PRACTICAL-9

AIM: Write a program to convert a number from BCD to Binary.

PROGRAM:

```
LDA 2010H
MOV B,A
ANI 0FH
MOV C,A
MOV A,B
ANI F0H
JZ SKIPMUL
RRC
RRC
RRC
RRC
MOV D,A
XRA A
MVI E,0AH
SUM: ADD D
DCR E
JNZ SUM
SKIPMUL: ADD C
STA 2020H
HLT
```

OUTPUT:

Memory View   ↻   0x   2020

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
201	92	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
202	5C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
203	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
204	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
205	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
206	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
207	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
208	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
209	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Start Address at: 0x [1100 ▾]

CONCLUSION: The above code is successfully executed in Lab.

PRACTICAL-10

AIM: Write a program to convert from binary to ASCII.

PROGRAM:

LXI H,8000H
MOV A,M
MOV B,A
STC
CMC
SUI 0AH
JC NUM
ADI 41H
JMP STORE
NUM: MOV A,B
ADI 30H
STORE:INX H
MOV M,A
HLT

Memory View

0x 8000

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
800	0F	46	00	00	00	00	00	00	00	00	00	00	00	00	00	00
801	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
802	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
803	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
804	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
805	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
806	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
807	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
808	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
809	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Start Address at: 0x 7700

**CONCLUSION: The above code is successfully executed in Lab.**

PRACTICAL-11

**AIM: Introduction to 8086 Microprocessor.**

Features of 8086
• 8086 is a 16bit processor. It's ALU, internal registers works with 16bit binary word.
• 8086 has a 16bit data bus. It can read or write data to a memory/port either 16bits or 8 bit at a time
• 8086 has a 20bit address bus which means, it can address upto 1MBmemory location
• Frequency range of 8086 is 6-10 MH.
Pin Diagram of 8086Microprocessor.
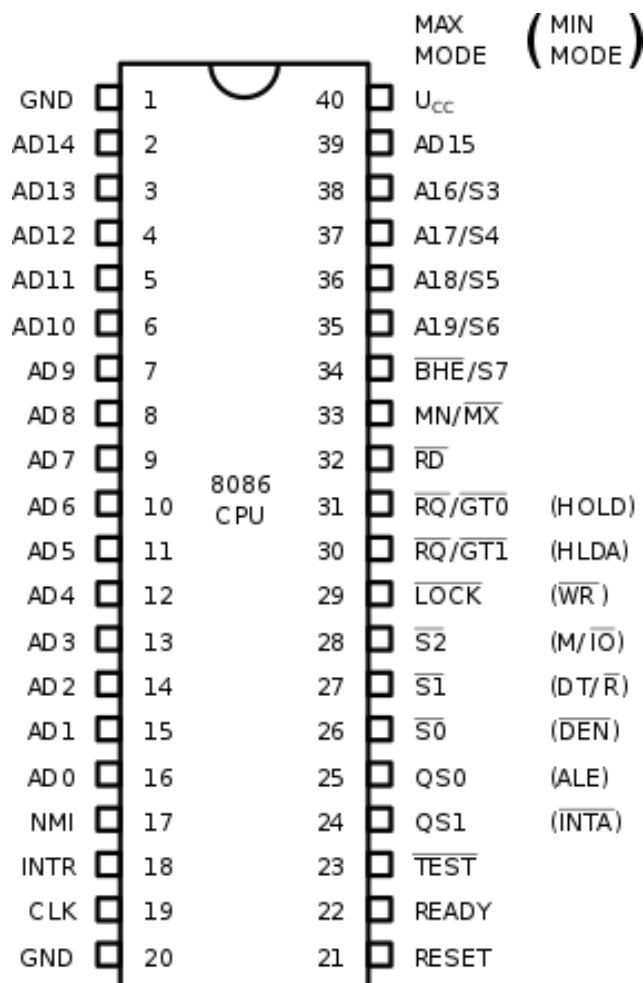
Architecture of 8086 Microprocessor

