Sub : _____ Day | Time : Date : / /

Md. NWT-E-Azam
ID: 151226 8042

CSE-331 LAB

LAB-1

☐ Required to run an assembly program:

1. Write the necessary assembly source code
2. save the assembly source code
3. compile/Assemble source code to create machine code
4. Emulate / Run the machine code

☐ Features of 8086

⇒ 8086 is a 16bit processor. It's ALU, internal registers work with 16bit binary word

⇒ 8086 has a 16bit data bus. It can read or write data to a memory/port either 16 bits or 8 bits at a time

Sub :

Day

Time : Date : / /

⇒ 8086 has a 20bit address bus which means it can address up to 2^20 = 1MB memory location.

☐ Registrar – Register – Resistor

⇒ Both ALU & FPU have a very small amount of super-fast private memory placed right next to them for their exclusive use. These are called registers.

⇒ The ALU & FPU store intermediate and final results from their calculations in these registers

⇒ processed data goes back to the data cache and then to the main memory from these registers.

Sub :

Day

Time : Date : / /

General purpose Registers (GPR)

The 8086 cpu has 8 general-purpose registers; each register has its own name :

1. AX :- The Accumulator register (AH/AL).
2. BX :- The Base Address register (BH/BL).
3. CX :- The count register (CH/CL).
4. DX :- The Data register (DH/DL).
5. SI :- Source Index register.
6. DI :- Destination Index register.
7. BP :- Base pointer.
8. SP :- Stack pointer.

Segment Registers

CS :- points at the segment containing the current program.

DS :- generally points at the segment where variables are defined.

ES :- extra segment register, its up to a coder to define its usage.

SS :- points at the segment containing the st

Sub :

Day

Time : Date : / /

Special purpose Registers

1. IP :- The Instruction pointer. Points to the next location of instruction in the memory.

2. Flage Register :- Determines the current state of the microprocessor modified automatically by the cpu after some mathematical operations determines certain types of result and determines how to transfer control of a program.

First Assembly code

```
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
        MOV AH,1
        INT 21H
        MOV BL, AL
        MOV AH, 2
        MOV DL, BL
        INT 21H
        MOV AH, 4CH
        INT 21H
END MAIN MAIN ENDP
```