Organization of IBM Computers

1

Computer Organization and Assembly Language

Organization of IBM Computers

- The microprocessor (8086)

IBM Computers

- IBM Computers are based on Intel 8086 family of microprocessors.

Intel 8086 Family of Microprocessors

- Intel 8086 family of microprocessors includes the 8086, 8088, 80186, 80188, 80286, 80386, 80386SX, 80486, and 80486SX.
- The 8088 is used In the PC and PC XT; the 80286 is used in the PC AT and PS/1.
- The PS/2 models use either the 8086, 80286, 80386, or 80486:

IBM Computers

- IBM Computers are based on Intel 8086 family of microprocessors.

The 8086 Microprocessor

- Intel introduced the 8086 in 1978 as its first 16-bit microprocessor (a 16-bit processor can operate on 16 bits of data at a time).
- The 8086 has a 16-bit data bus.
- The 8086 and 8088 have same instruction set and it forms the basic set of instructions for the other microprocessors in the family.

The 80186 Microprocessor

- The 80186 is enhanced version of the 8086.
- It incorporates all the functions of the 8086 plus some support chips.
- It can also execute some new instructions called the extended instruction set.
- The processor offered no significant advantage over the 8086.

The 80286 Microprocessor Operation Modes

- *Two modes of operations.*
- The 80286 can operate in **real addressing mode** or **protected virtual address mode**.
- In real address mode, 80286 behaves like 8086, and the 8086 programs can execute in this mode without modification.
- In protected virtual address mode, or protected mode, the 80286 supports multitasking, and memory protection, which is the ability to protect the memory used by one program from the actions of another program.

The 80286 Microprocessor Addressable Memory

- *More addressable memory*. The 80286 in protected mode can address 16 megabytes of physical memory (as opposed to 1 megabyte for the 8086 and 8088).

The 80286 Microprocessor Virtual Memory

- *The 80286 can treat external storage (that is, a disk) as if it were physical memory, and therefore execute programs that are too large to be contained in physical memory; such programs can be. up to 1 gigabyte (2^{30} bytes).*

The *80386 Microprocessor*

- First 32-bit microprocessor, 80386 (or 386) was introduced in 1985.
- It is much faster than the 80286 because it has a 12-bit data path, high clock rate (up to 33MHz), and the ability to execute instructions in fewer clock cycles than 80286.

The *80386 Microprocessor*

- The 386 can operate in either real or protected mode.
- In protected mode, it can emulate the 80286.
- It also has a virtual 8086 mode to run multiple 8086 app1ications under memory protection.
- The 386, in protected mode, can address 4 gigabytes of memory.

The *80486 Microprocessor*

- Introduced in 1989, the 80486 (or 486), is another 32-bit microprocessor.
- It incorporates the functions of the 386 together with those of other support chips, including the 80387 numeric processor, for floating-point number operations,
- It 486 supports an 8-KB fast cache memory to buffer data coming from the slower memory unit.
- With its numeric processor, cache memory, and advanced design, the 486 is three times faster than a 386 at the same clock speed.

Organization of 8086 Microprocessor

- We will study the organization of the 8086 processor.
- 8086 has the simplest structure.

