27
4-4-74
256 NTIS

UCID-17133

Lawrence Livermore Laboratory

UNDERSTANDING RAM AND ODT OPERATION
IN THE MCS-8 MICROCOMPUTER

C. W. Jensen

June 3, 1974

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the laboratory.

Prepared for U.S. Energy Research & Development Administration under contract No. W-7405-Eng-48.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

REFER QUESTIONS TO: C. W. Jensen	LAWRENCE RADIATION LABORATORY	ELECTRONICS ENGINEERING ENGINEERING NOTE LIVERMORE, CALIFORNIA	UNIVERSITY OF CALIFORNIA	LEN 22273
DATE: C. W. Jensen	UNDERSTANDING RAM AND ODT OPERATION IN THE MCS-8 MICROCOMPUTER			6/3/74
APPROVED: <i>[Signature]</i>				1 9

1.0 INTRODUCTION

This paper is written for you who are just beginning to work with the MCS-8 Microcomputer (and computers in general). The text is a guide to understanding some of the processes that occur when you write a program into RAM memory with the ODT*

2.0 RAM MEMORY

First, as you probably know, RAM memory for the MCS-8 microcomputer comes in two "packages". One type PC card has 256 words of 8-bit memory (LEA68-9068-97), while the other PC card has 1000 words of 8-bit memory (LEA68-9068-98). The random access feature means that when the microcomputer is turned ON, each of the memory locations will take on a random number. Each number can be altered to a desired number (the code for an alphabetical letter or numeral) by the operator. The method by which these numbers in memory are altered is by the ODT* program which is stored permanently in the PROM (Programmable Read Only Memory) located on the Memory Address and Storage card LEA68-9068-95. Also, the random access feature means that numbers can be written into and fetched from memory in a random manner.

Once RAM memory cards are installed in the microcomputer and a program has been written into memory, if you turn the computer OFF, the RAM memory will be wiped out. However, you can push the RESTART button and the memory will not be altered.

3.0 CHOOSING RAM MEMORY LOCATION

For the purpose of our discussion let us choose to write a clear text from RAM memory page 12₈** beginning with location 40₈. Further, let us decide to install two 8-Bit 256 Word Memory Cards LEA68-9068-97.

Before proceeding we must inspect the PC cards to see if either of them are jumpered for page 12. On the component side of these PC cards you will find a pair of jumpers to form a double-pole double-throw "switch". For our chosen page location, these "switches" should be jumpered A-C and D-F. This combination provides that the 2-bit binary code on lines A14, A15 will choose page locations 0 through 17. Next, look for the jumper at point "G". One of our two PC cards should have a jumper between "G" and the point labeled "12". With one PC card wired in this manner, page 12 can be chosen by typing "12S" on the teletype terminal. (Note there are the 16₁₀ pages--pages 0-17₈ that could be jumpered to "G" and, thus, wired for memory).

- (1) Work performed at Diablo Valley College during a course of private study #299. Submitted in accordance with P/O 4994805.

*Octal Debug Technique by E. R. Fisher and J. C. English (See LER72-103402)

**Note all references are to octal numbers unless otherwise noted. (12₈ = 10 in base ten.)

As mentioned in Section 2.0, we can load our program into RAM memory, execute it and debug it all with the ODT program stored in PROM memory. However, to do so, we need one page of RAM memory wired for page 10. Therefore, be sure one PC card LEA68-9068-97 is jumpered G to 10. The reason for this is that the ODT program uses the three high RAM memory locations of RAM 10 for the G (GO) command. With the two RAM cards wired--one to page 12 and one to page 10--we are ready to proceed in developing our program.

4.0 WRITING NUMBERS INTO RAM MEMORY

Before we actually input chosen "words" into RAM memory, let's run through the operation of the TTY (teletype) as it chooses and opens up a memory location.

Step	<u>THE OPERATOR INPUTS:</u>	<u>TTY RESPONDS WITH:</u>	<u>WHAT'S HAPPENING:</u>
A.	Pushes RESTART button	? (There is carriage return and line feed). Written hereafter CRLF	With this hardware, RESTART has been initiated and the ODT program has begun to execute as evidenced by ODT's first response-- a ? and CRLF
Step B.	12S	12S (also a CRLF)	ODT responds with the CRLF. The "S" is a set command to set a memory page address-- in our case, page 12. Computer is ready for the next command.
Step C.	40/	40/xxx	ODT responds with the contents of page 12, location 40, shown here as xxx this could be any group of octal numbers 000 through 377. If we want to see what is in the next higher memory location the operator----
Step D.	LF (LINE FEED)	PAGE LOCATION CONTENTS 012 041 xxx	ODT closes the last open address (40) and opens the next sequential address 41. Also, it prints the contents.
Step E.	LF	012 042 xxx	This action can be continued on through memory location 377. However, ODT can, also, change the contents in memory. This is done as follows.