1=main & index registers; 2=segment registers and IP; 3=address adder;
4=internal
address bus; 5=instruction queue; 6=control unit (very simplified!); 7=bus
interface;
8=internal databus; 9=ALU; 10/11/12=external address/data/control bus.
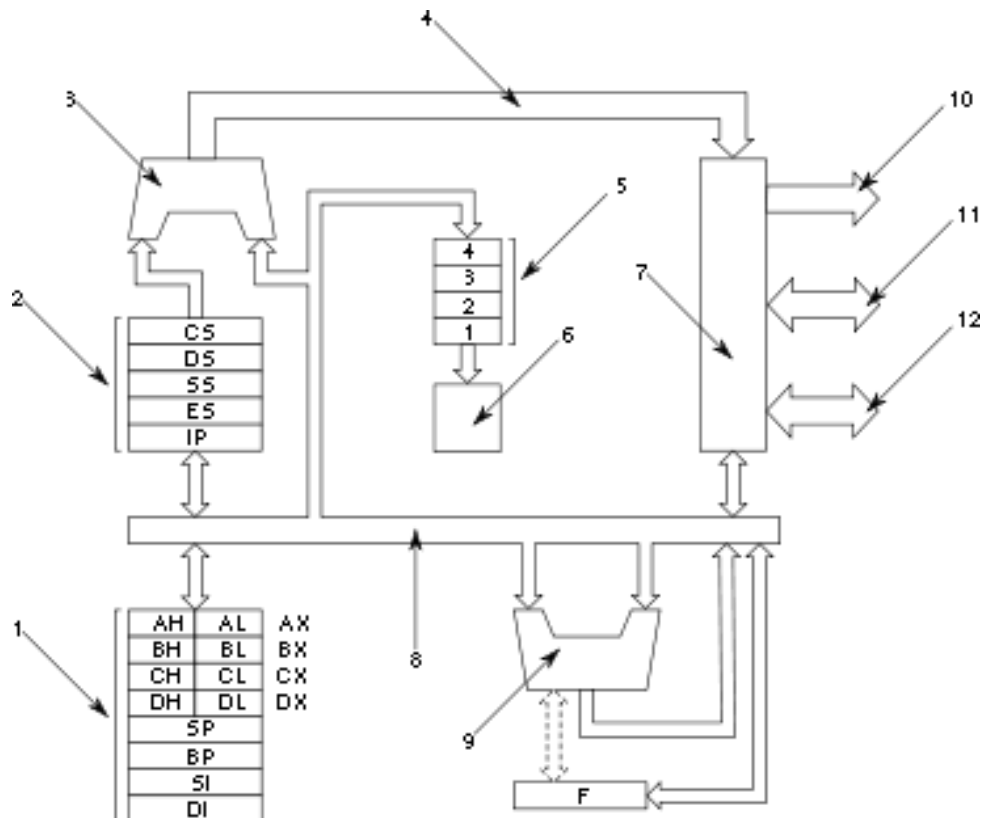
Registers
The 8086 has eight more or less general 16-bit registers (including the
stack pointer but excluding the instruction pointer, flag register and segment
registers). Four of them, AX, BX, CX, DX, can also be accessed as twice as
many 8- bit registers while the other four, BP, SI, DI, SP, are 16-bit only.

A 64 KB (one segment) stack growing towards lower addresses is
supported in hardware; 16-bit words are pushed onto the stack, and the top of
the stack is pointed to by SS:SP. There are 256 interrupts, which can be invoked
by both hardware and software. The interrupts can cascade, using the stack to
store the return addresses.

The 8086 has 64 K of 8-bit (or alternatively 32 K of 16-bit word) I/O port space.

Flags

8086 has a 16-bit flags register. Nine of these condition code flags are active, and indicate the current state of the processor: Carry flag (CF), Parity flag (PF), Auxiliary carry flag(AF), Zero flag (ZF), Sign flag (SF), Trap flag (TF), Interrupt flag (IF), Direction flag (DF), and Overflow flag (OF).

Segmentation

There are also four 16-bit segment registers that allow the 8086 CPU to access one megabyte of memory in an unusual way. Rather than concatenating the segment register with the address register, as in most processors whose address space exceeded their register size, the 8086 shifts the 16-bit segment only four bits left before adding it to the 16-bit offset (16×segment + offset), therefore producing a 20-bit external (or effective or physical) address from the 32-bit segment: offset pair. As a result, each external address can be referred to by 212 = 4096 different segment: offset pairs.

Types of Instruction Set

• Data Transfer Instructions
• Arithmetic Instructions
• Logical Instructions
• Shift and Rotate Instructions
• Transfer Instructions
• Subroutine and Interrupt Instructions
• String Instructions
• Processor Control Instructions

Conclusion: We have studied the basic structure of the 8086 micrprocessor.