNE     9$              ;EXIT IF RUBOUT
015052  032737  000001  001132                  BIT     #1(CNTLSW       ;WHICH CONTROL?
015060  001402                                  BEQ     12$
015062  000137  002336                          JMP     TTY1B
015066  000137  002154                  12 :    JMP     WAITF           ;SW REG CONTROL
015072  104022                          9 :     CR                      ;SEND CR
015074  012700  000014                          MOV     #14,R0          ;SEND FF
015100  104015                                  PRINTC
015102  112401                                  MOVB    (R4)+,R1        ;GET FILL COUNT
015104  005000                                  CLR     R0              ;SET NULL
015106  104015                          5$:     PRINTC                  ;SEND FILL
015110  005301                                  DEC     R1              ;DEC COUNT
015112  001375                                  BNE     5$              ;CONTINUE
015114  104007                                  TYPEM                   ;PRINT MESG
015116  015166                                  15$
```

```
      015120  104007                          TYPEM
      015122  000000                  6$:     .WORD   0
      015124  104007                          TYPEM
      015126  015174                          20$
      015130  022525                          CMP     (R5)+,(R5)+             ;INC TABLE POINTER
      015132  010537  015140                  MOV     R5,7$                   ;SET MESG
      015136  104007                          TYPEM                           ;PRINT MESG
      015140  000000                  7$:     .WORD   0
      015142  104007                          TYPEM
      015144  015204                          30$
      015146  104007                          TYPEM
      015150  015166                          15$
      015152  020527  015462                  CMP     R5,#55$                 ;DONE TEST?
      015156  001323                          BNE     4$                      ;NO, CONTINUE
      015160  104012                          CRLF                            ;CR-LF
      015162  104005                  10$:    CHAIN                           ;SELECT TEST OF LOOP
      015164  000650                          BR      1$                      ;LOOP ON TEST
```

```

015166 026455 026455 000055 15$:   .ASCIZ  /-----/
015174 020042 043106 027440 20$:   .ASCIZ  /" FF /<57>/ /
015202 000040
015204 020042 042516 052130 30$:   .ASCIZ  /" NEXT /
015212 000040
015214 051120 051505 020123 40$:   .ASCII  /PRESS TOP OF FORM RESET SWITCH/<ACRLF>/AFTER EACH SWITCH SETTING/<ACRLF
015222 047524 020120 043117
015230 043040 051117 020115
015236 042522 042523 020124
015244 053523 052111 044103
015252 040600 052106 051105
015260 042440 041501 020110
015266 053523 052111 044103
015274 051440 052105 044524
015302 043516    200
015305    124 050131 020105        .ASCII  /TYPE SPACE WHEN READY/<ACRLF>
015312 050123 041501 020105
015320 044127 047105 051040
015326 040505 054504    200
015333    055 026455 020055 45$:   .ASCIZ  /---- SET  3 INCH FORMFEED ----/
015340 042523 020124 031440
015346 044440 041516 020110
015354 047506 046522 042506
015362 042105 026440 026455
015370 000055
015372    002                 60$:   .BYTE   2
015373    005    010    021        .BYTE   5,8.,17.,20.,26.,32.,35.,50.,56.,68.
015376    024    032    040
015401    043    062    070
015404    104
       015406                      .EVEN

015406 020040 000063        50$:   .ASCIZ  /  3/
015412 027063 000065               .ASCIZ  /3.5/
015416 020040 000064               .ASCIZ  /  4/
015422 027065 000065               .ASCIZ  /5.5/
015426 020040 000066               .ASCIZ  /  6/
015432 020040 000067               .ASCIZ  /  7/
015436 020040 000070               .ASCIZ  /  8/
015442 027070 000065               .ASCIZ  /8.5/
015446 030440 000061               .ASCIZ  / 11/
015452 030440 000062               .ASCIZ  / 12/
015456 030440 000064               .ASCIZ  / 14/
015462 020040 000040        55$:   .ASCIZ  /   /
```

