IDENTIFICATION

Product Code: AC-R998E-MC
Product Name: CZLACE0 LA36 TERM (DL11 & KL11)
Date Created: August 1978
Maintainer: DIAGNOSTIC GROUP
Authors: Robert W. Baker
          R. Quenneville
          Ralph A. Schauber
          John V. Chatalian

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this manual.

The software described in this document is furnished to the purchaser under a license for use on a single computer system and can be copied (with inclusion of Digital's copyright notice) only for use in such system, except as may otherwise be provided in writing by Digital.

Digital Equipment Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Digital.

Copyright (C) 1974,1977,1978 by Digital Equipment Corporation
1.0 DECO CZL
AC-E-J

1.1.0 Closed Problem Report AA3318

1.1.1 It was reported that a DLll-A operated at 110 baud caused failure in the AREAD routine because the 200 msec. delay is not of sufficient duration to allow setting of the Receiver Register Status "DONE" bit through the Maintenance bit facility.

The time delay was increased from 200 to 600 msec.

1.2.0 Closed Problem Report AA3643

1.2.1 Tests 56, 57, 60, 61, 62, 63, 64, 65, and 66 do not run properly when run on an LSI-11. This problem was resolved by changing the branch after the CHAIN command to go back to test for the LSI-11 switch in order to effect the appropriate action during each test.

1.2.2 Second time-out in Test 64 allows excessive wait for operator response. The time delay was reduced from "177777" to "600".

1.2.3 Common routine TYPE does not save the contents of RO resulting in the loss of this information and consequent failure. Instructions were included to save the contents of RO on entry into the routine and to restore them upon exit.

1.2.4 Loss of stack contents for non-LSI-11 computers due to incorrect sequence of instructions in Test 65 was also reported in Problem Report AA3803. Refer to 1.3.1.

1.3.0 Closed Problem Report AA3803
1.3.1 Testing of non-LSI-II computer

Results in the program hanging up because the stack gets popped away in Test 65. The branch after the test for the LSI-II switch in Test 65 should go to the CHAIN command for proper exit from the test for non-LSI-II machines. This change supersedes the change released in DEP WD-11-DZLAC-D-1.
# TABLE OF CONTENTS

1.0 ABSTRACT

2.0 REQUIREMENTS

2.1 Equipment and Assignments

2.2 Storage

2.3 Preliminary Programs

2.4 Additional Programs

3.0 LOADING PROCEDURE AND INITIALIZATION

4.0 STARTING PROCEDURE

4.1 Addresses

4.2 Switch Register Control With I/O Tests

4.3 Switch Register Control Without I/O Tests

4.4 Keyboard Control With I/O Tests

4.5 Keyboard Control Without I/O Tests

5.0 OPERATING PROCEDURE

5.1 Switch Register Control

5.2 Keyboard Control

6.0 TEST DESCRIPTIONS

6.1 Printing Tests

6.1.1 Test0 - Data Path Test

6.1.2 Test1 - Printable Character Test

6.1.3 Test2 - Non-printable Character Test

6.1.4 Test3 - Carriage Return Test

6.1.5 Test4 - Multiple Line Feed Test

6.1.6 Test5 - Single Line Feed Test

6.1.7 Test6 - Backspace Test

6.1.8 Test7 - Overprint Test

6.1.9 Test8 - Printing Frequency Sweep Test

6.1.10 Test9 - Ribbon Feed Test

6.1.11 Test10 - Printer Bell Test

6.1.12 Test11 - Life Test
6.2 Echo Tests
  6.2.1 Test20 - Character Echo Test
  6.2.2 Test21 - Line Echo Test, Fast Rate
  6.2.3 Test22 - Line Echo Test, Slow Rate
  6.2.4 Test23 - Character/Code Echo Test
  6.2.5 Test24 - Selected Pat
  6.2.6 Test25 - Bell Echo Test
  6.4 Standard I/O Test
1.0 ABSTRACT

This diagnostic is divided into three basic sections:

1. A check of the console terminal interface logic.

2. A check of the printing characteristics and control logic.

3. An echo portion designed to check the keyboard and to aid in the diagnosis of terminal problems. 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 to run on all models of the PDP-11 computer with either a KL11 or DL11 console terminal interface. The diagnostic is preset to test up to 16 additional terminals (on DL11’s) assigned between addresses 776500 and 776676. This preset quantity (16) and preset address (776500) can be changed by depositing the quantity in DLNR and the starting address in DLADR. For example, to allow for up to 31 additional terminals, the address 775610 could be placed into DLADR and the octal equivalent of 31, i.e., (37) would be placed into DLNR. The number of additional DL11’s actually tested will be adjusted automatically downward based upon the first DL11 address (within the implied range) found to be unresponsive. Thus if there is no DL11 present to match the address in DLADR only the console terminal will be tested. Therefore, all DL11’s in excess of the console terminal must have contiguous address assignments with the lowest address correspond.
ponding to the value in DLADR.

The console terminal (assigned standard) can be reassigned by placing the address of its receiver status register into CO NADD and its receiver interrupt vector into CONVEC. This reassignment can be made to a terminal within the set of terminals implied by DLNR and DLADR without adverse effect. Note that a terminal with a slower speed (if any) will determine the speed at which all of the terminals are tested. Such a terminal should generally be excluded from the test, or tested separately. (Refer to the symbol definitions in the listing for the above mentioned 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 to the appropriate interface diagnostic for further help in locating the problem if needed.

2.4 ADDITIONAL PROGRAMS

This diagnostic is for verification of basic terminal functions only. If the terminals under test have hardware options installed run diagnostic MAINDEC-11-DZLAF-A, the LA36 TERMINAL OPTIONS TEST.

3.0 LOADING PROCEDURE AND INITIALIZATION

Load the LA36 diagnostic program tape following normal procedures. Before starting the program, refer to the description of the routine "DLYM". 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-standard terminal addresses and for testing multiple DL11 interfaces.

If a hardware switch register does not exist, the program will use the contents of location 176 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.

If the CPU is an LSI-11, 11/03 be sure to set switch register bit 9 to 1. Special tests are run on the DLV11 interface.
4.1 STARTING PROCEDURE

4.1 STARTING ADDRESSES

200(B) = Run with Switch Register Control
- Perform Console Terminal I/O tests.

204(B) = Run with Switch Register Control
- skip Console Terminal I/O tests.

210(B) = Run with Keyboard Control
- Perform Console Terminal I/O tests.

214(B) = 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 switch register bit 9 to a 1 if the processor is an LSI-11, 11/0.

C. Refer to Section 5.1.5.

D. Set the switch register bit 8 equal to 1 or 0 and press the start switch. A message will be printed indicating the number of DL11's being tested. Refer to Section 5.1.8.

E. If bit 8 were zero when starting, the Printer tests are executed sequentially, after the entire series of I/O tests are executed.

F. If bit 8 was set when the start switch was pressed, the entire series of I/O tests will be executed and the CPU will halt at location SELHLT. The program will then be waiting for control via the switch register.

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 of columns (octal). Refer to Section 5.1.9.

C. Set switch register bit 9 to a 1 if the processor is an LSI-II/11. Refer to Section 5.1.5.

D. Set switch 8 and press the start switch. A message will be printed indicating the number of DL11's being tested. Refer to Section 5.1.6.

E. If bit 8 was zero when the printer tests are executed sequentially after the entire series of I/O tests are executed.

F. 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 terminal keyboard. Refer to Section 5.2.

4.5 Keyboard Control - Without I/O Tests

Same as Section 4.4 except in 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 con-
sole 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 set as desired except as noted in the specific switch descriptions. Refer to the detailed switch descriptions for further, more complete information.

