Lab Manual for Computer Organization and Assembly Language

Lab-6

Loops

Table of Contents

1. Introduction

The Loop instruction provides a simple way to repeat a block of statements a specific number of times. CX is automatically used as a counter and is decremented each time the loop repeats.

2. Objective

- To get basic understanding of assembly language and loops.
- To write assembly programs and learn how to use loop statements.
- To get an understanding of identifying basic errors regarding loops.

3. Concept Map

The Loop instruction provides a simple way to repeat a block of statements a specific number of times. CX is automatically used as a counter and is decremented each time the loop repeats.

The JMP instruction can be used for implementing loops. For example, the following code snippet can be used for executing the loop-body 10 times.

```
MOV     CL, 10
L1:
<LOOP-BODY>
DEC     CL
JNZ     L1
```

The processor instruction set, however, includes a group of loop instructions for implementing iteration. The basic LOOP instruction has the following syntax −

```
LOOP    label
```

Where, *label* is the target label that identifies the target instruction as in the jump instructions. The LOOP instruction assumes that the **ECX register contains the loop count**. When the loop instruction is executed, the ECX register is decremented and the control jumps to the target label, until the ECX register value, i.e., the counter reaches the value zero.

The above code snippet could be written as –

```
mov ECX,10
l1:
<loop body>
loop l1
```

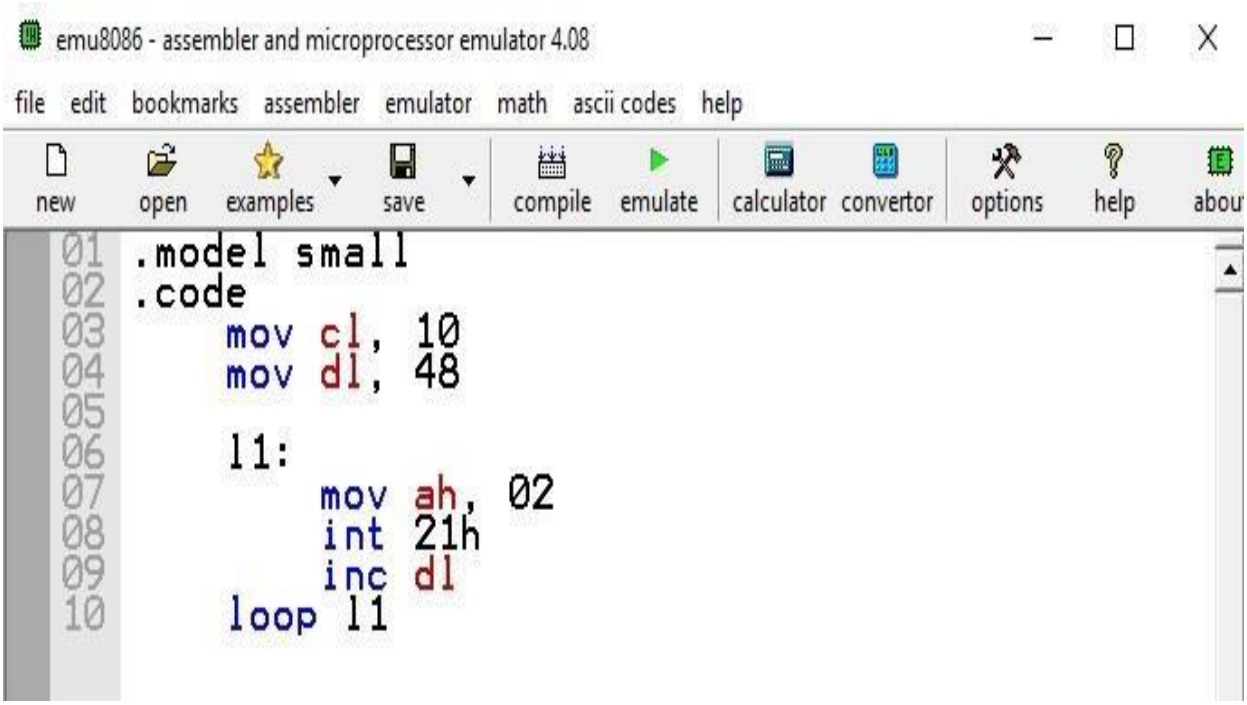
The execution of the Loop instruction involves two steps:

