# Computer Organization & Assembly Language

BS (CS) Fall 2024

Lab 06 Manual



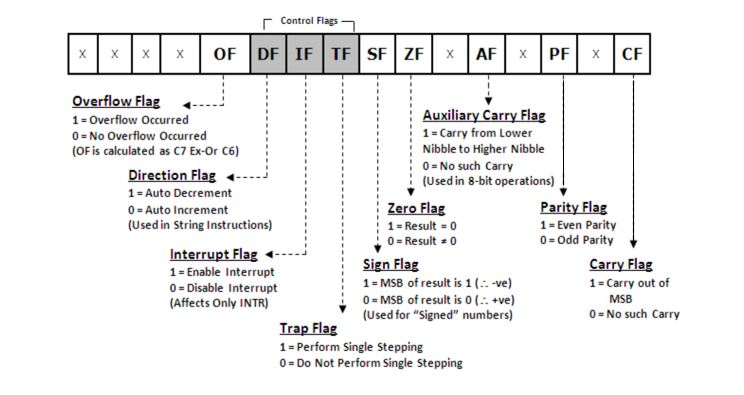
# Learning Objectives:

1. Recap of Flag Register, Jump instruction (JMP) and Loops

# **Lab 06**

# Flag Register:

These registers are used to store information about the state of the processor and the outcomes of various operations. Flag registers contain individual bits, each representing a specific condition or status of the CPU.



There are two types of flags

1. Status flags
2. Control flags

Status Flags:

* Allow the results of one instruction to influence later instructions.
* The arithmetic instructions use OF, SF, ZF, AF, PF, and CF
* Carry flag: Carry flag is set if carry is generated for unsigned integers.
* Overflow flag: Overflow flag is set if overflow is generated for signed integers.
* Zero flag: Zero flag is set if a zero is produced, as a result or as a comparison of two integers.
* Sign flag: Sign flag is set to one if the MSB of the destination operand is 1.

Controls flags:

These control flags serve specific purposes in controlling the operation of the CPU. Control flags in a CPU are essential for managing program execution, controlling interrupts, making decisions based on conditions, handling errors.

* Trap Flag (TF): When TF is set, the CPU enters single-step mode, where it executes one instruction at a time and generates a trap interrupt (interrupt 1) after each instruction.
* Interrupt Flag (IF): When IF is set, the CPU allows mask able interrupts to be processed. When it's cleared, mask able interrupts are disabled, preventing the CPU from responding to external interrupt requests.
* Directional Flag (DF):It control the direction of data movement in string operations. When DF is set, data moves from higher memory addresses to lower ones, and vice versa.

# Jump Instruction:

JMP is an unconditional jump command that is use to jump to a specific line. That line or block of code is identified by a label. Label can be any word other than the names of the variable used in the code or the keywords.

e.g., *START:*

*mov al,05*

*JMP START*

# Loops:

The keyword ‘loop’ runs a static loop of specific number of times and that number is stored in the Counter register, CX.

e.g., *mov cx,5*

*mov al,0*

*start:*

*inc al*

*loop start*

## Tasks

1. Write an assembly language program
   * 1. To add two unsigned numbers and observe which flags are changed
     2. To add two signed numbers and observe which flags are changed
     3. To subtract two unsigned numbers and observe which flags are changed
     4. To subtract two signed numbers and observe which flags are changed
2. Write an assembly language program to reverse an array using loop and indirect addressing.
3. Consider two binary numbers, A = 1011 and B = 1101. Perform addition, subtraction operation. Provide the binary result along with flags value. Explain why flags is set or not set.
4. Take two binary numbers, X = 1110 and Y = 0101. Execute a decrement, addition, subtraction, and Increment. Provide the binary result along with flags value. Explain flags is set.
5. Show the ASCII code of character a-z using loops.

## Submission Instruction

Paste the screenshot below each task in the word file.

Submit the word file (having all the screenshots) and .asm files in a zip folder 0in GCR’s submission tab.