<table>
<thead>
<tr>
<th>SWITCH NUMBER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>T AT END OF TEST</td>
</tr>
<tr>
<td></td>
<td>1(up) = HAL</td>
</tr>
<tr>
<td></td>
<td>0(down) = CONTINUE TEST SEQUENCE</td>
</tr>
<tr>
<td>14</td>
<td>ERROR</td>
</tr>
<tr>
<td></td>
<td>1(up) = CONTINUE ON ERROR</td>
</tr>
<tr>
<td></td>
<td>0(down) = HALT ON</td>
</tr>
<tr>
<td>13</td>
<td>INDIVIDUAL TEST</td>
</tr>
<tr>
<td></td>
<td>1(up) = DRIVE ONLY CONSOLE TERMINAL</td>
</tr>
<tr>
<td></td>
<td>0(down) = DRIVE ALL TERMINALS</td>
</tr>
<tr>
<td>11</td>
<td>1(up) = LOOP ON</td>
</tr>
<tr>
<td></td>
<td>0(down) = NORMAL TEST SEQUENCE</td>
</tr>
<tr>
<td>9</td>
<td>TEST NUMBER SELECTION</td>
</tr>
<tr>
<td></td>
<td>1(up) = CPU TYPE IS AN LSI-11, 11/03</td>
</tr>
<tr>
<td></td>
<td>0(down) = ALL OTHER PDP-11's</td>
</tr>
<tr>
<td>8</td>
<td>1(up) = RUN TEST ONCE AND HALT</td>
</tr>
<tr>
<td></td>
<td>0(down) = LOOP ON TEST SEQUENCE</td>
</tr>
<tr>
<td>5-0</td>
<td>ART-UP</td>
</tr>
<tr>
<td></td>
<td>NUMBER OF COLUMNS AT ST</td>
</tr>
</tbody>
</table>

5.1.1 Switch 15

With switch 15 in the up position, the program will halt at the end of the current test. Replacing switch 15 to the down position and press-
CONTINUE will continue the normal test operation. During the halt, any of the control 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 continue 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 RO. 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

Error halts can occur only during the I/O tests. The terminal is connected to a serial line and there is no error information returned to the program from the terminal. Therefore the program cannot report errors occurring in the terminal. Errors detected during the interface tests will result in halts as described above.

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 on the current test as long as switch 11 remains up. Replacing switch 11 down will cause the program to resume normal operation at the completion of the test.

5.1.5 Switch 9
Placing switch 9 up at the start of the test will cause an automatic change in the DELAY timing, and the execution of special DLVII I/O tests. The DLVII has no maintenance mode and will cause th
e program
to hang if tested as a DL11.

5.1.6 Switch 8

With switch 8
  in the down position the program will continue to loop
through the present test sequence. Placing switch 8 up will cause the
program to halt (at SFL/HLT) 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, and 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 SELL/HLT 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.

5.1.7 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.

5.1.8 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 leave switches 7 to 0 down to test the full 132(10) columns.
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 switch 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 following control characters:

. (period) = Run the selected test once and return for another test selection.

L "RUBOUT" = Loop on the selected test unit.

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 DESCRIPTION

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 C - 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 each received carriage return, there will be a blank line between each printed line.

EXAMPLE:

```
g*y*g*y*g*y*g*y*g*y*g*y*g*y
*y*g*y*g*y*g*y*g*y*g*y*g*y
`y*g*y*g*y*g*y*g*y*g*y*g*y
```

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.
EXAM PLE:

```

  AAA   BBB
  ###   ###
  CCC   CCC
  DDD   DDD
  E     
  FFF   FFF
  GGG   qqq
  HHH   hhh
  ---   ---
  JJJ   JJJ
  KKK   kkk
  LLL   LLL
  M MM   m mm
  NNN   nnn
  OOO   ooo
  PPP   ppp
  QQQ   qqq
  RRR   rrr
  SSS   sss
  TTT   ttt
  UUU   uuu
  VVV   vvv
  WWW   www

  xxx   xxx
  yyy   yyy

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,
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.
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 zeros on either side of it.

EXAMPLE:

```
006 ACK 001 SOH 002 STX
020 DLE 021 DC1 022 DC2
023 DC3 024 DC
4 025 NAK 026 SYN 027 ETB 030 CAN
031 EM 032
SUB 034 FS 035 GS 036 RS 037 US
177 DEL
```

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 diagonal 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
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 between the two dashed lines, i.e., 63(10) 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 Line
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 back slashes "\" and line feeds to print a diagonal line down the paper. When a back slash 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 misprints or missing characters during the first 16(10) characters of the middle reference line. Also, any paper feed problems will cause misalignment of the slashes forming the diagonal line.
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.
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
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
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-M
- Row 3: S-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 M M M M M
S S S S S S S S S S
```

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
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 1.0 seconds using the following formula to create a logarithmic increase:

\[
\text{New Delay} = \text{Old Delay} + \frac{\text{Old Delay}}{16} + \frac{\text{Old Delay}}{1}
\]

The last two lines do just the reverse. The time interval between characters is decreased from 1.0 seconds to 30(10) milliseconds using the following formula to again create a logarithmic decrease:

\[
\text{New Delay} = \text{Old Delay} - \frac{\text{Old Delay}}{16} - \frac{\text{Old Delay}}{128}
\]

Look for possible misalignment 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:

```
H
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
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
XX
XXX
XXXX

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 eight 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:

```
AAAAAA
AAAAAA

AA
RRBRRRBBRRRRBRRRBBBBBRRRRRBBRRRRRBBRR
RRBRRRBBRRRRBRRRBBBBBRRRRRBBRRRRRBBRR

BBBBBB
```

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:

```
AAAAAA
AAAAAA
```
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 SELHLT) 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 "rhubout") will be echoed to the printer.

If the 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 "rhubout". 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 E
YEN
240 SP EVEN
```

6.2.5 Test 24 - Selected Pattern Echo Test

This test is designed to give maintenance the flexibility to choose the 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 occurred 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.
6.4 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 control 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 "ER RHALT" if switch 14 is down.
# Table of Contents

<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-100</td>
<td>Switch Register Options</td>
</tr>
<tr>
<td>2-1250</td>
<td>Special Operational Information</td>
</tr>
<tr>
<td>3-1350</td>
<td>System Equates</td>
</tr>
<tr>
<td>4-1350</td>
<td>Trap Catcher &amp; Starting Addresses</td>
</tr>
<tr>
<td>5-1910</td>
<td>Symbol Definitions</td>
</tr>
<tr>
<td>6-100</td>
<td>Program Initialization &amp; Control</td>
</tr>
<tr>
<td>13-46300</td>
<td>Common Routines Used by LA36 Tests</td>
</tr>
<tr>
<td>26-100</td>
<td>I/O Logic Tests</td>
</tr>
<tr>
<td>42-100</td>
<td>LA36 Printer Tests</td>
</tr>
<tr>
<td>55-100</td>
<td>LA36 Echo Tests</td>
</tr>
<tr>
<td>62-100</td>
<td>Misc. Diagnostic Messages</td>
</tr>
</tbody>
</table>
.TITLE  CIZLACEO LA36  TERM (OLII & KLII) MACRO MIIIO  25-AUG-78 10:13 PAGE 1

.SWITCH  SPECIAL OPERATIONAL INFORMATION

1.---  THE STANDARD CONSOLE TERMINAL INTERRUPT VECTOR AND REGISTER
       LOCATION ARE DETERMINED BY THE "SWITCH" REGISTERS "05-00" AND "07-00"
       CHANGED BEFORE START-UP.

