# Lab 2

### Ex.No.2 Arithmetic operation using 8086

### 2.1 Introduction:

The purpose of this experiment is to learn about the registers, instruction sets, and arithmetic operators of 8086 by addition, subtraction, multiplication and division for the given two 16-bit numbers and store them in a memory location.

### 2.2 Hardware Requirement:

The 8086 Microprocessor kit, Power Supply.

### 2.3 Program Logic:

The add instruction requires either the addend or the augend to be in a register, irrespective of the addressing modes. Consider we have two 16-bit numbers in AX and BX registers. Addition is performed using ADD instruction in 8086 microprocessor. The result is transferred to the destination memory location.

Similar to addition operation subtraction is performed using SUB instruction in 8086 microprocessor.

The 8086 processor provides both signed and unsigned multiplication in their instruction set to overcome the loss of efficiency in performing the repeated addition. The MUL instruction can have both 16-bit and 8-bit operands and the multiplicand should be in AX or AL register, accordingly the result for a byte multiply is a 16-bit number in AX while that for a word multiply is a 32-bit number, the lower word of which is in AX and the higher word in DX.

To perform division, the divisor should be in AX register and dividend can be in any register. Using DIV instruction, the operation is performed and the results will be available in AX and DX registers. In AX we have the quotient and in DX remainder. The results are moved to the memory locations.

### 2.4 Program

**Addition without carry:**

| ADDRESS | LABEL | MNEMONICS | OPCODE | COMMENTS |
|---|---|---|---|---|
| | | MOV AX, [1100H] | | |
| | | MOV BX, [1102H] | | |
| | | ADD AX, BX | | |
| | | MOV [1200H], AX | | |
| | | HLT | | |

**Observation**

| IN PUT ADDRESS | DATA |
|---|---|
| 1100H | |
| 1101H | |
| 1102H | |
| 1103H | |

| OUT PUT ADDRESS | DATA |
|---|---|
| 1200H | |
| 1201H | |

**Subtraction without borrow:**

| ADDRESS | LABE L | MNEMONICS | OPCOD E | COMMENTS |
|---|---|---|---|---|
| | | MOV AX, [1100H] | | |
| | | MOV BX, [1102H] | | |
| | | SUB AX, BX | | |
| | | MOV [1200H], AX | | |
| | | HLT | | |

**Observation**

| IN PUT ADDRESS | DATA |
|---|---|
| 1100H | |
| 1101H | |
| 1102H | |
| 1103H | |

| OUT PUT ADDRESS | DATA |
|---|---|
| 1200H | |
| 1201H | |

**Multiplication:**

| ADDRESS | LABE L | MNEMONICS | OPCOD E | COMMENTS |
|---|---|---|---|---|
| | | MOV AX, [1100H] | | |
| | | MOV BX, [1102H] | | |
| | | MUL BX | | |
| | | MOV [1200H], AX | | |
| | | MOV [1202H], DX | | |
| | | HLT | | |

**Observation**

| OUT PUT ADDRESS | DATA |
|---|---|
| **1200H** | |
| **1201H** | |
| **1202H** | |
| **1203H** | |

| IN PUT ADDRESS | DATA |
|---|---|
| **1100H** | |
| **1101H** | |
| **1102H** | |
| **1103H** | |

**Division:**

| ADDRESS | LABE L | MNEMONICS | OPCOD E | COMMENTS |
|---|---|---|---|---|
| | | MOV AX, [1100H] | | |
| | | MOV BX, [1102H] | | |
| | | DIV BX | | |
| | | MOV [1200H], AX | | |
| | | MOV [1202H], DX | | |
| | | HLT | | |

**Observation**

| OUT PUT ADDRESS | DATA |
|---|---|
| 1200H | |
| 1201H | |
| 1202H | |
| 1203H | |

| IN PUT ADDRESS | DATA |
|---|---|
| 1100H | |
| 1101H | |
| 1102H | |
| 1103H | |

## 2.5 Pre Lab Questions

1. Give the format of Flag register and state the conditions to set the flags.
2. Assume DS = 2000H, calculate the physical addresses of all datas (input and output).
3. List the addressing modes used in each program.

## 2.6 Post Lab Questions

1. Write an ALP to solve the following expressions:
   (i) 2a + 3b
   (ii) (a + b) / (a – b)
   (iii) (a * b) / (a + b) * (a – b)

2. Simulate the programs using emulator 8086.

**Result:**