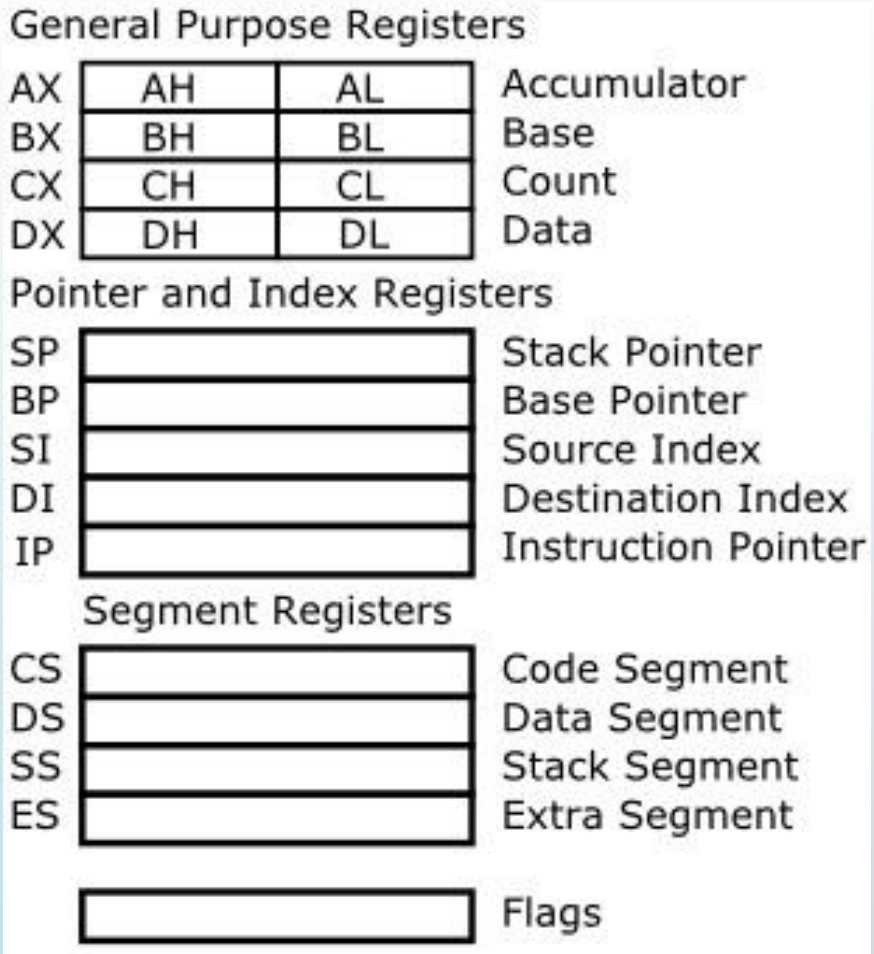
Registers

- Registers are data storage units inside microprocessor.
- They are classified according to their function.
- Data *registers* hold data for an operation,
- *address registers* hold the address of instruction or data,
- *flags or status register* keeps the current status of the processor.

Registers

- 8086 has four general data registers; the address registers are divided into segment, pointer and index registers; and the status register is called *FLAGS* register.
- In total there are fourteen registers of 16-bit each.

8086 Registers



Data Registers: AX, BX, CX, DX

- These are available to the programmer for general data manipulation.
- The processor can operate on data stored in memory, but same instruction is faster (requires fewer clock cycles) if the data is stored in registers.
  - This is why modern processors tend to have a lot of registers

Data Registers: AX, BX, CX, DX

- The high and low bytes of the data registers can be accessed separately.
- The high byte of AX is called AH. and the low byte is AL.
- Similarly, the high and low bytes of BX, CX, and DX are BH and BI, CH and CL, DH and DL.
- This arrangement gives us more registers to use when dealing with byte-size data.

Data Registers: AX, BX, CX, DX

- The high and low bytes of the data registers can be accessed separately.
- The high byte of AX is called AH. and the low byte is AL.
- Similarly, the high and low bytes of BX, CX, and DX are BH and BI, CH and CL, DH and DL.
- This arrangement gives us more registers to use when dealing with byte-size data.

AX (Accumulator Register)

- In addition to being general-purpose registers, these registers also perform special functions:
- AX is the preferred register to use in arithmetic, logic, and control transfer instructions.
- In multiplication and division operations, one of the numbers involved must be in AX or AL.
- Input and output operations also require the use of AL and AX.

BX (Base Register)

- BX also serves as an address register; an example is a table look-up Instruction called XLAT (translate).

CX (Count Register)

- The loops in are facilitated by CX, which serves as a loop counter.
- Another example of using CX as counter is REP (repeat), which controls a special class of instructions called *string operations*.
- CL is used as a count in instructions that shift and rotate bits.

DX (Data Register)

- DX Is used in multiplication and division.
- It is also used in I/0 operations.

Segment Registers: CS, DS, SS, ES

- Address registers store addresses of instructions and data in memory.
- These values are used by the processor to access memory locations.
- Each memory byte has an address, starting with 0.
- The 8086 processor assigns a 20-bit physical address to its memory locations.
- Thus it is possible to address 2^{20} or 1,048,576 bytes (one megabyte) of memory.

Segment Registers: CS, DS, SS, ES

- Addresses are usually expressed in HEX. Thus

- and so on. The highest address is FFFFFh.

Memory Segment

- The memory segments, is a direct consequence of using a 20-bit address in a 16-bit processor.
- The addresses are too big to fit in a 16 bit register or memory word.
- The 8086 gets around this problem by partitioning its memory into segments.

Memory Segment

- A memory segment is a block of 2^{16} (or 64 KB) consecutive memory bytes.
- Each segment is identified by a segment number, starting from 0.
- A segment number is 16 bits, so the highest segment number is FFFFh.

Memory Segment

- Within a segment, a memory location is specified by an offset, which is the number of bytes from the beginning of the segment.
- With a 64-KB segment, the offset can be given as a 16-bit number.
- The first byte in a segment has offset 0. The last offset in a segment is FFFFh.