```
                                  ;XXXXXXXXXXXX
                                  ;TEST34 - HORIZONTAL TAB OPTION
                                  ;XXXXXXXXXXXX

015466  000034                    TEST34: 34
015470  015474                            1$
015472  104016                            PRTHDR                          ;PRINT TEST HEADER
015474  005004                    1$:     CLR     R4                      ;SET TABLE POINTER
015476  012737  015672  015670            MOV     #11$,10$+2              ;RESET JUMP INSTR FOR FIRST TIME THRU
015504  013703  001150            2$:     MOV     WIDTH,R3                ;SET COLUMN COUNT
015510  012700  000033                    MOV     #33,R0                  ;CLEAR OLD TABS
015514  104015                            PRINTC
015516  012700  000062                    MOV     #62,R0
015522  104015                            PRINTC
015524  104022                            CR                              ;DO A CARRIAGE RETURN
015526  016401  015760            3$:     MOV     20$(R4),R1              ;GET SPACE COUNT FOR TAB
015532  000405                            BR      5$                      ;SUBTRACT 1 FOR TAB SET
015534  012700  000040            4$:     MOV     #40,R0                  ;GET SPACE
015540  104015                            PRINTC                          ;SEND IT
015542  005303                            DEC     R3                      ;DEC COLUMN COUNT
015544  001420                            BEQ     6$                      ;CR IF DONE LINE
015546  005301                    5$:     DEC     R1                      ;DEC SPACE COUNT
015550  001371                            BNE     4$                      ;CONTINUE IF NOT DONE
015552  012700  000033                    MOV     #33,R0                  ;SEND IT
015556  104015                            PRINTC
015560  012700  000061                    MOV     #61,R0                  ;SEND 1, -SET TAB
015564  104015                            PRINTC
015566  012700  000010                    MOV     #10,R0                  ;SET BACKSPACE
015572  104015                            PRINTC                          ;SEND IT
015574  012700  000117                    MOV     #'O,R0                  ;PRINT O
015600  104015                            PRINTC
015602  005303                            DEC     R3                      ;DEC COLUMN COUNT
015604  001350                            BNE     3$                      ;CONTINUE IF NOT DONE LINE
015606  104022                    6$:     CR                              ;SEND CR
015610  013703  001150                    MOV     WIDTH,R3                ;RESET COLUMN COUNT
015614  016401  015760            7$:     MOV     20$(R4),R1              ;RESET COLUMN COUNT FOR TAB
015620  020127  000001                    CMP     R1,#1                   ;ADD 1 IF FIRST LINE
015624  001001                            BNE     8$
015626  005201                            INC     R1
015630  016402  016004            8$:     MOV     30$(R4),R2              ;SET FILL CHAR COUNT
015634  160103                            SUB     R1,R3                   ;SUBTRACT TAB FROM COLUMN COUNT
015636  002413                            BLT     10$                     ;BRANCH IF TOO MANY COLUMNS
015640  012700  000011                    MOV     #11,R0                  ;SET TAB
015644  104015                            PRINTC                          ;SEND IT
015646  005000                            CLR     R0                      ;YES, SET NULL
015650  104015                    9$:     PRINTC                          ;SEND FILL CHAR
015652  005302                            DEC     R2                      ;DEC FILL COUNT
015654  001375                            BNE     9$                      ;CONTINUE FILLS
015656  012700  000130                    MOV     #'X,R0                  ;SET X CHARACTER
015662  104015                            PRINTC                          ;SEND IT
015664  000761                            BR      8$                      ;CONTINUE
015666  000137  015672            10$:    JMP     @#11$                   ;SKIP FOLLOWING AFTER FIRST TIME
015672  012737  015720  015670    11$:    MOV     #12$,10$+2
015700  104022                            CR                              ;SEND CR
```

