#### **Control Registers**

Overflow Flag (OF)
Direction Flag (DF)
Interrupt Flag (IF)
Trap Flag (TF)
Sign Flag (SF)
Zero Flag (ZF)
Auxiliary Carry Flag (AF)
Parity Flag (PF)
Carry Flag (CF)

#### **Segment Registers**

**Code Segment** all the instructions to be executed. CS register stores the starting address of the code segment

**Data Segment**: contains data, constants and work areas.DS register stores the starting address of the data segment.

**Stack Segment**: data and return addresses of procedures. It is implemented as a 'stack' data structure. The Stack Segment register or SS register stores the starting address of the stack.

**Extra Segment** is an extra segment register, that was not by default tied to any specific register

All right reserved by Bappy Nur youtube link https://www.youtube.com/watch?v=6hkKb4hOZmQ&list=PLiceGnDCE4XZdGymw66QH3o1tp2JbXLGA

# **Segment Registers**

Code Segment	CS	
Data Segment	DS	
Stack Segment	SS	
Extra Segment	ES	

All right reserved by Bappy Nur youtube link https://www.youtube.com/watch?v=6hkKb4hOZmQ&list=P LiceGnDCE4XZdGymw66QH3o1tp2JbXLGA

## **General Purpose Registers**

		15 0	
Accumulator	AX		Multiply, divide, I/O
Base	$\mathbf{B}\mathbf{X}$		Pointer to base addresss (data)
Count	CX		Count for loops, shifts
Data	DX		Multiply, divide, I/O

### **Pointer and Index Registers**

		.5 0	
Stack Pointer	SP		Pointer to top of stack
Base Pointer	BP		Pointer to base address (stack)
Source Index	SI		Source string/index pointer
Destination Index	DI		Destination string/index pointer
		5 0	

## **Segment Registers**

Code Segment	CS	
Data Segment	DS	
Stack Segment	SS	
Extra Segment	ES	

All right reserved by Bappy Nur youtube link https://www.youtube.com/watch?v=6hkKb4hOZmQ&list=P LiceGnDCE4XZdGymw66QH3o1tp2JbXLGA