Segment:Offset Address

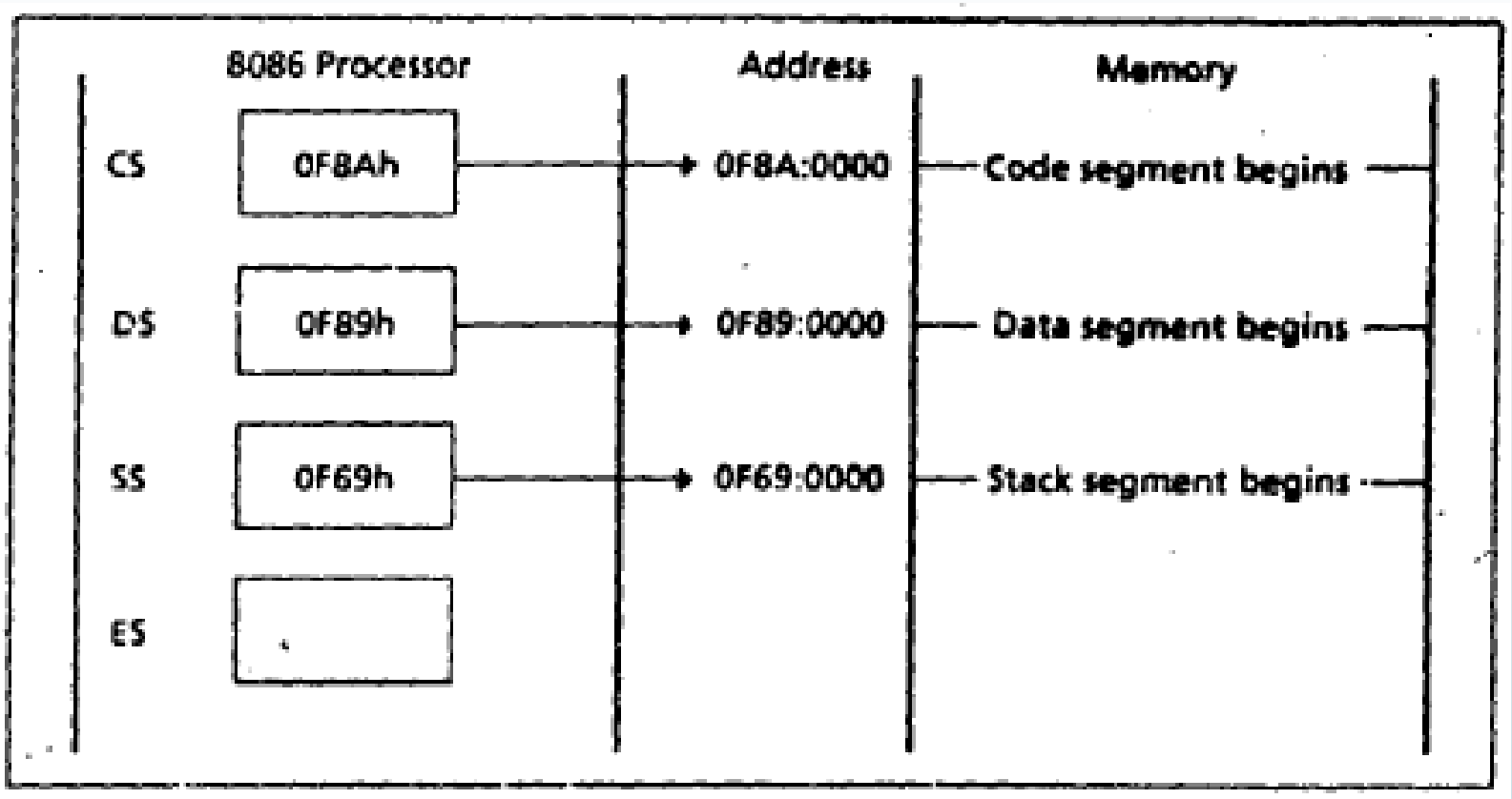
- A memory location is specified by providing a segment number and an offset, written in form segment:offset ; known as logical address.
- For example, A4FB:4872h means offset 4872h within segment A4FBh.
- To obtain a 20-bit physical address, the 8086 shifts the segment address 4 bits to the left (this is equivalent to multiplying by 10h), and then adds the offset.
- Thus the physical address for A4FB:4872 is

