LAB 3: Intel 8051 CPU PROGRAMMING
PROGRAM LOOPS AND SUBROUTINES

OBJECTIVES

At the end of the laboratory work, you should be able to write the Intel 8051 assembly language program using looping technique and subroutines.

INTRODUCTION

The programs considered in the previous laboratory exercises so far are all the straight line, sequential variety. In practice, programs almost invariably involve the repetition of sections of the routine in order to achieve a solution or complete a control process. Such repetitions are known as loops and it is in these that the real power of microprocessor lies. The flowchart of generalized loop procedure is shown.

The loop normally requires setting up initial values for use by the process and also by the loop, for continuing and terminating the loop. The number of loops executed is stored in the loop counter.
The process is the “compound instructions” that will be executed repeatedly when the loop repeats itself. Extra care must be exercised to formulate these instructions.
The instruction used to test for terminating the loop is called conditional instructions, and is of the form Jcc, where J stands for Jump and cc is the condition. Some commonly used conditional instructions are JZ, JNZ, JC, JNC, JB, JNB, CJNE and DJNZ.

Subroutine is a technique of writing programs so that it gives the advantages of using less memory space for programs and the modular program structure which leads to easier debugging.

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 MCU 8051 IDE assembler/simulator. To observe the results, the programs have to be compiled and simulate. The contents of the affected registers and memory locations have to be examined through single stepping or breakpoints setting.
EXERCISES

1. Write and execute a program that copy twelve bytes of data starting from memory locations 1000H to memory locations starting 100BH.

2. Write and execute a program that place the larger of the contents of memory locations 1200H and 1250H into memory location 1300H.

3. Three hexadecimal numbers are stored in memory locations 1010H, 1011H and 1012H. Sort these numbers into ascending order with the smallest number in memory location 1013H.

4. Write a program, which inspect the contents of memory location 1008H and, if the contents are greater than 0AH, subtract the value with 05H and store the result in memory location 200AH. Otherwise, add it with 10H and store the result in location 200BH.

5. Twelve 8-bit numbers are stored in memory locations starting with 1250H. Add the content of these memory locations using indirect addressing mode and looping technique. Execute the program and explain on the result obtained.

6. Write and execute a program that compute the area of two rectangular whose width and length are stored in memory locations 2100H and 2101H, and save the area of these rectangular in memory locations 2200H and 2300H. The calculation process of the area must be done in a subroutine.