```
015702  012700  000130                   MOV     #'X,R0             ;PRINT X
015706  104015                           PRINTC
015710  013703  001150                   MOV     WIDTH,R3           ;RESET COLUMN COUNT
015714  005303                           DEC     R3                 ;SUBTRACT ONE FOR X
015716  000736                           BR      7$                 ;CONTINUE
015720  104012                   12$:    CRLF                       ;SEND CR-LF
015722  005724                           TST     (R4)+              ;INC TABLE POINTER
015724  016401  015760                   MOV     20$(R4),R1         ;GET COLUMN COUNT FOR TAB
015730  001403                           BEQ     13$                ;EXIT IF DONE TABLE (0)
015732  020137  001150                   CMP     R1,WIDTH           ;CHECK IF TOO LARGE
015736  101662                           BLOS    2$                 ;CONTINUE TEST, OK
015740  012700  000033           13$:    MOV     #33,R0             ;CLEAR HORIZONTAL TABS
015744  104015                           PRINTC
015746  012700  000062                   MOV     #62,R0
015752  104015                           PRINTC
015754  104005                           CHAIN                      ;SELECT TEST OR LOOP
015756  000646                           BR      1$                 ;LOOP ON TEST

015760  000001  000002  000004   20$:    .WORD   1,2,4,8.,16.,32.,64.,128.,132.,0
015766  000010  000020  000040
015774  000100  000200  000204
016002  000000
016004  000001  000002  000003   30$:    .WORD   1,2,3,5,9.,18.,36.,71.,73.,0
016012  000005  000011  000022
016020  000044  000107  000111
016026  000000
```

```
                                        ;XXXXXXXXXXXXXXX
                                        ;TEST35 -        VERTICAL TAB OPTION
                                        ;XXXXXXXXXXXXXXX

016030  000035                          TEST35: 35
016032  016034                                  1$
016034  104016                          1$:     PRTHDR                          ;PRINT TEST HEADER
016036  104007                                  TYPEM                           ;TYPE INSTR
016040  016365                                  20$
016042  104020                                  READ                            ;WAIT FOR KYBD FLAG
016044  012700  000036                          MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
016050  104010                                  DELAY
016052  022737  000177  001176                  CMP     #177,TEMPCH             ;CHECK CHAR
016060  001505                                  BEQ     8$                      ;EXIT IF RUBOUT
016062  005004                                  CLR     R4                      ;SET LINE COUNT
016064  012700  000033                          MOV     #33,R0                  ;CLEAR VERTICAL TABS
016070  104015                                  PRINTC
016072  012700  000064                          MOV     #64,R0
016076  104015                                  PRINTC
016100  104007                          2$:     TYPEM                           ;TYPE REF LINE
016102  016343                                  15$
016104  005204                                  INC     R4                      ;INC LINE COUNT
016106  020427  000013                          CMP     R4,#13                  ;CHECK IT
016112  003013                                  BGT     4$                      ;BRANCH IF DONE REF.
016114  010401                                  MOV     R4,R1                   ;GET LF COUNT
016116  104014                          3$:     LF                              ;SEND LF
016120  005301                                  DEC     R1                      ;DEC COUNT
016122  001375                                  BNE     3$                      ;CONTINUE
016124  012700  000033                          MOV     #33,R0                  ;SET TAB FOR THIS LINE
016130  104015                                  PRINTC
016132  012700  000063                          MOV     #63,R0
016136  104015                                  PRINTC
016140  000757                                  BR      2$                      ;CONTINUE
016142  012700  000014                  4$:     MOV     #14,R0                  ;SEND FF
016146  104015                                  PRINTC
016150  104007                                  TYPEM                           ;TYPE MESG
016152  016457                                  30$
016154  104020                                  READ                            ;WAIT FOR KYBD FLAG
016156  012700  000036                          MOV     #30.,R0                 ;DELAY FOR HALF DUPLEX
016162  104010                                  DELAY
016164  022737  000177  001176                  CMP     #177,TEMPCH             ;CHAR = RUBOUT?
016172  001440                                  BEQ     8$                      ;EXIT IF RUBOUT
016174  005004                                  CLR     R4                      ;RESET LF COUNT
016176  104007                          5$:     TYPEM                           ;TYPE REF LINE
016200  016342                                  10$
016202  005204                                  INC     R4                      ;INC LINE COUNT
016204  020427  000013                          CMP     R4,#13                  ;CHECK IT
016210  003014                                  BGT     7$                      ;BRANCH IF DONE
016212  012700  000013                          MOV     #13,R0                  ;SEND TAB
016216  104015                                  PRINTC
016220  010401                                  MOV     R4,R1                   ;SET FILL COUNT
016222  162701  000020                          SUB     #16.,R1                 ;SUBTRACT 16
016226  003763                                  BLE     5$                      ;SKIP NULLS IF COUNT <0
016230  005000                                  CLR     R0                      ;SET NULL CHAR
```