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

MASTER

THE OPERATOR INPUTS:TTY RESPONDS WITH:WHAT'S HAPPENING/INSTRUCTIONS

Step

F. LF 001*
 *The zeros can be
 omitted by typing
 the "1" only.

012 043 xxx 001

ODT has written the desired
 number (here 001, for example)
 into memory location 43.

Step

G. LF 002

012 044 xxx 002

ODT has written another
 desired number into memory,
 this time in location 44.

We will now return to page
 12, location 40 and in-
 put our desired numbers
 that correspond to a chosen
 clear text.

Step

H. RESTART

? LF

ODT begins, again, to execute.
 Repeat steps B and C above.

40/xxx

ODT has opened up the address
 at location 40 and we are
 now ready to place numbers
 in memory according to the
 ASCII Code. We will discuss
 this, then return to our
 procedure here.

5.0 THE ASCII CODE

The ASCII Code is a standard 8-level code that provides compatability between teletype machines and digital equipment. The complete code can be found in many texts and it is part of other MCS-8 reference material. If you look at this code you will see that A = 301₈, B = 302₈-----etc. With this ASCII Code as reference, verify the code numbers for the example we have chosen to be written into RAM memory. It is:

PROGRAM IS WORKING!

Verify the code:

P	320
R	322
O	317
G	307
R	322
A	301
M	315
	240
I	311
S	323
	240
W	327
O	317
R	322
K	313
I	311
N	316
G	307
!	241

(All numbers are octal)

(space)

(space)

NOTE THAT THE CHOSEN WORDS HAVE 23 NUMBERS (WITH SPACES). REMEMBER THIS FACT AS IT IS A SIGNIFICANT FACTOR IN WRITING THE PROGRAM LATER.

Having found the corresponding ASCII Code (numbers) for each letter, we can proceed to write (input) these octal numbers into page 12. Starting at memory location 40 and continuing upward sequentially we can now write numbers that correspond to the letters in the words as we would read them.

Returning to part H. above:
THE OPERATOR INPUTS

TTY RESPONDS WITH

WHATS HAPPENING/INSTRUCTIONS

Step H.	320 LF 322	40/xxx 320 012 041 xxx 322	"P" is written into memory The contents of location 41 is replaced with "R"
	LF 317 ↓ ↓ ↓ LF 241	012 042 xxx 317 ↓ ↓ ↓ 012 062 xxx 241	Continue this sequence of pressing LINE FEED and entering the desired ASCII Code for "PROGRAM IS WORKING!"
	RESTART	? LF	Here we stop and check to see that proper entries have been made.
	12S	12S	
	40/	40/320	Memory location now has 320--a "P" stored in location 40.
	LF	012 041 322	ODT has stored an "R" in memory location 41.
	RESTART	? LF	Continue pressing LINE FEED until all octal numbers are verified through 062 location.
	RESTART	? LF	<u>NOTE: DO NOT TURN OFF COMPUTER:</u>

6.0 A QUICK SUMMARY

Thus far, what we have done is input a "clear text" into RAM memory. We could have started in any memory location and, if we had entered our "letters" beginning at 000, we would have written a message (including spaces) 377 characters (computer words) long.

The above operations can be summarized by showing the steps A through H, section 5.0, on a FLOW CHART. If a computer programmer were to show these operations he would have shown them, perhaps, as:

Flow Chart for writing (inputting)
data into RAM memory.

Our purpose, now, is to write a program so that when a proper command is given, the TTY will type out our desired message "PROGRAM IS WORKING!" This procedure is given in the sections that follow.

7.0 SOME NECESSARY INFORMATION ABOUT THE CPU

In the process of learning how our clear text is to be pulled from RAM memory page 12, locations 40-62, we must look at Intel's Central Processor chip--the 8008. Located on PC card LEA68-9068-93, this chip performs all the arithmetic operations in the microcomputer. It is the very "heart" of the system.

For our example, of particular importance is the block entitled "Accumulator and Scratch Pad Memory" located in the upper right of the 8008 Block Diagram, (See Figure 3, Intel User's Manual). There are seven registers in this Accumulator and Scratch Pad Memory. They are A (Accumulator) B, C, D, E, H, and L. Of these registers, we are interested in four--the H, L, A and C registers in that order.

To find and print our clear text now stored in memory, the computer has to do four operations (among some others!)

1. It has to find on what page the information is stored. This is done by addressing the H register. In our example we will address the H register to page 12.
2. It has to find the location of the information on the particular page. This is accomplished by addressing the L register. In our example, we will address the L register to location 40.
3. It has to print the information that is in RAM memory. This is executed by transferring (loading) the contents of RAM memory sequentially into A register, then printing the character.
4. Finally, it has to know when to stop. This is where register C is brought into action. Recall that in our example "PROGRAM IS WORKING!" we counted 23 characters? By telling register C we want 23 characters and no more, we let C register control the program length.

