

Microprocessor 8086 Features

How many bits is the 8086 Microprocessor?

16 bit

What and how many 16 bit general purpose registers are there?

4, 16 bit general purpose registers. They are: AX, BX, CX, DX

What and how many 8 bit general purpose registers are there?

8, 8 bit general purpose registers. They are: AL, AH, BL, BH, CL, CH, DL, DH

What is the maximum for an external memory of the 8086 microcontroller?

1 MB

How many bits is the data bus?

16 bits

How many bits is the address bus?

20 bits

How many bytes is the instruction queue?

6 byte

How many bits is the ALU (Arithmetic Logic Unit)

16 bits

Microprocessor 8086 Features (cont)

How many segmented memories are there and what's the maximum available space?

CS (Code Segment), DS (Data Segment), SS (Stack Segment), EX (Extra Segment), all of which are 64 KB each in space

What are the corresponding points for the segments?

IP (Instruction Pointer), SP (Stack Pointer), BP (Base Pointer), SI (Source Index), DI (Destination Index)

What is the clock frequency of the microcontroller?

5MHz frequency

How many pins does the microcontroller have?

40 pins

A microprocessor is the brain of the computer, it is:

1. Multipurpose
2. Clock Driven
3. Register Based
4. Programmable
5. Digital Integrated Circuit

Notes:

IP (Instruction Pointer) is responsible for storing the instruction of the next executable instruction

Addressing Modes

Direct addressing mode

The addressing mode in which the effective address of the memory location is written directly in the instruction.

Register addressing

It means that the register is the source of an operand for an instruction.

Immediate addressing

The addressing mode in which the data operand is a part of the instruction itself is known as immediate addressing mode.

Register indirect addressing

This addressing mode allows data to be addressed at any memory location through an offset address held in any of the following registers: BP, BX, DI & SI.

Base addressing

In this addressing mode, the offset address of the operand is given by the sum of contents of the BX/BP registers and 8-bit/16-bit displacement.

Addressing Modes (cont)

Index addressing

In this addressing mode, the operands offset address is found by adding the contents of SI or DI register and 8-bit/16-bit displacements.

Base-index addressing

In this addressing mode, the offset address of the operand is computed by summing the base register to the contents of an Index register.

Relative addressing

In this the effective address is calculated with reference to instruction pointer.

Addressing applied in string processing instructions

This addressing mode is related to string instructions. In this the value of SI and DI are auto incremented and decremented depending upon the value of directional flag.



By **Deathtitan77**
(Deathtitan77)

Published 10th May, 2020.
Last updated 13th May, 2020.
Page 1 of 3.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Addressing Modes (cont)

Stack addressing

The stack memory addressing mode is used whenever you perform a push or pop operation. Always a word will be entered or popped from the stack in this addressing mode, and the value of the Stack Pointer (SP) will be incremented or decremented accordingly.

Examples below

1. `MOV AX, [1592H]`
2. `MOV CX, AX`
`ADD BX, AX`
3. `ADD AX, 2387H`
4. `MOV AX, [BX]`
5. `MOV DX, [BX+04]`
6. `MOV BX, [SI+16]`
7. `ADD CX, [AX+SI]`

General Purpose Registers

AX

This is the accumulator. It is of 16 bits and is divided into two 8-bit registers AH and AL to also perform 8-bit instructions. It is generally used for arithmetical and logical instructions but in 8086 microprocessor it is not mandatory to have accumulator as the destination operand.

General Purpose Registers (cont)

BX

This is the base register. It is of 16 bits and is divided into two 8-bit registers BH and BL to also perform 8-bit instructions. It is used to store the value of the offset.

CX

This is the counter register. It is of 16 bits and is divided into two 8-bit registers CH and CL to also perform 8-bit instructions. It is used in looping and rotation.

DX

This is the data register. It is of 16 bits and is divided into two 8-bit registers DH and DL to also perform 8-bit instructions. It is used in multiplication an input/output port addressing.