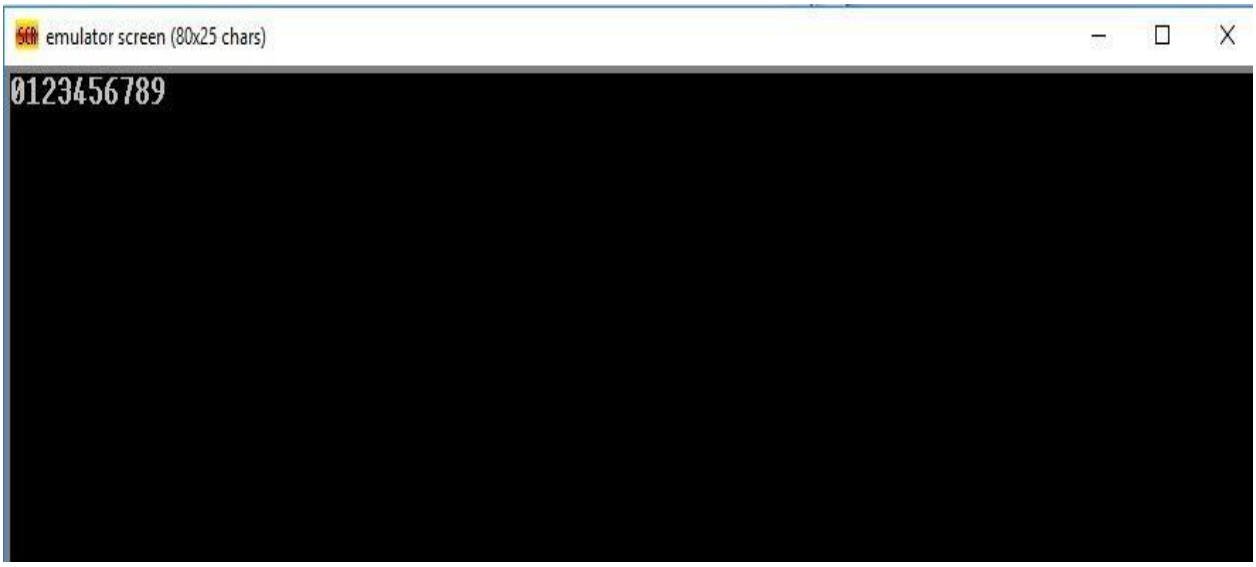
- First, it subtracts 1 from ECX.
- Next, it compares ECX to zero.
- If CX is not equal to zero; a jump is taken to the label identified by destination.
- Otherwise, if CX equals zero, no jump takes place and control passes to the instruction following the loop.

4. Walk Through Task

4.1 Program 1

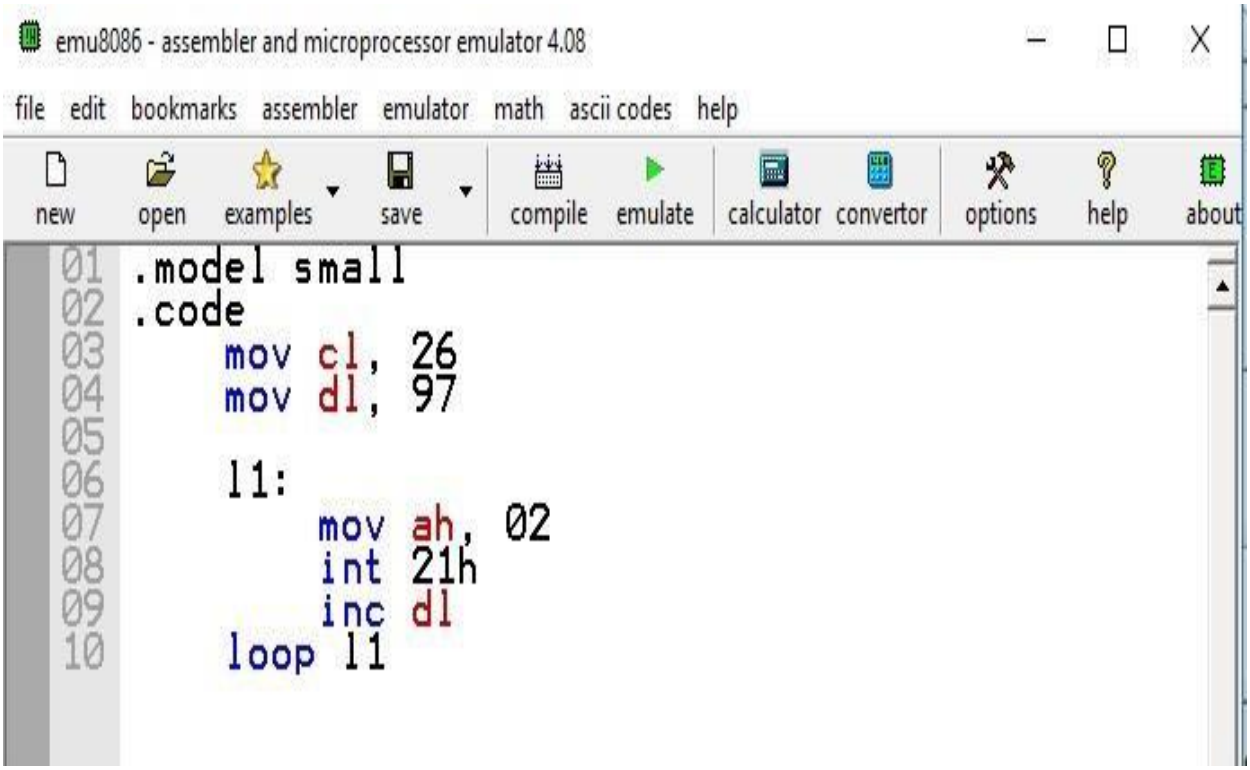
To print digits from 0 to 9 using loop.

