Name – Faria Islam Hridy

ID – 1712990642

Section: 10

1. Why it is never a good idea to store any data in the segment registers?

Although it is possible to store any data in the segment registers, this is never

a good idea. Segment registers have been allocated from variable data storage to various tasks. It effectively saves the stack address. The processor combines the segment address in the segment register with the offset value of the location in order to reference any memory location in a segment. So, storing data inside the section register is not an ideal idea. The segment registers have a very special purpose - pointing at accessible blocks of memory. This will be discussed further in upcoming classes.

1. Explain GPR in terms of storing data.

The main purpose of a register is to keep a number (variable). The size of the above registers is 16 bits. General purpose registers are used to store temporary data within the microprocessor. There are 8 general purpose registers in 8086 MP.

4 general-purpose registers (AX, BX, CX, DX) are made of two separates 8-bit registers, for example if AX= 0011000000111001b, then AH=00110000b and AL=00111001b. Therefore, when you modify any of the 8-bit registers 16-bit registers are also updated, and vice-versa. The same is for other 3 registers, "H" is for high and "L" is for low part. Since registers are located inside the CPU, they are much faster than a memory.

**AX –** The accumulator is this. It has 16 bits and is split into two AH and AL 8-bit registers to execute 8-bit instructions as well. It is commonly used for arithmetical and logical instructions but it is not mandatory to have accumulator as the destination operand in 8086 MP.

Example: ADD AX, AX (AX = AX + AX)

**BX –** This is the register at the foundation. It is 16-bit and is split into two BH and BL 8-bit registers to execute 8-bit instructions as well. It is used to store the value of the offset.

Example: MOV BL, [500] (BL = 500H)

**CX –** This is the ledger for the counters. It has 16 bits and is split into two CH and CL 8-bit registers to execute 8-bit instructions as well. It is used for looping and rotation operations.

Example: MOV CX, 0005 LOOP

**DX –** This is the Register of Data. It has 16 bits and is split into two DH and DL 8-bit registers to execute 8-bit instructions as well. It is used for addressing an input/output port for multiplication.

Example: MUL BX (DX, AX = AX \* BX)