

---

---

# 8051 Microcontroller Assembly Language Programming

Prepared by : A. B. Modi  
Target Audience : 5<sup>th</sup> Semester Students

---

---

# Learning Objectives

After learning this chapter, Students should be able to:

- Basic concepts required for Assembly Language programming
- Describe basic concepts of Assembly language programming
- Explain why Assembly Language Programming is used
- Describe the assembly language programming process

# Basic Concepts

Hardware Concepts

Computer Concepts

Software Concepts

# Hardware Concepts

- ❖ Central Processing unit
- ❖ Memory
  - ROM
  - RAM

# Software Concepts

## ❖ Code numbers

- Program operation (instruction)
- Opcode
- Operand
- Code memory
- CPU fetches Code Numbers

## ❖ Data Memory

- Addresses in Memory meant to be read or written by CPU
- Data Address contains Data Number

## ❖ Assembly Language Mnemonics

- Program is written in mnemonics, then it is translated to Machine Code
- A text editor program, an assembler program, a program that operates the computer

# Computer Concepts

- ❖ A computer model
  - Memory Organization
  - Address lines
  - Data lines
- ❖ Starting Up
  - Resetting or booting
- ❖ Machine Language
  - Unique bit pattern to perform a specific operation
  -

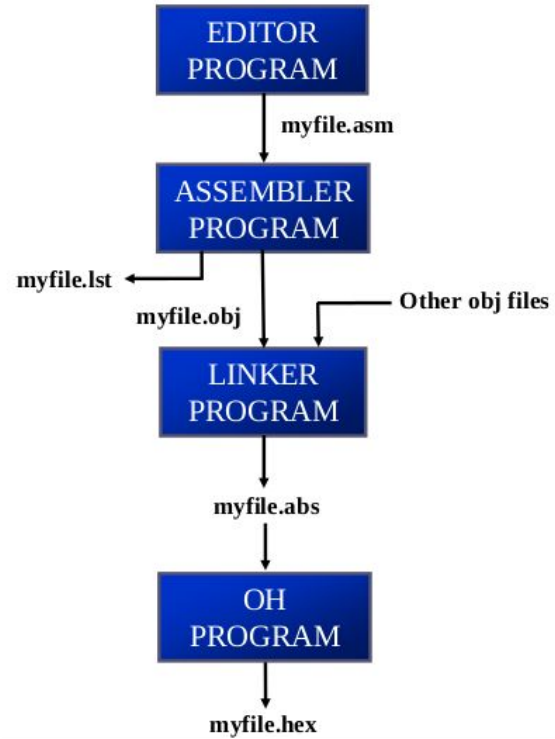
# Why use Assembly Language Programming?

1. To speed computer operation.
  - a. Program can be stored compactly in code memory
  - b. Compilers produce excess or over-head code
2. To reduce the size of program.
  - a. No excess code is required, because of awareness of the exact needs of the program
3. To write program for special situation.
  - a. Speed of response is critical, assembly-coded programs execute rapidly because of the exact fit of program code to task requirements
4. To save money.
  - a. Reducing code size reduces the cost of associated ROM chips
5. To better understand how computer operates.

# Assembly Language Programming Process

1. An operating system program
2. A word processing
  - a. Written using Text editor with mnemonics
  - b. Stored with .ASM file extension
3. An assembler program
  - a. Takes .ASM file and converts it to .OBJ file
4. A testing program
  - a. Utility programs for testing programs is called debuggers or simulators





# Basics

- ❑ Manual and automated assembly both use a text file as an input and produce a hex file as an output.
- ❑ The program must begin at 0000h and proceed sequentially.
- ❑ Instructions are stored with opcode first, then the data operands are stored in next code memory.
- ❑ For 2-byte number is stored low byte first in code memory, then high byte next in code memory.

- ❑ Understanding the problem to be solved
- ❑ Designing the program
- ❑ Writing the program
- ❑ Testing the program

# Assembly Language Instruction method

Label: Instruction ;comments

Instruction Part: mnemonic destination, source  
/opcode (operands)