2.---  BEFORE START UP REFER TO THE DESCRIPTION OF THE ROUTINE "DLY"-
       SHOWN BE SET UP BEFORE RUNNING THE DIAGNOSTIC.

3.---  IF CPU IS A POP-11, LSI-II SET SWITCH REGISTER
       BIT 8 TO A 1. SPECIAL TESTS ARE RUN ON THE CPU.

4.---  SYSTEMS WITHOUT A HARDWARE SWITCH REGISTER SHOULD USE
       MEMORY LOCATION 176 AS A SOFTWARE SWITCH REGISTER.

5.---  THIS DIAGNOSTIC IS FOR VERIFICATION OF BASIC TERMINAL
       FUNCTIONS ONLY. IF THE TERMINAL SHOWN TESTS HARDWARE
       OPTIONS INSTALLED FOR DIAGNOSTIC METHOD-II-H/G/E, THE
       LA36 TERMINAL OPTIONS TEST.

.SPECIAL OPERATIONAL INFORMATION

1.---  THE STANDARD CONSOLE TERMINAL INTERRUPT VECTOR AND REGISTER
       LOCATION ARE DETERMINED BY THE "SWITCH" REGISTERS "05-00" AND "07-00"
       CHANGED BEFORE START-UP.

2.---  BEFORE START UP REFER TO THE DESCRIPTION OF THE ROUTINE "DLY"-
       SHOWN BE SET UP BEFORE RUNNING THE DIAGNOSTIC.

3.---  IF CPU IS A POP-11, LSI-II SET SWITCH REGISTER
       BIT 8 TO A 1. SPECIAL TESTS ARE RUN ON THE CPU.

4.---  SYSTEMS WITHOUT A HARDWARE SWITCH REGISTER SHOULD USE
       MEMORY LOCATION 176 AS A SOFTWARE SWITCH REGISTER.

5.---  THIS DIAGNOSTIC IS FOR VERIFICATION OF BASIC TERMINAL
       FUNCTIONS ONLY. IF THE TERMINAL SHOWN TESTS HARDWARE
       OPTIONS INSTALLED FOR DIAGNOSTIC METHOD-II-H/G/E, THE
       LA36 TERMINAL OPTIONS TEST.
.SEITL  PROGRAM INITIALIZATION & CONTROL

*******

COMMON SALT---WHEN IN SWITCH REGISTER CONTROL THE CPU

A NEW TEST WILL BE EXPECTED TO BE STARTED

*******

CHLT:    TST    MOV    MOV    MOV    MOV    MOV    MOV
         R5V    TST    SELHLT:  en T
         START2             R
START3:  MOV
         BR

ROSWR

MOY

CLR

CLR
PROGRAM INITIALIZATION & CONTROL

************

READ THE PAPER WIDTH, NUMBER OF COLUMNS:
CONVERT TO ASCII CHARACTERS & WRITE OUT
THE TEST COLUMN POSITION & WIDTH OF COMBINED
THE INITIALIZATION MESSAGE & THE PROGRAM TITLE OR THE DLI1 COUNT
RESULT WILL BE DISPLAYED ON THE PANEL.

************

CPLACO LAS TERM DLI1 MACHIIII 25-AUG-78 10:11:3 PAGE 9
SEQ 0055

************

EXCLUDE THE STRING OF CONSOLE TERMINAL EOF TESTS.
WHEN EITHER CALL A LOCATION DECAY CONTINUE WITH
STOP TESTS IS A PROJECTION OF 38 167 4.

************

CPLACO LAS TERM DLI1 MACHIIII 25-AUG-78 10:11:3 PAGE 9
SEQ 0055

************

EXCLUDE THE STRING OF CONSOLE TERMINAL EOF TESTS.
WHEN EITHER CALL A LOCATION DECAY CONTINUE WITH
STOP TESTS IS A PROJECTION OF 38 167 4.
**CZLACEO LA36 TERM (OLLII & NTLII) MACRO MillO 25-AUG-78 10:13 PAGE II**  
**PROGRAM INITIALIZATION & CONTROL**

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>17606</td>
<td>POPD2</td>
<td>PUSHP2 FROM STACK</td>
</tr>
<tr>
<td>17610</td>
<td>DEX</td>
<td>DEX</td>
</tr>
<tr>
<td>17612</td>
<td>COM</td>
<td>COM</td>
</tr>
<tr>
<td>17614</td>
<td>CMP</td>
<td>CMP</td>
</tr>
<tr>
<td>17616</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17618</td>
<td>RTI</td>
<td>RTI</td>
</tr>
<tr>
<td>17620</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17622</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17624</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17626</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17628</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17630</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17632</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17634</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17636</td>
<td>JSR</td>
<td>JSR</td>
</tr>
</tbody>
</table>

**TTY1: PROGRAM FOR TTY1**

- **PUSHP2 FROM STACK**
- **DEX**
- **COM**
- **CMP**
- **BIC**
- **RTI**
- **JSR**
- **BIC**
- **JSR**
- **BIC**
- **JSR**
- **BIC**
- **JSR**
- **BIC**
- **JSR**

**SEQ 0058**

**CZLACEO LA36 TERM (OLLII & NTLII) MACRO MillO 25-AUG-78 10:13 PAGE II-1**  
**PROGRAM INITIALIZATION & CONTROL**

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>17606</td>
<td>POPD2</td>
<td>PUSHP2 FROM STACK</td>
</tr>
<tr>
<td>17610</td>
<td>DEX</td>
<td>DEX</td>
</tr>
<tr>
<td>17612</td>
<td>COM</td>
<td>COM</td>
</tr>
<tr>
<td>17614</td>
<td>CMP</td>
<td>CMP</td>
</tr>
<tr>
<td>17616</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17618</td>
<td>RTI</td>
<td>RTI</td>
</tr>
<tr>
<td>17620</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17622</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17624</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17626</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17628</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17630</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17632</td>
<td>JSR</td>
<td>JSR</td>
</tr>
<tr>
<td>17634</td>
<td>BIC</td>
<td>BIC</td>
</tr>
<tr>
<td>17636</td>
<td>JSR</td>
<td>JSR</td>
</tr>
</tbody>
</table>

**SEQ 0059**
**THIS SECTION CONTAINS MOST ROUTINES CALLED BY THE VARIOUS TESTS OTHER THAN TRAP THROUGHS LOCATION 7000 AND BY SUBROUTINE CALLS (JOB RC,...)**

**TYPE-- A COMMON ROUTINE USED TO TYPE MESSAGES ON THE COMPUTER TERMINAL. THE CALL TERMINATES THE CURSOR SEQUENCE CALLED SEQUENCE MESSAGE ADDRESS OF MESSAGE

************************************************************************************************
**ENTRTY TYPE-- INTERNAL MESS ROUTINE
**ENTRTY DESC-- INSTRUCTIONAL PAGE
**ENTRTY LOC-- PROCEDURE SETUP
**ENTRTY ADDR-- PROCEDURE SETUP
**ENTRTY DEST-- PROCESSOR/DEVICE ROUTINE
**ENTRTY TXT-- PROCESSOR/DEVICE ROUTINE, MULTI DEVICES
**ENTRTY ADDR-- PROCESSOR/DEVICE ROUTINE
**ENTRTY TXT-- PROCESSOR/DEVICE ROUTINE"
TYPM--MULTI TYPE A COMMON ROUTINE TO OUTPUT A MESSAGE IN ALL CAPS IF THE TYPEM \nIS NOT READY, THE CHARACTER WILL NOT BE TYPED.