```
016232  104015                      6$:   PRINTC                         ;SEND FILLS
016234  005301                            DEC    R1                      ;DEC COUNT
016236  001375                            BNE    6$
016240  000756                            BR     5$                      ;CONTINUE
016242  012700  000033              7$:   MOV    #33,R0                  ;CLEAR VERTICAL TABS
016246  104015                            PRINTC
016250  012700  000064                    MOV    #64,R0
016254  104015                            PRINTC
016256  012700  000014                    MOV    #14,R0                  ;SEND FF
016262  104015                            PRINTC
016264  104012                            CRLF                           ;SEND CARRIAGE RETURN - LINE FEED
016266  104014                            LF                             ;SEND BLANK LINE
016270  104005                            CHAIN                          ;NEXT TEST OR REPREAT TEST
016272  000660                            BR     1$                      ;REPEAT TEST
016274  012700  000033              8$:   MOV    #33,R0                  ;CLEAR TABS
016300  104015                            PRINTC
016302  012700  000064                    MOV    #64,R0
016306  104015                            PRINTC
016310  012700  000014                    MOV    #14,R0                  ;SEND FF
016314  104015                            PRINTC
016316  104012                            CRLF
016320  104014                            LF
016322  032737  000001  001132            BIT    #1,CNTLSW               ;WHICH CONTROL
016330  001402                            BEQ    9$
016332  000137  002336                    JMP    TTY1B                   ;GO TO KYBD CONTROL
016336  000137  002154              9$:   JMP    WAITF                   ;SW REG CONTROL

016342     040                      10$:  .ASCII <40>
016343     055  026455  026455      15$:  .ASCIZ /--------<<<<<<<</<15>
016350  026455  036055  036074
016356  036074  036074  006474
016364     000
016365     123  052105  030440      20$:  .ASCII /SET 14 INCH FORM FEED/<ACRLF>
016372  020064  047111  044103
016400  043040  051117  020115
016406  042506  042105     200
016413     104  050105  042522            .ASCIZ /DEPRESS TOP OF FORM RESET SWITCH/<ACRLF><12><12>
016420  051523  052040  050117
016426  047440  020106  047506
016434  046522  051040  051505
016442  052105  051440  044527
016450  041524  100110  005012
016456     000
016457     001  000401  042522      30$:  .ASCIZ <1><1><1>/REMOVE /<57>/ RESET REF/
016464  047515  042526  027440
016472  051040  051505  052105
016500  051040  043105     000
```

