LAB 1: Intel 8051 CPU PROGRAMMING
DATA TRANSFER INSTRUCTIONS

OBJECTIVES

At the end of the laboratory works, you should be able to write simple assembly language programs for the Intel 8051 CPU using data transfer instructions and execute the programs on the Intel 8051 trainer board.

INTRODUCTION

The Intel 8051 Central Processing Unit has 8-bit data bus and a 16-bit address bus which is capable of accessing a linear address space of 64K Byte. The Intel 8051 microcontroller/microprocessor consists of two types of internal memory; 4K ROM and 128 RAM. The Intel 8051 Trainer Board consists of 256K External ROM and 256K External RAM. The start address for ROM is 0000H and starting address for RAM is 2000H. User programs are written into the External ROM. This memory starts at address 0000H up to 1FFFH. Therefore, user programs should be started at address 0000H.

In order to write assembly language programs for the Intel 8051 CPU, it is important to understand the programming model of the CPU, or also known as the register set.

![Programming Model](image)

Figure 1 68000 Programming model

The Intel 8051 CPU have Accumulator A and B, 8 data registers, R0 to R7, Program Status Word (PSW), status register (SR), Program Counter and Data Pointer as shown in Figure 1. All registers are 8 bit except for the Program Counter and Data Pointer are 16 bit in length. The data registers are used to store information (data) within the Intel 8051 CPU itself. Accumulator A is used for all Arithmetic and logic instructions.
There are three types of data transfer instructions provided by Intel 8051 microprocessor:

a) MOV, MOVc, MOVX  
b) PUSH and POP  
c) XCH and XCHD

Data transfer instructions involve instructions such as MOV, MOVc, MOVX to copy or move data between the various locations.

**EQUIPMENTS**

1. A Personal computer installed with the MCU 8051 IDE Editor/Assembler/Simulator  
2. Intel 8051 Trainer Board.

**PROCEDURE**

All the programs in the exercises have to be written and assembled using the MCU8051IDE Assemble/simulator. To observe the results, the programs have to be compiled and executed in the Simulator. The contents of the affected registers and memory locations have to be examined through single stepping or breakpoints setting.

**EXERCISES**

1. Write a program that will store 30H into Accumulator A, 48H into R0 and D8H into R1. Execute the program and explain the differences in the results.

2. Write a program that will store 2004H in DPTR. Execute the program and comment on the results.

3. Write and execute a program that will store data 21H into external memory location 2050H. Explain the results.

4. Write a program that will send logic 11010011B to R5. Execute the program and comment on the results.

5. Write a program that will exchange data between R0 and ACC A. Execute the program and comment on the results.