TYPM: MOV $25, R0 \n; GET POINTER TO ADDR OF MSEG
15: MOV $12, R1 \n; ADDR OF MSEG TO R1
15: CMP R0 \n; CHECK IF MSEG AUTO CR-LF
25: INY \n; CONTINUE IF NOT NULL
25: JNZ 15 \n; CONTINUE
35: PRINTC \n; SET PTR CH
UP \n; SET NEXT CHAR.
SCBR: CH \n; SET CH
SPRCH: MOV $10, R0 \n; TRANSMIT
PRINTC \n; SEND NUL CODE.
TYPEM, PRINT MESSAGE
HDRS: MOV $6N, R0 \n; SEND
PRINTC \n; MSG
ASH ASR R0 \n; TRANSMIT
BIC #77770, R0 \n; MASK LAST DIGIT
ADD $50, R0 \n; MAKE ASCII
PRINTC \n; PRINT
E LF \n; SEND CR
RTI \n; RETURN TO CALLER

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

E R R A-- COMMON ERROR RETURN FROM I/O TESTS, HALTS \nWITH ADDRESS OF ERROR IN REG. TO CONOBJE \nPUT SAME TEST BUT NOT IN REG TO CONOBJE \nGET THE SCOPE BIT (14) = 1 AND PRESS CONTINUE

ERR: LOPR: #BIT15,PRGIO \n; CHECK SCOPE SWITCH, BRANCH
LSTPRA: MOV R1 \n; SET RETURN ADDR AND RETURN TO TEST FOLLOWING CALL

****** STLSRV-- THIS ROUTINE SETS UP KEYBOARD INTERRUPT VECTOR AND PRIORITY CALLING SEQUENCE \n
STLSRV: MOV R0 \n; SET RETURN ADDR AND VECTOR
STPPRA: MOV R0 \n; CALL INTO R0

****** STLSPV-- THIS ROUTINE SETS UP PRINTER INTERRUPT VECTOR AND PRIORITY CALLING SEQUENCE \n
STLSPV: MOV R0 \n; SET RETURN ADDR AND VECTOR
STPPRA: MOV R0 \n; CALL INTO R0

******** END
DELAY--A COMMON ROUTINE TO DELAY PROCESSING
A COUNTER NUMBER OF MSEC.

- CALL $40,40
- MOV $40,0
- ADD $11,0
- S$1,4

THE DELAY IS EFFECTED BY THE EXECUTION OF THE LOOP;
15: DEC $11
16: BNE LOOP
17: MOV MOY DEC $11
18: BNE RV
19: RTC
20: MOV RDI,0
21: JMP $0

1: LOOP
2: DEC $11
3: MOV MOY DEC $11
4: BNE RV
5: RTC
6: MOV RDI,0
7: JMP $0

SECONDARY ROUTINE TO DELAY PROCESSING
A COUNTER NUMBER OF MSEC.

- CALL $40,30
- MOV $40,0
- ADD $11,0
- MOV $11,0
- S$1,4

THE DELAY IS EFFECTED BY THE EXECUTION OF THE LOOP;
15: DEC $11
16: BNE LOOP
17: MOV MOY DEC $11
18: BNE RV
19: RTC
20: MOV RDI,0
21: JMP $0

1: LOOP
2: DEC $11
3: MOV MOY DEC $11
4: BNE RV
5: RTC
6: MOV RDI,0
7: JMP $0

PFAIL--POWER FAIL ROUTINE
SAVE ALL REGISTERS AND SET RESTART ADDRESS INTO LOCATION 24

1: MOV MOY DEC $11
2: BNE RV
3: RTC
4: MOV RDI,0
5: JMP $0

RESTART--POWER FAIL RECOVERY

120 100 RESTORE ALL REGISTERS AND GO TO START

PHA

174312

MOV SP, R4

JMP 27000

ASCIZ <CRLF>/POWER/<CRLF>

.EVEN
FORWARD--THIS ROUTINE TRANSFERS THE 7 DO 4 ARGUMENTS
FROM THE TEST ROUTINE. THEY ARE:
1. ROUTINE NUMBER
2. ADDRESS OF PARAMETERS
3. SCOPE ENTRY ADDRESS (FOR TESTS ONLY)

FORWARD: MOV RSTP, R
IND OF NEXT TEST TO BE
NEXT TESTコーネ.そこで再度のテストを

FORWDA: MOV TSTP, R
SET INIT TO GET

FORWDB: MOV SCARP, R
FORCE INTEST COUNT OF 1

FORWARD--A ROUTINE WHICH, THROUGH THE FACILITY OF
THE HARDWARE, TO UPDATE THE OFF-BOARD
STATUS FOR USE IN TESTS. IF THE TESTS ARE TO CONTINUE
PRESS CONTINUE TO CONTINUE TESTS.

SAREAD: MOV CONADD, R; KEYBOARD STATUS ADDRESS (777560) TO CONADD
MOV TBS, R; KBST (777562)

SARER: MOV KBVTR, R; KBVTR (60)
MOV KBVTMS, R; KBVTMS (64)

CONT--THIS ROUTINE SETS UP THE DEVICE ADDRESSES
AND INTERRUPT VECTORS FOR THE CONSOLE.

COMPL: MOV COMADD, R; CONSOLE KEYBOARD STATUS ADDRESS (775420) TO COMADD
MOV TBS, R; KBST (775422)
MOV KBVTMS, R; KBVTMS (775423)

CONVE: MOV CONVEC, R; CONSOLE KEYBOARD INTERRUPT VECTOR (60) TO CONVEC
MOV TPS, R; TPS (64)

CONP: MOV CONP, R; CONSOLE PRINT INTERRUPT VECTOR (64) TO CONP
MOV TPR, R; TPR (64)

TPSS: .WORD 0
TPSR: .WORD 0

LIST TEST STATES REG ADD
LIST TEST BUFFER REG ADD
**BINARY TO ASCII CONVERSION (1 TO 5 ASCII CHARACTERS)**

**CALLING SEQUENCE**

1. MOV ADDRESS OF LOC. TO STORE FIRST ASCII CHAR. INTO R0
2. MOV BINARY NUMBER TO BE CONVERTED INTO ASCII
   NUMBER TO BE CONVERTED AS A POWER OF TEN INTO R2
3. MOV ASCII

**RTASC**: MOV R2, C1 + C2

1. ADD @DTEMP, @DTEMP + 1
2. MOV BINARY NUMBER TO BE CONVERTED AS A POWER OF TEN INTO R3
3. MOV BINARY NUMBER TO BE CONVERTED AS A POWER OF TEN INTO R4

**CPL**

1. JSR TST
2. BR JSR
3. CMP BEQ JMP COMB
4. BR RTI

**SEND**

1. JSR CSC

**PRINT**

1. JSR SCR

**HD**: JSR HD

**SPRINT**: JSR SCR

**COMMON ROUTINES USED BY LA2**: TESTS

**SPLINT**, **TSTB**: BRANCH IF NOT SET, RETURN

**SPRINT**: CHECK PRINTER READY FLAG

**HD**: CHECK THE KEYBOARD

**SEND**: COMMON ROUTINE TO CHECK THE KEYBOARD
32800
LOGIC TESTS

**********
ONLY THE CONSOLE TERMINAL IS TESTED.
UPON COMPLETION, THE CPU WILL EITHER HILT IF 51
TESTS ARE PASSED, OR CONTINUE IF 51 TESTING
IF AN I/O TEST FAILS, THE CPU WILL NOT BE Powered
WITH THE ADDRESS OF THE ERROR IN THE ERROR
LOGIC TESTS WILL CAUSE THE CPU TO HILT 14
LONG TEST, AS IN 51, BEFORE THE CONSOLE SWITCH 15
PRESSURE, THE FAILED TEST WILL LOOP ON ITSELF