```
                                                    .SBTTL  MISC. DIAGNOSTIC MESSAGES

        016505     200  046417  044501    STARTM:  .ASCII  <ACRLF><17>/MAINDEC-11-DZLAC-C/<ACRLF>
        016512  042116  041505  030455
        016520  026461  055104  040514
        016526  026503  100103
        016532  040514  033063  052040             .ASCII  /LA36 TERMINAL DIAGNOSTIC/<ACRLF>
        016540  051105  044515  040516
        016546  020114  044504  043501
        016554  047516  052123  041511
        016562     200
        016563     104  030514  020061             .ASCIZ  /DL11 & KL11 INTERFACE/<ACRLF><12>
        016570  020046  046113  030461
        016576  044440  052116  051105
        016604  040506  042503  005200
        016612     000
        016613     200  042412  042116    ENDPAS:  .ASCII  <ACRLF><12>/END OF PASS  /
        016620  047440  020106  040520
        016626  051523  020040
        016632  030060  030060  005200    PASMES:  .ASCIZ  /0000/<ACRLF><12>
        016640     000
        016641     200  047503  051516    DL11S:   .ASCII  <ACRLF>/CONSOLE & /
        016646  046117  020105  020046
        016654  030060  042040  030510    DL11S1:  *ASCIZ  /00 DL11'S UNDER TEST/<ACRLF><12>
        016662  023461  020123  047125
        016670  042504  020122  042524
        016676  052123  005200     000
        016703     200  005017  042524    HDRMSG:  .ASCIZ  <ACRLF><17><12>+TEST #/
        016710  052123  021440     000
        016715     060  030060  041440    HDR0:    .ASCIZ  /000 CKLUMNS/<ACRLF><12>
        016722  046117  046525  051516
        016730  005200     000
        016733     124  050131  020105    E025MA:  .ASCII  /TYPE ANY PRINTABLE CHARACTER /
        016740  047101  020131  051120
        016746  047111  040524  046102
        016754  020105  044103  051101
        016762  041501  042524  020122
        016770  047101  020104  044514             .ASCIZ  /AND LISTEN FOR BELL.............../
        016776  052123  047105  043040
        017004  051117  041040  046105
        017012  027114  027056  027056
        017020  027056  027056  027056
        017026  027056  027056     000
        017033     200  047516  020124    E025MB:  .ASCIZ  <ACRLF>/NOT ENOUGH COLUMNS/<ACRLF>
        017040  047105  052517  044107
        017046  041440  046117  046525
        017054  051516  000200
        017060  042600  044103  020117    ECOEND:  .ASCIZ  <ACRLF>/ECHO TEST TERMINATED/<ACRLF>
        017066  042524  052123  052040
        017074  051105  044515  040516
        017102  042524  100104     000
        017107     040     040            E023M:   .ASCII  /  /
        017111     040  020040     040    LINES:   .ASCII  /    /         ;MESG FOR TEST E024
        017115     040  020040  020040    LINESA:  .ASCIZ  /       /<ACRLF>
```

```
017122  020040  100040     000
017127     200  005017  042523  MESG3:  .ASCIZ  <ACRLF><17><12>/SELECT TEST NUMBER  /
017134  042514  052103  052040
017142  051505  020124  052516
017150  041115  051105  020040
017156     000
017157     105  042526     116  EVEN:   .ASCII  /EVEN/
017163     117  042104     040  ODD:    .ASCII  /ODD /