With regard to the above four operations, some more must be said about the second and fourth.

Even though we regard computers as being "smart", they are actually very dumb in that they must be told at each step in the solution of a problem just exactly what to do. In our instructions to them, nothing can be omitted. For this reason, our directions must be detailed and explicit. Therefore, we are going to direct the L register to location 40 and cause it to increment (increase) sequentially. As it does so, the "characters" (octal numbers) will be transferred, one at a time, from RAM memory to register A, then printed on the TTY.

During the incrementing of L register we will decrement (decrease) the number 23 stored in C register. In other words, L register will be counting up while C register is counting down. When C register reads zero we will stop the program. This can be shown as follows:

*Numbers are octal.

Operation terminates
with a stop command.

As these two registers count (L up, C down) the C register is looked at during each step to see if it has reached zero. When it does reach zero, the program stops because all characters in our text have been printed out.

8.0 WRITING A FLOW CHART FOR THE PROGRAM

Having this in mind, let us write a FLOW CHART that will result in our text "PROGRAM IS WORKING!" being printed. For the coding, refer to the paper "Octal Debugging Program" (ODT) MCS-8 Computer, LER 72-103402. Also, refer to the 8008 User's Manual, pages 8 and 9. Study the Index Register Instructions on page 8 and the notes at the bottom of page 9.

The Index Register Instruction tells us LrI means to load index register r immediately, ie, with the number that is typed next in sequence. The instruction code is 00 DDD 110. Look at the first step in the following flow chart. The command we want is LCI (Load C register immediate). Since we are concerned with index register C, the instruction code becomes 00 010 110 (See Note 1 at bottom of page 9). Converting the binary number 00 010 110 to octal, we get 26. Therefore, to load C register we type 26 on the teletype and follow by entering a LF, then 23.

With the above paragraph as a guide you can now relate each of the commands (LCI, LHI, LLI, LAM, INL, DDC) in the following FLOW CHART to the Index Register Instructions on page 8 of the 8008 User's Manual. By converting each binary number to an octal number you can see where each teletype input number comes from.

WHAT IT MEANS/INSTRUCTIONS

FLOW CHART

Press RESTART. Enter on TTY (10S) and (0/).
Note: After each entry press
LINE FEED.

This is the command "Load C Register
Immediate." Enter (26)--the code for LCI
Enter (23). This is the octal number
of characters in our text.

This is the comand "Load H Register
Immediate". Enter (56)--the code for LHI.
Enter (12). This is the page (high address).

This is the command "Load L Register
Immediate". Enter (66)--the code for LLI,
Enter (40). This chooses location 40 in RAM,
page 12.

This command is "Load Memory into A Register."
Enter (307)--the code for LAM.

Enter (55)--the code command to print the
character.

We want to increment the L register.
Enter (60)--the code to sequentially step
the L register upward.

We want to decrement the C register.
Enter (21)--the Code to sequentially step
the C register downward.

Is C register zero? If not, jump back and
print out the next higher sequential char-
acter. Jump instructions are in three
bytes. The first byte is the jump instruc-
tion itself. The second byte is the loca-
tion on the page; the third byte is the
page. Therefore, enter (110)--the code for
JUMP IF NOT ZERO. Enter (6)--the step
where our operation was LAM. Enter (10)--
the page. Enter (5)--code for RESTART ODT.
Hit RETURN. Enter (0G)--the GO command to
write the text.

9.0 SUMMARY

When the instructions opposite the FLOW CHART are followed the result should be a typed-out "PROGRAM IS WORKING!" For simplicity the steps in the flow chart are given below.

Press RESTART
Type (10 S) (0/)

<u>The steps are:</u>	<u>Type</u>	
0	26	Press LINE FEED
1	23	"
2	56	"
3	12	"
4	66	"
5	40	"
6	307	"
7	55	"
10	60	"
11	21	"
12	110	"
13	6	"
14	10	"
15	5	"

Type the GO Command: 0G

ACKNOWLEDGEMENT

I wish to thank Walter Bengé for his guidance and assistance in helping me to learn the ODT program.

C. W. Jensen

DISTRIBUTION

LLL Internal Distribution:

TID File 15

External Distribution:

TIC 27

Robert Swartz 1
Highland Park
Illinois 60035

Printed in the United States of America
Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy \$ ____*; Microfiche \$2.25

* Pages

1-50
51-150
151-325
326-500
501-1000

NTIS
Selling Price

\$4.00
\$5.45
\$7.60
\$10.60
\$13.60

NOTICE

"This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research & Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately-owned rights."

Technical Information Department

LAWRENCE LIVERMORE LABORATORY

University of California | Livermore, California | 94550