**********
TEST 640--TESTS THE ABILITY TO REFERENCE THE
RECEIVER STATUS WORD (TSW) WITHOUT TRAPPING.

**********
TEST 655--TESTS THE ABILITY TO REFERENCE THE
TRANSMITTER STATUS WORD (TPS) WITHOUT TRAPPING.

**********
TEST 665--TESTS THE ABILITY TO REFERENCE THE
TRANSMITTER BUFFER (TPB) WITHOUT TRAPPING.

**********
TEST 675--TESTS THE ABILITY TO REFERENCE THE
RECEIVER BUFFER (TPB) WITHOUT TRAPPING.
LOGIC TESTS

********

ATV: 44 TEST NUMBER

TEST 44--TESTS THE ABILITY TO SET AND CLEAR THE RECEIVED INTERRUPT ENABLE BIT.

********

ATV: 45

TEST NUMBER

TEST 45--CHECKS THAT THE RECEIVED INTERRUPT ENABLE BIT CAN BE CLEARED WITH RESET INSTRUCTION.

********

ATV: 46

TEST NUMBER

TEST 46--TESTS THE ABILITY TO SET AND CLEAR THE TRANSMITTED INTERRUPT ENABLE BIT.

********

ATV: 47

TEST NUMBER

TEST 47--TESTS THE ABILITY TO CLEAR TRANSMITTED INTERRUPT ENABLE BIT WITH RESET INSTRUCTION.
MW MM

**********

AT10: 50
JTEST NUMBER
I TEST COUNT
001000 172732 15:
DOP$,$5111,56H
RTX 25
RESET
172624
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********

AT11: 51
JTEST NUMBER
I TEST COUNT
000000 172624 15:
DOP$,$226,90
RTX 25
DELAY
150 MSEC.
172524
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********

AT12: 52
JTEST NUMBER
I TEST COUNT
000226 172524 15:
DOP$,$226,90
RTX 25
DELAY
150 MSEC.
172524
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********

AT13: 53
JTEST NUMBER
I TEST COUNT
000226 172624 15:
DOP$,$226,90
RTX 25
DELAY
150 MSEC.
172524
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********

AT14: 54
JTEST NUMBER
I TEST COUNT
000000 172732 15:
DOP$,$5111,56H
RTX 25
RESET
172624
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********

AT15: 55
JTEST NUMBER
I TEST COUNT
001000 172732 15:
DOP$,$5111,56H
RTX 25
RESET
172624
XRFTB 75
ERROR
RDY
25:
CHAIN TO NEXT TEST

**********
**TEST 54**--Tests that the Transmit Ready does cause an
interrupt when the processor is in a privileged
mode lower than the transmit interrupt request level.

**TEST 55**--Tests that the Transmit Ready does not
interrupts after an RTI when the Ready bit has
just been reset.

**TEST 56**--Checks that reset clears the receiver done bit.

---

**TEST 54**--Tests that the Transmit Ready does cause an
interrupt when the processor is in a privileged
mode lower than the transmit interrupt request level.

**TEST 55**--Tests that the Transmit Ready does not
interrupts after an RTI when the Ready bit has
just been reset.

**TEST 56**--Checks that reset clears the receiver done bit.

---

**TEST 54**--Tests that the Transmit Ready does cause an
interrupt when the processor is in a privileged
mode lower than the transmit interrupt request level.

**TEST 55**--Tests that the Transmit Ready does not
interrupts after an RTI when the Ready bit has
just been reset.

**TEST 56**--Checks that reset clears the receiver done bit.

---

**TEST 54**--Tests that the Transmit Ready does cause an
interrupt when the processor is in a privileged
mode lower than the transmit interrupt request level.

**TEST 55**--Tests that the Transmit Ready does not
interrupts after an RTI when the Ready bit has
just been reset.

**TEST 56**--Checks that reset clears the receiver done bit.

---

**TEST 54**--Tests that the Transmit Ready does cause an
interrupt when the processor is in a privileged
mode lower than the transmit interrupt request level.

**TEST 55**--Tests that the Transmit Ready does not
interrupts after an RTI when the Ready bit has
just been reset.

**TEST 56**--Checks that reset clears the receiver done bit.
THAT REFERENCING THE RECEIVER BUFFER

In AT17 --

HIT.

a 806430 000057 AT17: 57 ; TEST NUMBER 33 6

06432 006500 AT20; NEXT TEST

m~ 001000 172246 IS: 

B ITLSR, hR Set Test

ERROR

R0 6 

H: CHNN

5 6478R 1$; REPEAT TEST

ms; i*::*;T*60--CHECK THAT RECEIVER DONE HIT IS ABLE TO

THE 35500 THE

i CAUS AN

INTERRUPT.

**** ***

005~ 46 CLR -(SP) ; SET PROCESS STATUS TO ZERO

80~240 000100 172034 35: RTI

m mg

379 6566 Chain to NEXT TEST 38000 006570 000751

BR

376 006560 104001 ERROR IERROR, RECEIVER FAILED TO INTERRUPT

mun~m

BR 5$ ; ENABLE RECEIVER INTERRUPT

5S: POPSP2 I ERROR, RECEIVER INTERRUPTED, CLEAN STACK

BP 45 ,ORANCH 45

---

THAT REFERENCING THE RECEIVER BUFFER

In AT17 --

HIT.

a 806430 000057 AT17: 57 ; TEST NUMBER 33 6

06432 006500 AT20; NEXT TEST

m~ 001000 172246 IS: 

B ITLSR, hR Set Test

ERROR

R0 6 

H: CHNN

5 6478R 1$; REPEAT TEST

ms; i*::*;T*60--CHECK THAT RECEIVER DONE HIT IS ABLE TO

THE 35500 THE

i CAUS AN

INTERRUPT.

**** ***

005~ 46 CLR -(SP) ; SET PROCESS STATUS TO ZERO

80~240 000100 172034 35: RTI

m mg

379 6566 Chain to NEXT TEST 38000 006570 000751

BR

376 006560 10401 ERROR IERROR, RECEIVER FAILED TO INTERRUPT

mun~m

BR 5$ ; ENABLE RECEIVER INTERRUPT

5S: POPSP2 I ERROR, RECEIVER INTERRUPTED, CLEAN STACK

BP 45 ,ORANCH 45

---

THAT REFERENCING THE RECEIVER BUFFER

In AT17 --

HIT.

a 806430 000057 AT17: 57 ; TEST NUMBER 33 6

06432 006500 AT20; NEXT TEST

m~ 001000 172246 IS: 

B ITLSR, hR Set Test

ERROR

R0 6 

H: CHNN

5 6478R 1$; REPEAT TEST

ms; i*::*;T*60--CHECK THAT RECEIVER DONE HIT IS ABLE TO

THE 35500 THE

i CAUS AN

INTERRUPT.