Every digit of hex corresponds to 4 bits

Examples:

```
MOV AX, 05H
MOV [500], AX

MOV BL, [500]
```

Addressing Modes

Immediate Addressing Mode

In this immediate data is the part of the instruction itself.
Example: `Mov AX, 0005H`

Addressing Modes (cont)

Direct Addressing Mode

In it, a 16-bit memory address (offset) or an input/output address is directly specified in the instruction as a part of it.

Register Addressing Mode

Here data is stored in a register and referred using the particular register.

Register Indirect Addressing Mode

In this offset address of data is in either Bx, SI, DI, (Base register, source index or Destination index) default segment is either DS or ES. Data is supposed to be available at the address pointed to by the content of any of the above registers in the default data segment.

Indexed Addressing Mode

Here offset of the operand is stored in one of the index registers. DS is the default segment for SI and DI in string instruction DS and ES default segment for register SI and DI.

Addressing Modes (cont)

Register Relative Addressing Mode

In it, data is available at an effective address formed by adding an 8 bit or 16-bit displacement with content, any one of the registers Bx, Bp, SI, DI in the default (DS or ES) segment.

Based Indexed Addressing Mode

The effective address of data is formed by adding content of base register Bx or Bp to the content of index register.

Relative Based Indexed Addressing Mode

Here the effective address is formed by adding an 8 bit or 16-bit displacement with the sum of the content of any one of the index registers in the default segment.

Index Registers

SI

This is the source index register. It is of 16 bits. It is used in the pointer addressing of data and as a source in some string related operations. It's offset is relative to data segment.



Index Registers (cont)

DI

This is the destination index register. It is of 16 bits. It is used in the pointer addressing of data and as a destination in some string related operations. It's offset is relative to extra segment.

Segment Registers

CS

Points to the segment containing the current program

DS

Points to the segment where the variables are defined

ES

It is an extra segment register, and it is up to the coder for the usage

SS

Points at the segment containing the stack

Note:

These registers each have an available space of 64 KB and 256 KB in total. The rest of the 1 MB total memory goes to the microcontroller. It can be assumed as an "Operating System".

Pointer Registers

SP

This is the stack pointer. It is of 16 bits. It points to the topmost item of the stack. If the stack is empty the stack pointer will be (FFFE)H. It's offset address relative to stack segment.

BP

This is the base pointer. It is of 16 bits. It is primary used in accessing parameters passed by the stack. It's offset address relative to stack segment.

IP

To access instruction the 8086 uses the register CS and IP. The CS register contains the segment number of the next instruction and IP contains the offset.

Control Flags

Directional Flag (D)

This flag is specifically used in string instructions. If directional flag is set (1), then access the string data from higher memory location towards lower memory location. If directional flag is reset (0), then access the string data from lower memory location towards higher memory location.

Control Flags (cont)

Interrupt Flag (I)

This flag is for interrupts. If interrupt flag is set (1), the microprocessor will recognize interrupt requests from the peripherals. If interrupt flag is reset (0), the microprocessor will not recognize any interrupt requests and will ignore them.

Trap Flag (T)

This flag is used for on-chip debugging. Setting trap flag puts the microprocessor into single step mode for debugging. In single stepping, the microprocessor executes a instruction and enters into single step ISR. If trap flag is set (1), the CPU automatically generates an internal interrupt after each instruction, allowing a program to be inspected as it executes instruction by instruction. If trap flag is reset (0), no function is performed.

Status Flags

Overflow Flag (O)

This flag will be set (1) if the result of a signed operation is too large to fit in the number of bits available to represent it, otherwise reset (0).

There are also 5 other flags, but they are the same as the 8085.



By **Deathtitan77**
(Deathtitan77)

Published 10th May, 2020.
Last updated 13th May, 2020.
Page 3 of 3.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>