Computer Organization & Assembly Language

(Lecture 07)

Omar Bin Samin

Lecturer

Institute of Management Sciences, Peshawar.

Loop Instructions

- Loop instructions are used to execute the given instructions for number of times

- There are two types of loop instructions:

1. Unconditional Loop Instructions
2. Conditional Loop Instructions

Unconditional Loop Instructions

- These instructions are used to loop a group of instructions until the condition satisfies, i.e. CX = 0

- Syntax:

  loop tag

- Consider the following piece of code:

```
[org 0x0100]
xor ax,ax
mov bx,Tag1

add ax,[bx]
add bx,2
add ax,[bx]
add bx,2
add ax,[bx]
add bx,2

mov [Tag2],ax
mov ax,0x4c00          ; exit
int 0x21

Tag1: dw 5, 10, 15
Tag2: dw 0
```

Same instructions are repeated 3 times

- Same task as discussed in the previous slide can also be implemented using unconditional loops. Consider the following piece of code:

```
[org 0x0100]
xor ax,ax
mov bx,Tag1
mov cx,3

Tag3:
add ax,[bx]
add bx,2
loop Tag3

mov [Tag2],ax
mov ax,0x4c00          ; exit
int 0x21

Tag1: dw 5, 10, 15
Tag2: dw 0
```

Task 01

- Show values of AX, BX and CX for all executable instructions.

```
[org 0x0100]
xor ax,ax
xor bx,bx
xor cx,cx
mov al,1
mov bl,7
mov cl,3

Tag1:
add al,bl
loop Tag1

mov ax,0x4c00   ; exit
int 0x21
```

Task 02

- Generate the table of 2 using mul operation and unconditional loop instruction such that the program exits when destination register value is equal to **000A**.

Conditional Loop Instructions

- In these types of instructions, the processor must check for the particular condition. If it is true + CX !=0, only then the loop executes

- The arithmetic and logic operations set flags

- The conditional loop statements tests the flag and jump if the relevant flag is set along with CX !=0

- Conditional loop instructions are as follows:
  - LOOPE / LOOPZ
  - LOOPNE / LOOPNZ

LOOPE / LOOPZ

- LOOPE stands for ‘Loop if Equal’

- LOOPZ stands for ‘Loop if Zero’

  - Decrements CX
  - If CX !=0 and ZF = 1, then jumps to label/tag
  - Else, do nothing

LOOPNE / LOOPNZ

- LOOPNE stands for ‘Loop if Not Equal’

- LOOPNZ stands for ‘Loop if Not Zero’
    - Decrements CX
    - If CX !=0 and ZF = 0, then jumps to label/tag
    - Else, do nothing

Example

- Show values of AX, BX, CX and ZF for all executable instructions.

```
[org 0x0100]
mov al,1
mov bl,2
mov cl,5

Tag1:
add al,bl
cmp al,3
loopz Tag1

Mov ax,0xFFFF

mov ax,0x4c00      ; exit
int 0x21
```

Solution

AX		BX		CX		ZF
AH	**AL**	**BH**	**BL**	**CH**	**CL**	
00	01	00	00	00	00	0
00	01	00	02	00	00	0
00	01	00	02	00	05	0
00	03	00	02	00	05	0
00	03	00	02	00	05	1
00	03	00	02	00	04	1
00	05	00	02	00	04	0
00	05	00	02	00	04	0
00	05	00	02	00	03	0
FF	FF	00	02	00	03	0

Task 03

- Generate the following sequence using conditional loop instruction “loopne”:

1

3

9

1B

51