**** ***

005~ 46 CLR -(SP) ; SET PROCESS STATUS TO ZERO

80~240 000100 172034 35: RTI

m mg

379 6566 Chain to NEXT TEST 38000 006570 000751

BR

376 006560 10401 ERROR IERROR, RECEIVER FAILED TO INTERRUPT

mun~m

BR 5$ ; ENABLE RECEIVER INTERRUPT

5S: POPSP2 I ERROR, RECEIVER INTERRUPTED, CLEAN STACK

BP 45 ,ORANCH 45
SEQ 0088

**LOGIC TESTS**

**SEQUENCE 0088**

- **AT22:**
  - **TEST NUMBER:** 23
  - **TEST:** TESTS THAT THE RECEIVER DONE DOES CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
  - **ITERATION COUNT:** 3

**TEST 23**

- **TEST:** CHECKS THAT THE RECEIVER DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 1

**AT22:**

- **LAYOUT:** LAYOUT
- **TEST NUMBER:** 23
- **TEST:** TESTS THAT THE RECEIVER DONE DOES CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 3

**TEST 23**

- **TEST:** CHECKS THAT THE RECEIVER DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 1

**AT22:**

- **LAYOUT:** LAYOUT
- **TEST NUMBER:** 23
- **TEST:** TESTS THAT THE RECEIVER DONE DOES CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 3

**TEST 23**

- **TEST:** CHECKS THAT THE RECEIVER DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 1

**AT22:**

- **LAYOUT:** LAYOUT
- **TEST NUMBER:** 23
- **TEST:** TESTS THAT THE RECEIVER DONE DOES CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 3

**TEST 23**

- **TEST:** CHECKS THAT THE RECEIVER DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS AT A PRIORITY ONE LEVEL
- **ITERATION COUNT:** 1
**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**

**LOGIC TESTS PAGE**
I70 LOGIC TESTS

****

AT26--TEST66--CHECK THAT READING TEN CLEAR'S DONE BIT
AND THAT DONE CLEAR'D DOES NOT CAUSE AN INTERRUPT

****

AT26: 66: I TEST NUMBER

001000 171406 15: LSI, LSII1, #SH

001000 171276 25: READ BITS

001000 171370 35: CHECK THE DONE BIT

001000 171242 45: RECEIVED D0 E1 NOT SET

001000 171232 55: MAKE SURE DATA BUFFER DID NOT CLEAR DONE

001000 171222 65: RECEIVE INTERRUPT

001000 171212: CLEAN UP

001000 171202: CLEAN UP THE STACK

001000 171192: CLEAN UP THE STACK
THE LA36 PRINTER TESTS WILL BE EXECUTED IN A
CONTINUOUS LOOP OUTPUTTING TO ALL MULTIPLE DL1'S.
THE 25-PRINT TESTS ARE EXECUTED INDIVIDUALLY ONCE ON THE FIRST LOADED OR
REPRINTED PAPER. THE TEST OF THE ENTIRE LOADED OR PRINTED
TESTS DEPENDS ON INSTRUCTIONS IN THE INTRODUCTION
FOR PROPER MODE OF OPERATION.

PT5 -- DATA PATH TEST -- FOUR LINES OF ALTERNATING
WIDTHS. THE PATTERN WILL APPEAR AS FOLLOWS:

XXXXXXXXX

PT6: 0   TEST NUMBER
1   NEXT TEST
PT10: PRINT COLUMN & MESC
157: MOV 126 R3
25: MOV 126 R3
351: MOV WIDTH R1
457: DEC R1

XXXXXX

PT7: PSI -- PRINTER CHARACTER TEST --- PRINTS ALL PRINTABLE CHARACTERS

XXXXXXXXX

PT1: 1   TEST NUMBER
5: NEXT TEST
151: MOV 126 R3
251: MOV 126 R3
351: MOV WIDTH R1
451: MOV 126 R3
551: MOV WIDTH R1
651: MOV 126 R3
751: MOV WIDTH R1
851: MOV 126 R3
951: MOV WIDTH R1

XXXXX

PT8: PP5: 0   PRINT COUNT TO 0
10: PRINT CHAR TO R4
19: PRINT CHAR TO R4
29: PRINT CHAR TO R4
39: PRINT CHAR TO R4
49: PRINT CHAR TO R4
59: PRINT CHAR TO R4
69: PRINT CHAR TO R4
79: PRINT CHAR TO R4
89: PRINT CHAR TO R4
99: PRINT CHAR TO R4

XXX
PT2 -- NON-PRINTING CHARACTER TEST.  THIS TEST
OF ALL NON-PRINTING CHARACTERS.  FOLLOWING EACH
CHARACTER IS THE OCTAL CODE FOLLOWED BY THE NON-PRINTING
CHARACTER.  THESE ARE THE CODES FOR PRINTING THOSE NON-PRINTING
CHARACTERS.  IF THESE CODES ARE NOT PRINTED, REFER TO THE DOCUMENT FOR A LIST OF THOSE
TESTED.

**XXXXXXX**

PT2:  TEST NUMBER

**XXXXXXX**

PT2:  2

IDE:  .ASCII /000 NUL001 SOM002 STY/

IDE:  .ASCII /006 ACK020 DLE021 DC1/

IDE:  .ASCII /022 DC023 DC024 DC4/

IDE:  .ASCII /025 MACK026 SYM027 ETR/

IDE:  .ASCII /030 CAN031 EW 032 SUB/

IDE:  .ASCII /034 FS 035 GS 036 RS /

IDE:  .ASCII /037 US 038 DEL /

IDE:  .ASCII /055 NPCODE1:  .BYTE 0,2,6,20,21,22,23,24

IDE:  .ASCII /056 NPCODE2:  .BYTE 25,26,27,28,31,32,34,35

IDE:  .ASCII /057 NPCODE3:  .BYTE 36,37,177,95

IDE:  .ASCII /058 NPCODE4:  .BYTE 177
**PT3 -- Carriage Return Test**

THE LINE CONSISTS OF A STRING OF G'S AND
COPIES WHAT IS PAIRED OUT TO THE LAST
COLUMN WITH A SPACE SIMILAR TO THE
CHARACTER FIELD. THE SPACE IS SURROUNDED
PROPERLY AND THEN RETURNED FOR "MARKING". IFA
SPACE IS CONTROLLED, ALL SPACES BETWEEN
THE CHARACTERS HAVE BEEN FILLED.

**PT4 -- Multiple Line Feed Test -- 63 Line Feed Add**

A NUMBER IS PRINTED WHICH INDICATES THE NUMBER OF LINE
FEEDS. A REFERENCE LINE IS PRINTED.

**PT5 -- Carriage Return Test**

THE LINE CONSISTS OF A STRING OF G'S AND
COPIES WHAT IS PAIRED OUT TO THE LAST
COLUMN WITH A SPACE SIMILAR TO THE
CHARACTER FIELD. THE SPACE IS SURROUNDED
PROPERLY AND THEN RETURNED FOR "MARKING". IFA
SPACE IS CONTROLLED, ALL SPACES BETWEEN
THE CHARACTERS HAVE BEEN FILLED.

**PT6 -- Multiple Line Feed Test -- 63 Line Feed Add**

A NUMBER IS PRINTED WHICH INDICATES THE NUMBER OF LINE
FEEDS. A REFERENCE LINE IS PRINTED.
**PRINT TEST** - A ROW OF ALTERNATING "N'S AND SPACES ARE PRINTED OUT TO THE LAST COLUMN AND OVERPRINTED TWICE. THEN A SECOND ROW OF ALTERNATING "N'S AND SPACES FOLLOWED BY A THIRD ROW OF ALTERNATING "N'S AND SPACES.