        000001                          .END
```

ACRLF =	000200	ADTENP	004312	AREAD =	104021	AT0	005520
ATDX	005522	AT1	005552	AT10	006170	AT11	006216
AT12	006256	AT13	006330	AT14	006410	AT15	006474
AT16	006574	AT17	006632	AT2	005604	AT20	006672
AT21	006754	AT22	007044	AT23	007142	AT3	005636
AT4	005670	AT5	005760	AT6	006030	AT7	006120
BIT0 =	000001	BIT1 =	000002	BIT10 =	002000	BIT11 =	004000
BIT12 =	010000	BIT13 =	020000	BIT14 =	040000	BIT15 =	100000
BIT2 =	000004	BIT3 =	000010	BIT4 =	000020	BIT5 =	000040
BIT6 =	000100	BIT7 =	000200	BIT8 =	000400	BIT9 =	001000
BRCTR	001162	BTOASC=	104023	BUFR	012654	CHAIN =	104005
CHAINN	001666	CHAINY	002010	CHALT =	104006	CHLT	001214
CNTLSW	001132	CNVCTR	004304	CONADD	001102	CONIT	004122
CONSET	004126	CONVEC	001104	COUNT3	001164	CR =	104022
CRBUF	001144	CRLF =	104012	CTRA	001146	CURTST	001174
DELAY =	104010	DIGIT	004306	DISPRE	000174	DLADR	001106
DLCNT	001154	DLY	003650	DL11S	016641	DL11S1	016654
ECOEND	017060	EHALT =	104002	EHLT	003560	EMTINT	003176
EMTTAB	003252	ENDPAS	016613	END2	001606	END2A	001574
END3	001536	END4	001610	E020	011606	E021	011656
E021A	011664	E022	011714	E022A	011722	E023	012166
E023M	017107	E024	012510	E024B	012516	E025	013256
E025MA	016733	E025MB	017033	ERR	003532	ERRHLT	003566
ERROR =	104001	EVEN	017157	FORWD =	104024	FORWDA	004026
FORWDB	004034	FSTDL	001130	HDRMSG	016703	HDR0	016715
HERE	011436	ICTR	001156	INCHK	001206	LEVEL	001152
LF =	104014	LFCNT	001204	LINE2	007626	LINE3	010350
LINE5	017111	LINE5A	017115	LOGICA	011426	MACHER	000004
MESG3	017127	MG24	012500	MG25	012504	MONIC	011762
MOVNUM	012426	NEXT	002130	NEXT1	002150	NITRSW=	004000
NXTST	001136	ODD	017163	OPEN =	000000	OUTPUT	012602
PARITY	001200	PASCNT	011604	PASMES	016632	PC =	%000007
PCHAR	001202	PFAIL	003672	POPSP =	005726	POPSP2=	022626
PRGID	001142	PRGTAB	002776	PRINTC=	104015	PRNT =	104017
PRTHDR=	104016	PRTY4 =	000200	PRTY7 =	000340	PSW =	177776
PT0	007242	PT1	007316	PT10	011172	PT11	011330
PT12	011362	PT12A	011370	PT17	011452	PT17B	011442
PT17D	011470	PT2	007440	PT3	010070	PT4	010210
PT5	010366	PT5AL	010534	PT6	010572	PT7	010760
READ =	104020	READC =	104025	READ1	004424	REF	010320
REPT	001160	RESTRT	003730	RTNNO	001134	R0 =	%000000
R1 =	%000001	R2 =	%000002	R3 =	%000003	R4 =	%000004
R5 =	%000005	SAVR6	003726	SCOPSW=	040000	SCOPTR	001140
SCRLF =	104013	SELHLT	001230	SKIP	002104	SP =	%000006
SPBOT	001100	SPC	007430	SPCNT	001172	SPSP	007410
SP2	007422	SR	001212	START	001304	STARTM	016505
STARTX	001320	START1	001232	START2	001250	START3	001266
STLSPV	003620	STLSRV	003570	STORE	014655	STPCHV=	104004
STPPA	003636	STPRA	003606	STRDRV=	104003	STRLN	012350
SWREG	000176	TEMP	001210	TEMPCH	001176	TENPWR	004310
TERM	012644	TESTC	002734	TEST30	013366	TEST31	013550
TEST32	014424	TEST33	014702	TEST34	015466	TEST35	016030
TIMER	001170	TKB	001112	TKLVL	001122	TKS	001110
TKVTR	001120	TPB	001116	TPBS	004216	TPLVL	001126

TPS	001114	TPSS	004214	TPVTR	001124	TTYCTL=	104011
TTY1	002252	TTY1B	002336	TYP	003326	TYPE =	104000
TYPEM =	104007	TYPM	003404	T30SC	013546	WAITF	002154
WIDTH	001150	XCSR	001166	$AREAD	004052	$BTASC	004220
$CR	003446	$CRLF	003434	$FORWD	003774	$LF	003436
$PRHDR	003456	$PRNT	004526	$PRTC	004536	$READ	004324
$READC	004416	$SCRLF	003362	. =	017167		

ERRORS DETECTED: 0

```
*,DZLACC+DZLACC/SOL
RUN-TIME: 10 19 0 SECONDS
CORE USED: 5K
```

N08.

```
Spooler runtime 12 Seconds, 376 KCS, 352 disk reads, 3 disk writes, 103 pages
[redacted] Date 14-Oct-76 15:[redacted]:55 Monitor IPC-0 5078 [1A3] [redacted]
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000001111111111111111111111111111111110
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111122222222223331 2
[redacted]#567890123456789012 **
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000001111111111111111111111111111111110
9000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111122222222223331 2
```