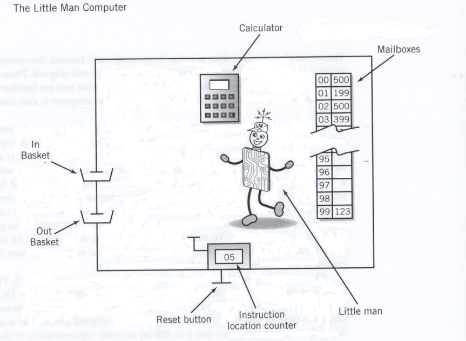
Chapter 6 The Little Man Computer

Consider this model of the LMC and answer the questions below.



Discussion Questions

*Please refer to this table of op codes for the discussion questions*

Opcode Definition

0 Halt

1 ADD

2 SUBTRACT

3 STORE

5 LOAD

6 BRANCH UNCONDITIONALLY

7 BRANCH ON ZERO

8 BRANCH ON POSITIVE

901 INPUT

902 OUTPUT

1) Using the LMC program below, add comments to explain what the result (value in the calculator) is after the completion of eachinstruction. The first one is completed as an example.

Mailbox Contents Result after completion

00 901 *Read contents from in basketand store in calculator*

01 319

02 901

03 320

04 219

05 709

06 518

07 902

08 000

09 517

10 902

11 000

----------------

17 DAT

18 DAT

19 DAT

20 DAT

Sol:

Mailbox Contents Result after completion

00 901 ***Read contents from in basketand store in calculator***

01 319 ***Value is stored in mailbox 19***

02 901 ***Read contents from in basketand store in calculator***

03 320 ***Value is stored in mailbox 20***

04 219 ***Subtracting value from calculator by stored value in 19***

05 709 ***if calculator is 0 branch to amilbox***

06 518 ***load value from mailbox 18 to calculator***

07 902 ***put this value in outbox***

08 000 ***coffee break***

09 517 ***load value from 17 to calculator***

10 902 ***move this value to outbox***

11 000 ***coffee break***

----------------

17 DAT

18 DAT

19 DAT

20 DAT

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

2) Refer to the LMC program in question (1). Suppose the contents of mailbox 17 = 5; contents of mailbox 18 = 1.

a) What is the final value in outbasket if the first in basket is 56 and second in basket is 89?

b) What is the final value in outbasket if the first in basket is 75 and second in basket is 75?

c) What is the final value in outbasket if the first in basket is 89 and second in basket is 56?

Sol: The order of input does not matter.

a) ***5***

b) ***1***

c) ***5***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

3) Describe what the LMC program in question (1) does. Suppose the contents of mailbox 17 = 5; contents of mailbox 18 = 1.

Sol: ***When the program executes there are two outcomes from the [rogram given that is if values are different output is 5 or it will be 1.***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

4) Refer to the LMC program below and the table of op codes given above.

a) What is the first number placed in the outbasket?

b) What is the last number placed in the outbasket?

Mailbox Contents

00 517

01 218

02 902

03 705

04 601

05 000

……………..

17 100 DAT

18 2 DAT

Sol:

a) ***98***

b) ***0***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

5) Refer to the LMC program in question (4). Change DAT in mailbox 18 to 4.

a) What is the first number placed in the out basket?

b) What is the last number placed in the out basket?

Sol:

a) ***96***

b) ***0***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

6) Describe what the LMC program in question (4) does.

Sol: ***It is a program of displaying even numbers in the output basket. It displays even numbers in backward order starting from 98 to 0 only even digits.***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

7) Refer to the LMC program below. Writedown what the calculator will hold after the instruction is complete in each loop. The first one is completed as an example.

Mailbox Contents Calculator after instruction is complete

Loop1 Loop2 Loop3 Loop4

00 517 *1 2 3 4*

01 118

02 317

03 219

04 710

05 600

……………..

17 1 DAT

18 1 DAT

19 5 DAT

Sol:

Mailbox Contents Calculator after instruction is complete

Loop1 Loop2 Loop3 Loop4

**00 517** **1 2 3 4**

**01 118 2 3 4 5**

**02 317 2 3 4 5**

**03 219 -3 -2 -1 0**

**04 710 -3 -2 -1 0**

**05 600 -3 -2 -1 NA**

……………..

17 1 DAT

18 1 DAT

19 5 DAT

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

8) Refer to the LMC program in question (7). How did the contents of mailboxes 17-19 change for each loop?

Sol:

***Contents of mailbox 17 is 2 3 4 5.***

***While other contents do not change.***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

9) What instruction should be placed in mailbox 02 so the program loops 4 times? Refer to the table of op codes above.

Mailbox Contents

00 517

01 218

02 ???

03 317

04 902

05 600

06 000

……………..

17 10

18 2

Sol: ***In mailbox 02 instruction 706 should be placed for the program to loop 4 times.***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

10) Describe what the following LMC program does. Refer to the table of op codes above.