**PRINT TEST**

<table>
<thead>
<tr>
<th>Action</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOV A, 67</td>
<td>15</td>
<td>Move accumulator A to 67</td>
</tr>
<tr>
<td>MOV A, 68</td>
<td>16</td>
<td>Move accumulator A to 68</td>
</tr>
<tr>
<td>PRINTC</td>
<td>17</td>
<td>Print character</td>
</tr>
<tr>
<td>MOV A, 91</td>
<td>18</td>
<td>Move accumulator A to 91</td>
</tr>
<tr>
<td>MOV A, 92</td>
<td>19</td>
<td>Move accumulator A to 92</td>
</tr>
<tr>
<td>JNZ 11</td>
<td>20</td>
<td>Jump if not zero to label 11</td>
</tr>
<tr>
<td>DEC A</td>
<td>21</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 17</td>
<td>22</td>
<td>Move accumulator A to 17</td>
</tr>
<tr>
<td>MOV A, 00</td>
<td>23</td>
<td>Move accumulator A to 00</td>
</tr>
<tr>
<td>DEC A</td>
<td>24</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 67</td>
<td>25</td>
<td>Move accumulator A to 67</td>
</tr>
<tr>
<td>MOV A, 68</td>
<td>26</td>
<td>Move accumulator A to 68</td>
</tr>
<tr>
<td>JNZ 11</td>
<td>27</td>
<td>Jump if not zero to label 11</td>
</tr>
<tr>
<td>DEC A</td>
<td>28</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 91</td>
<td>29</td>
<td>Move accumulator A to 91</td>
</tr>
<tr>
<td>MOV A, 92</td>
<td>30</td>
<td>Move accumulator A to 92</td>
</tr>
<tr>
<td>JNZ 11</td>
<td>31</td>
<td>Jump if not zero to label 11</td>
</tr>
<tr>
<td>DEC A</td>
<td>32</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 17</td>
<td>33</td>
<td>Move accumulator A to 17</td>
</tr>
<tr>
<td>MOV A, 00</td>
<td>34</td>
<td>Move accumulator A to 00</td>
</tr>
<tr>
<td>DEC A</td>
<td>35</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 67</td>
<td>36</td>
<td>Move accumulator A to 67</td>
</tr>
<tr>
<td>MOV A, 68</td>
<td>37</td>
<td>Move accumulator A to 68</td>
</tr>
<tr>
<td>JNZ 11</td>
<td>38</td>
<td>Jump if not zero to label 11</td>
</tr>
<tr>
<td>DEC A</td>
<td>39</td>
<td>Decrease accumulator A</td>
</tr>
<tr>
<td>MOV A, 91</td>
<td>40</td>
<td>Move accumulator A to 91</td>
</tr>
<tr>
<td>MOV A, 92</td>
<td>41</td>
<td>Move accumulator A to 92</td>
</tr>
<tr>
<td>JNZ 11</td>
<td>42</td>
<td>Jump if not zero to label 11</td>
</tr>
<tr>
<td>DEC A</td>
<td>43</td>
<td>Decrease accumulator A</td>
</tr>
</tbody>
</table>

**PRINT TEST** - A ROW OF ALTERNATING "N'S AND SPACES ARE PRINTED OUT TO THE LAST COLUMN AND OVERPRINTED TWICE. THEN A SECOND ROW OF ALTERNATING "N'S AND SPACES FOLLOWED BY A THIRD ROW OF ALTERNATING "N'S AND SPACES.
PT10: PRINTING FREQUENCY TEST -- 120 H'S ARE PRINTED ON 4 LINES

NEW DELAY = OLD DELAY + ( OLD DELAY X 120 )

-tag... THE TEST IS CANCELLED IF THE TEST AND RECORD COUNT IS SAME AS TEST DURATION.

--)

IF THE DELAY IS DECREASED IN THE SAME FASHION "Iron", PRINTC DEC PT11--

PT11: 10  TEST NUMBER

PB 00100 000000
00110 001000
00120 001100
00130 001010
00140 000100
00150 000010
00160 000000
00170 000000
00180 000000
00190 000000
001A0 000000
001B0 000000
001C0 000000
001D0 000000
001E0 000000
001F0 000000

PT12: 20  TEST NUMBER

PB 00200 000000
00210 001000
00220 001100
00230 001010
00240 000100
00250 000010
00260 000000
00270 000000
00280 000000
00290 000000
002A0 000000
002B0 000000
002C0 000000
002D0 000000
002E0 000000
002F0 000000

PT12A: 10  TEST NUMBER

PB 00300 000000
00310 001000
00320 001100
00330 001010
00340 000100
00350 000010
00360 000000
00370 000000
00380 000000
00390 000000
003A0 000000
003B0 000000
003C0 000000
003D0 000000
003E0 000000
003F0 000000

PT12A: 10  TEST NUMBER

PB 00300 000000
00310 001000
00320 001100
00330 001010
00340 000100
00350 000010
00360 000000
00370 000000
00380 000000
00390 000000
003A0 000000
003B0 000000
003C0 000000
003D0 000000
003E0 000000
003F0 000000

PT12A: 10  TEST NUMBER

PB 00300 000000
00310 001000
00320 001100
00330 001010
00340 000100
00350 000010
00360 000000
00370 000000
00380 000000
00390 000000
003A0 000000
003B0 000000
003C0 000000
003D0 000000
003E0 000000
003F0 000000

PRINTC DEC PT11--
PT17: LIFE TEST

PT17: LIFE TEST PRINTS 5 FULL LINES OF EACH PRINTABLE

PT17: LIFE TEST PRINTS 5 FULL LINES OF EACH PRINTABLE

PT17: LIFE TEST PRINTS 5 FULL LINES OF EACH PRINTABLE

PT17: LIFE TEST PRINTS 5 FULL LINES OF EACH PRINTABLE
CZLACEO LA36 TERM (OL11 & KL11) MACRO M1110 25-AUG-78 10:13 PAGE 53
LA36 PRINTER TESTS

SEQ 0108

CEPOS: CMP $4,91 JBG IT TIME TO INSERT PASS #7
CPNT: CMP $4,90 JPRINT A SPACE
PWRTC: PASHC $90 JPRINT MSG OF PASS COUNT
PWRTC: PASHC $90 JPRINT A SPACE
PWRTC: PASHC $90 JJUSTIFY ON POSTER BOARD
PWRTC: PASHC $90 JRUN A SPACE

15: RTX PC

00000000 000040 000001 000003 000004 000005 000006 000007 000008 000009

00000004 00000005 00000006 00000007 00000008 00000009 0000000A 0000000B 0000000C 0000000D

0000000E 0000000F 00000010 00000011 00000012 00000013 00000014 00000015 00000016 00000017

00000018 00000019 0000001A 0000001B 0000001C 0000001D 0000001E 0000001F 00000020 00000021

00000022 00000023 00000024 00000025 00000026 00000027 00000028 00000029 0000002A 0000002B

0000002C 0000002D 0000002E 0000002F 00000030 00000031 00000032 00000033 00000034 00000035

00000036 00000037 00000038 00000039 0000003A 0000003B 0000003C 0000003D 0000003E 0000003F

CEPOS: CMP $4,91 JBG IT TIME TO INSERT PASS #7
CPNT: CMP $4,90 JPRINT A SPACE
PWRTC: PASHC $90 JPRINT MSG OF PASS COUNT
PWRTC: PASHC $90 JPRINT A SPACE
PWRTC: PASHC $90 JJUSTIFY ON POSTER BOARD
PWRTC: PASHC $90 JRUN A SPACE

15: RTX PC

SEQ 0109

CZLACEO LA36 TERM (OL11 & KL11) MACRO M1110 25-AUG-78 10:13 PAGE 54
LA36 PRINTER TESTS

61100
CALL A 136 ECHO TESTS

.16PLA L336 ECHO TESTS

CHARACTER ECHO TESTS—ALL PRINTABLE AND
NON-PRINTING CHARACTERS TYPED ON THE KEYBOARD
AND DED TO DRIVE THE PRINTER. ONE CHARACTER AT
A TIME WILL CAUSE THE TEST TO BE
TERMINATED.

