



PERMANENT
MEMORANDUM

Paul Sadriac

M -1121

PAGE 1 OF 5

DATE

August 9, 1961

SUBJECT Memory Addressing Test Program
TO PDP Distribution List

ABSTRACT

This is a maintenance program designed for checking memory address registers and decoders. The program checks for incorrect addressing and is capable, by use of sense switches, to continuously check any address designated by the test word switches.

FROM

Steve Lambert

APPROVED BY

Harlan E. Anderson

INTRODUCTION

To test memory address registers and decoders, all addresses not used by the address test program are loaded with their own address. That is to say, each register contains its own address. Then each register is checked for proper contents. Should an error be found, the address at which the error occurred will appear in the IO and its contents in the AC.

There are two control sense switches in this program. Sense Switch 1 is used if it is desired to continuously check one register and the contents of all other registers to be sure that only one register contains a given address. This is done by first clearing memory, then storing the address located in the test word switches at the address designated by the test word switches. Now, a check is made on memory to find all non-zero registers and compare them against the address set in the test word switches. If there is more than one address that contains the contents of the test word switches or a wrong address that contains the contents of the test word switches, then the computer will come to a halt indicating the address in the IO and its contents in the AC.

OPERATING INSTRUCTIONS

#1 Put up sense switch #2

#2 Load tape into reader, turn reader on and activate read in mode switch.

#3 Set address in test word switches and put up SS#1 if it is desired to check one register. SS#1 down, tests all addresses designated by the program.

#4 Select +10 margin switches of memory address registers and decoders. Vary the margins until the program stops. Record what happened in the computer log book.

#5 The first program read in starts at register 7700. To read in the low version, put SS#2 down. After the reader begins reading in the new program, put SS#2 up immediately. Otherwise, the reader will continue reading in after the new program has read in.

NOTES:

Sense switch 2 is used in this program to keep it in a closed

loop. Otherwise, a new program will be read in on top of the old.

There are two address checking routines. The high version starts at register 7700. The low starts at register 0.

The following indicates how address checking is done.

<u>Address</u>	<u>Contents</u>
0	0
1	1
2	2
.	.
.	.
.	.
100	100
101	101
102	102
.	.
.	.
.	.
1100	1100
1101	1101
1102	1102
.	.
.	.
.	.
7677	7677
7700	---- Program starts here

The address is checked against contents.

```
,Address checker test program
,7/24/61
,S. Lambert
,High Checker
org 7700
start          law 0          ,initial location
               dap 0 & 4
               dap check
               dap temp
               dzm * 0 & 1
               dap
               idx 0 - 1
               sas finish    ,final location
               jmp start & 4
```

```
check      lio temp      ,IO contains address
           lac
           sas temp
           hlt      ,incorrect address
           idx temp
           idx check
           sas trailend
           jmp check - 1
zhit       szs 10      ,check one reg. continuously
           jmp hit
           szs 20      ,read in new tape
           jmp start
read       rpb
           dio temp
           lac temp
           dap stop
           and stop
           sad stop
stop       jmp
           rpb
           dio * temp
           jmp read
hit        lac start    ,clear memory
           dap  $\phi$  & 2
           dap x & 1
           dzm
           idx  $\phi$  - 1
           dzm
           idx  $\phi$  - 1
           sas last
           jmp  $\phi$  - 3
           lat
           and stp
           sad stp
           jmp zhit
           lat
           dap  $\phi$  & 1
           dap
x          lac
           sza
           jmp  $\phi$  & 6
           idx  $\phi$  - 3
           lio  $\phi$  - 4    ,IO has address of reg. being checked
           sas trailend
           jmp x & 1
           jmp zhit
           sas * x
```

```
temp      hlt  
finish    jmp zhit  
stp       0  
last      dap 7700  
trailend  7700  
          dzm 7700  
          lac 7700  
jmp start end .
```