Mailbox Contents

00 901

01 309

02 207

03 902

04 708

05 602

06 000

07 1

08 000

09 DAT

Sol: ***The program first takes a number as input from the user then loops the program with the input. It then prints the number backwards in the basket following all the number before the input. For eg. Input Is 5 basket shows 4-3-2-1-0.***

Section 6.2 Operation of the LMC

Section 6.4 An Extended Instruction Set

11) The contents in memory occasionally have to be moved to another area of memory. When that happens, the mailbox references must be adjusted so that the program continues tofunction properly. Rewrite the LMC code in problem (10) so that it occupies mailboxes 05 through 14 only; mailboxes 01 through 04 will be used by another program, so they can't be used. Assume that instruction 605 remains in mailbox 00.

Mailbox Contents

00 605

01 used by other program

02 used by other program

03 used by other program

04 used by other program

05 ???

06 ???

07 ???

08 ???

09 ???

10 ???

11 ???

12 ???

13 ???

14 ???

Sol:

Mailbox Contents

00 605

01 used by other program

02 used by other program

03 used by other program

04 used by other program

05 901

06 314

07 212

08 902

09 713

10 606

11 000

12 1

13 000

14 DAT

12) Describe the LMC three-digit instruction format. How does the LMC know what part of the value is an instruction, and what part is an address?

Sol:

the Little Man to do some useful work. For this purpose, we have invented a small group of instructions that he can perform. Each instruction will consist of a single digit. We will use the first digit of a three-digit number to tell the Little Man which operation to perform. , the operation will require the LittleMan to use a particular mailbox to store or retrieve data. Since the instruction only requires one digit, we can use the other two digits in a three-digit number to indicate the appropriate mailbox address to be used as a part of the instruction. Thus, using the three digits on a slip of paper, we can describe an instruction to the Little Man according to the following diagram: 3 | 25 instruction | mailbox address The instruction part of the three-digit code is also known as an “operation code”, or op code for short. The op code number assigned to a particular instruction is arbitrary, selected by the computer designer based on various architectural and implementation factors. The op codes used by the author conform to the 1979 version of the Little Man Computer model. The opcode consist of XYY format where X lies between 0-9 and YY lie between 0-99.

Section 6.2 Operation of the LMC

13) How does the LMC "know" if a particular mailbox contains data or instructions?

Sol:

This cycle, which is similar for all the instructions, can be broken into two parts:

1. The fetch portion of the cycle, in which the Little Man finds out what instruction he is to execute,
2. The execute portion of the cycle, in which he actually performs the work specified in the instruction. The fetch portion of the cycle is identical for every instruction. The Little Man walks to the location counter and reads its value. He then goes to the mailbox with the address that corresponds to that value and reads the three-digit number stored there. That three-digit number is the instruction to be performed.

The fetch portion of the cycle has to occur first: until the Little Man has performed the fetch operation, he does not even know what instruction he will be executing. all require the Little Man to move data from one place in the mailroom to another. The first four instructions all involve the use of a second mailbox location for the data. The LOAD instruction is typical. First, the Little Man fetches the instruction. To perform the execute phase of the LOAD instruction, the Little Man first looks at the mailbox with the address that is contained in the instruction. He reads the three-digit number on the slip of paper in that mailbox and returns the slip of paper to its place.

Section 6.5 The Instruction Cycle

14) What happens if the LMC is executing a program and never encounters a "HALT" command?

Sol: The Little Man walks to the location counter and reads its value. He then goes to the mailbox with the address that corresponds to that value and reads the three-digit number stored there. That three-digit number is the instruction to be performed. Little Man Computer follows some specific steps when it comes to perform instruction set which is also known as instruction cycle. This cycle has two parts of execution:

1)The fetch portion of the cycle, in which the Little Man finds out what instruction he is to execute.

2)The execute portion of the cycle, in which he actually performs the work specified in the instruction. With the exception of the step in which the Little Man increments the location counter, the steps must be performed in the exact sequence. The fetch steps must occur before the execution steps within the fetch, the Little Man must look at the location counter before he can pull the instruction from its mailbox. For eg.

The LMC follows following format:

1)The little man computer reads the address from the location counter

2)Then walks over to the mailbox that corresponds to the location counter

3) After that and reads the number on the slip of paper.

4)He then puts the slip of paper back, in case he should need to read it again later.

So, as per the LMC rules if no HALT is found in the program LMC will continue to execute the program as it will not find 000 to stop executing. Where 000 is HALT instruction even if it is data. LMC will continue to execute until a location 000 is found by the executer or location which ends with 4 which is also an invalid operation code. Hence, it assumes that incrementing the program counter past 99 returns it to zero

Section 6.2 Operation of the LMC

15) Describe how the LMC is von Neumann architecture.

Sol: We need three things:

1) Memory holds both programs and data; this is known as the stored program concept. The stored program concept allows programs to be changed easily.

2) Memory is addressed linearly; that is, there is a single sequential numeric address for each and every memory location.

3) Memory is addressed by the location number without regard to the data contained within.