XXXXXXX

021:    251    15    GR
        211    76    B
466    01555    08079

CALL A 136 ECHO TESTS

022—LINE ECHO TEST SLOW RATE—SAME AS 021 EXCEPT
THAT A DELAY IS INTRODUCED BETWEEN CHARACTERS
TO PRODUCE A LCE ACTION

XXXXXXX

022:    251    15    GR
        211    76    B
466    01555    08079
THIS FOLLOWING TABLE IS USED BY TEST E023

<table>
<thead>
<tr>
<th>ASCII</th>
<th>ASCII</th>
<th>ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>114</td>
<td>ASCII /NUL /</td>
</tr>
<tr>
<td>117</td>
<td>110</td>
<td>ASCII /SOH /</td>
</tr>
<tr>
<td>124</td>
<td>130</td>
<td>ASCII /STX /</td>
</tr>
<tr>
<td>124</td>
<td>130</td>
<td>ASCII /ETX /</td>
</tr>
<tr>
<td>117</td>
<td>124</td>
<td>ASCII /EOT /</td>
</tr>
<tr>
<td>116</td>
<td>121</td>
<td>ASCII /ENQ /</td>
</tr>
<tr>
<td>103</td>
<td>113</td>
<td>ASCII /ACK /</td>
</tr>
<tr>
<td>104</td>
<td>114</td>
<td>ASCII /BEL /</td>
</tr>
<tr>
<td>123</td>
<td>040</td>
<td>ASCII /HS /</td>
</tr>
<tr>
<td>124</td>
<td>040</td>
<td>ASCII /HT /</td>
</tr>
<tr>
<td>106</td>
<td>040</td>
<td>ASCII /LF /</td>
</tr>
<tr>
<td>124</td>
<td>040</td>
<td>ASCII /VT /</td>
</tr>
<tr>
<td>106</td>
<td>040</td>
<td>ASCII /FF /</td>
</tr>
<tr>
<td>122</td>
<td>040</td>
<td>ASCII /CR /</td>
</tr>
<tr>
<td>117</td>
<td>040</td>
<td>ASCII /SO /</td>
</tr>
<tr>
<td>111</td>
<td>040</td>
<td>ASCII /SI /</td>
</tr>
<tr>
<td>114</td>
<td>105</td>
<td>ASCII /SO /</td>
</tr>
<tr>
<td>193</td>
<td>061</td>
<td>ASCII /DC1 /</td>
</tr>
<tr>
<td>193</td>
<td>062</td>
<td>ASCII /DC2 /</td>
</tr>
<tr>
<td>193</td>
<td>063</td>
<td>ASCII /DC3 /</td>
</tr>
<tr>
<td>193</td>
<td>064</td>
<td>ASCII /DC4 /</td>
</tr>
<tr>
<td>101</td>
<td>113</td>
<td>ASCII /DEL /</td>
</tr>
<tr>
<td>131</td>
<td>116</td>
<td>ASCII /FS /</td>
</tr>
<tr>
<td>124</td>
<td>102</td>
<td>ASCII /ES /</td>
</tr>
<tr>
<td>191</td>
<td>116</td>
<td>ASCII /CAN /</td>
</tr>
<tr>
<td>115</td>
<td>040</td>
<td>ASCII /EM /</td>
</tr>
</tbody>
</table>

EVEN
FILE ERRORS

FILE ERRORS-- SELECTED PATTERN ECHO TEST-- SELECT 1 TO 456
CHARACTER. EACH CHARACTER WILL BE ECHOED, A CONTINUING STRING UNTIL EITHER THE RETURN IS
SELECTED TO TERMINATE OR THE CHARACTER IS SELECTED.
CHARACTER OTHER THAN A GROUP WILL CAUSE A NEW STRING TO BE ECHOED.

SECTION TO OUTPUT CONTINUOUS STRING

OUTPU: CWO D2,800R
CHECK IF POINTER IS AT START OF TABLE
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)
CHECK IF COMMAND IS COMMAND IN TABLE
DEP. (00,00)}
START: ASCII C7>C2><ACRLEF><17>/CCLACED LA36 TERM (OLII & KLII)<ACRLEF>

ASCII /LA36 TERMINAL DIAGNOSTIC/<ACRLEF>

ASCII /OLII & KLII INTERFACE/<ACRLEF><12>

ENDPASS: ASCII <ACRLEF><12>/END OF PASS /

FAIL: ASCII /ERROR/<ACRLEF><12>

OLII: ASCII <ACRLEF>/CONSOLE & /

OLII: ASCII /80 DLII'S UNDER TEST/<ACRLEF><12>

PASS: ASCII /ERROR/<ACRLEF><12>/TEST #/

HORMSCI: ASCII C7>C2><ACRLEF><17><12>/TEST #/

ASCII /800 COLUMNS/<ACRLEF><12>
### MSC. DIAGNOSTIC MESSAGES

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1200</td>
<td>E025MA: ASCII /TYPE ANY PRINTABLE CHARACTER/</td>
</tr>
<tr>
<td>1300</td>
<td>ASCII /AND LISTEN FOR BELL.............../</td>
</tr>
<tr>
<td>1400</td>
<td>E025MB: ASCII &lt;ACRLF&gt;/NOT ENOUGH COLUMNS/&lt;ACRLF&gt;</td>
</tr>
<tr>
<td>1500</td>
<td>Echodn: ASCII &lt;ACRLF&gt;/ECHO TEST TERMINATED/&lt;ACRLF&gt;</td>
</tr>
<tr>
<td>1600</td>
<td>LImes: ASCII / /</td>
</tr>
<tr>
<td>1700</td>
<td>LImes: ASCII / /</td>
</tr>
<tr>
<td>1800</td>
<td>MESC3: ASCII &lt;ACRLF&gt;&lt;17&gt;&lt;12&gt;/SELECT TEST NUMER /</td>
</tr>
<tr>
<td>1900</td>
<td>Even: ASCII /EVEN/</td>
</tr>
<tr>
<td>2000</td>
<td>END</td>
</tr>
</tbody>
</table>

---

### MSC. DIAGNOSTIC MESSAGES

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2100</td>
<td>Opcode: ASCII /OOP /</td>
</tr>
<tr>
<td>2200</td>
<td>Opmeg: ASCII /TYPE ANY CHARACTER/</td>
</tr>
<tr>
<td>2300</td>
<td>Nmpsp: ASCII /USE SOFTWARE SWITCH REG AT MEMORY ADDR 176/&lt;7&gt;</td>
</tr>
<tr>
<td>2400</td>
<td>END</td>
</tr>
</tbody>
</table>

---

### MSC. DIAGNOSTIC MESSAGES

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2500</td>
<td>END</td>
</tr>
</tbody>
</table>