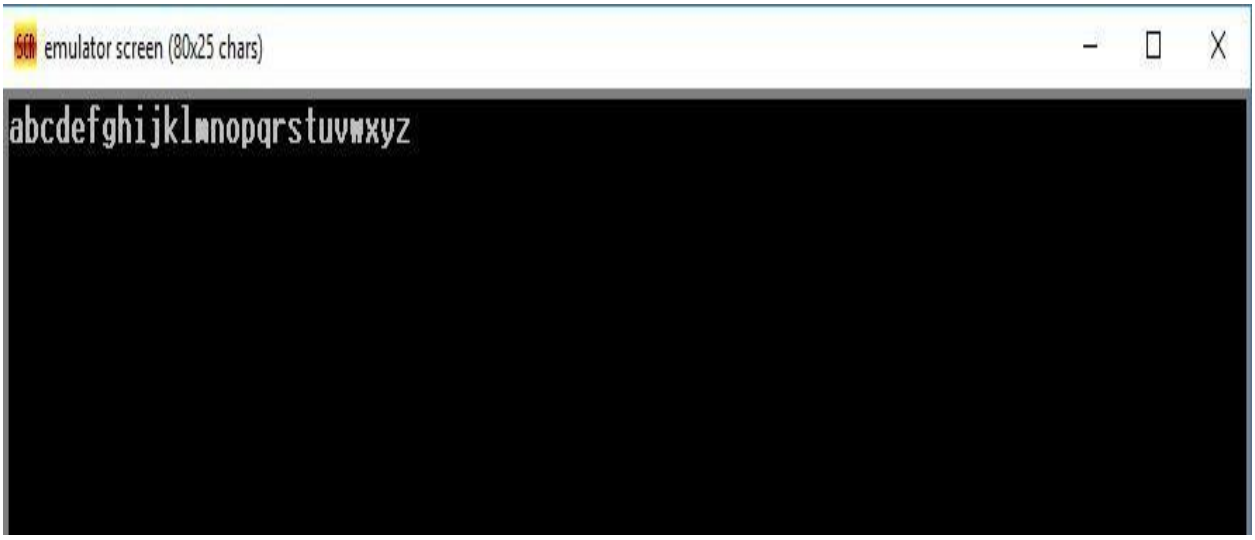




4.2 Program 2

To print small letter alphabets (a to z) using loop.





5. Procedure& Tools

In this section you will know about the assembler used in the course.

5.1 Tools

- Download emu 8086 from (http://www.emu8086.com/files/emu8086v408r11.zip)
- Just extract the emu8086.15.zip on C
- Install emu8086

6. Practice Tasks

This section will provide more practice exercises which you need to finish during the lab. You need to finish the tasks in the required time. When you finish them, put these tasks in the following folder:
\\fs\assignments$

6.1 Practice Task 1 [Expected time = 15mins]

Write an assembly language program using the Loop instruction to print all letters from A to Z.

6.2 Practice Task 2 [Expected time = 15mins]

Write an assembly language program using the Loop instruction to print all letters from a to z.

6.3 Practice Task 3 [Expected time = 10mins]

Write a program to that reads 10 numbers from user and print the highest number. Use loop to read number 10 times and compare number with the highest in loop?

6.4 Practice Task 4 [Expected time = 10mins]

Write a program to print 0-9 in reverse order?

6.5 Practice Task 5 [Expected time = 10mins]

Write a program to print first five odd numbers in reverse order?

6.5 Practice Task 6 [Expected time = 10mins]

Write a program to print first five even numbers in reverse order?

7. Out comes

After completing this lab, student will be able to implement and solve loop problem in assembly language.

8. Evaluation Task (Unseen) [Expected time = 30mins for tasks]

The lab instructor will give you unseen task depending upon the progress of the class.

9. Evaluation criteria

The evaluation criteria for this lab will be based on the completion of the following tasks. Each task is assigned the marks percentage which will be evaluated by the instructor in the lab whether the student has finished the complete/partial task(s).

Table 3: Evaluation of the Lab

Sr. No.	Task No	Description	Marks
1	4	Problem Modeling	20
2	6	Procedures and Tools	10
3	7	Practice tasks and Testing	35
4	8	Evaluation Tasks (Unseen)	20
5		Comments	5
6		Good Programming Practices	10

10. Further Reading

This section provides the references to further polish your skills.

10.1 Slides

The slides and reading material can be accessed from the folder of the class instructor available at \\fs\lectures$\