  A4FBOh

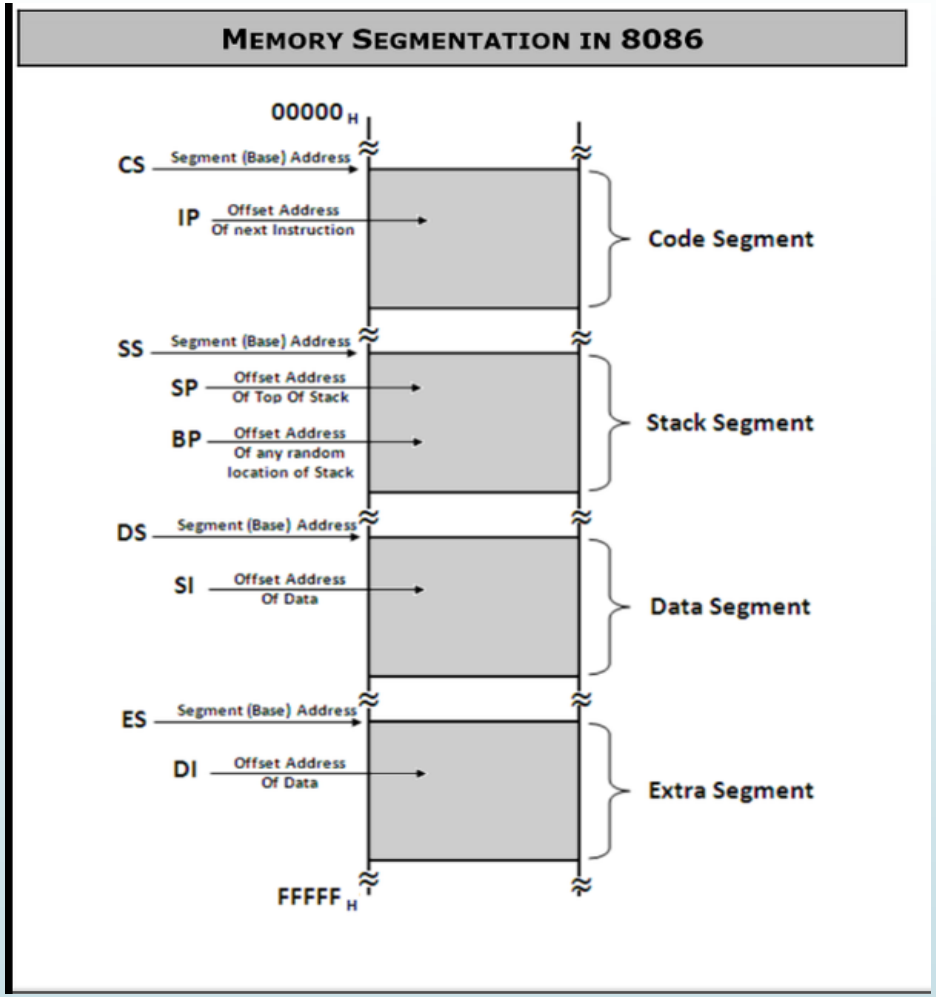
  + 4872h

  = A9822h (20-bit physical address)

Segment Registers



Segment Registers



Program Segments

- A typical machine language program consists of instructions (code) and data.
- The stack data structure is used by the processor to implement procedure calls.
- The program's code, data, and stack arc loaded into different memory segments, we call them the code segment. data segment, and stack segment.

Program Segments

- To keep track of the various program segments, 8086 is equipped with four segment registers.
- The CS, DS, and SS registers can contain the code, data, an stack.
- If a program needs to access a second data segment, it can use the ES (extra segment) register.

Pointer and Index Registers: SP, BP, SI, DI

- SP, BP, SI, and DI registers normally contain the offset addresses of memory locations.
- Unlike segment registers, the pointer and index registers can be used in arithmetic operations.

SP (Stack Pointer)

- SP (stack pointer) register is used in conjunction with SS for accessing the stack segment.

BP (Base Pointer)

- The BP (base pointer) register is used primarily to access data on the stack.
- Unlike SP, we can also use BP to access data in the other segments.

SI (Source index)

- The SI (source index) register is used to point to memory locations in the data segment addressed by OS.

DI (Destination Index)

- The DI (destination Index) register performs the same functions as SI.
- There is a class of instructions, called string operations, that use DI to access memory locations addressed by ES.

IP (*Instruction Pointer*)

- The registers covered so far are for data access.
- To access Instructions, 8086 uses the registers CS and IP.
- The CS register contains the segment number of the next instruction, and the IP contains the offset.
- IP is updated each time an instruction is executed so that it will point to the next Instruction.
- IP cannot be directly manipulated by an instruction.

FLAGS Register

- The FLAGS register indicates the status of the microprocessor by the setting of individual bits called flags.
- There are two kinds of flags: status flags and control flags.
- The status flags reflect the result of an executed instruction.
- For example, when a subtraction operation results in a 0, the ZF (zero flag) is set to 1 (true).
- A subsequent instruction can examine the ZF and branch to some code that handles a zero result.

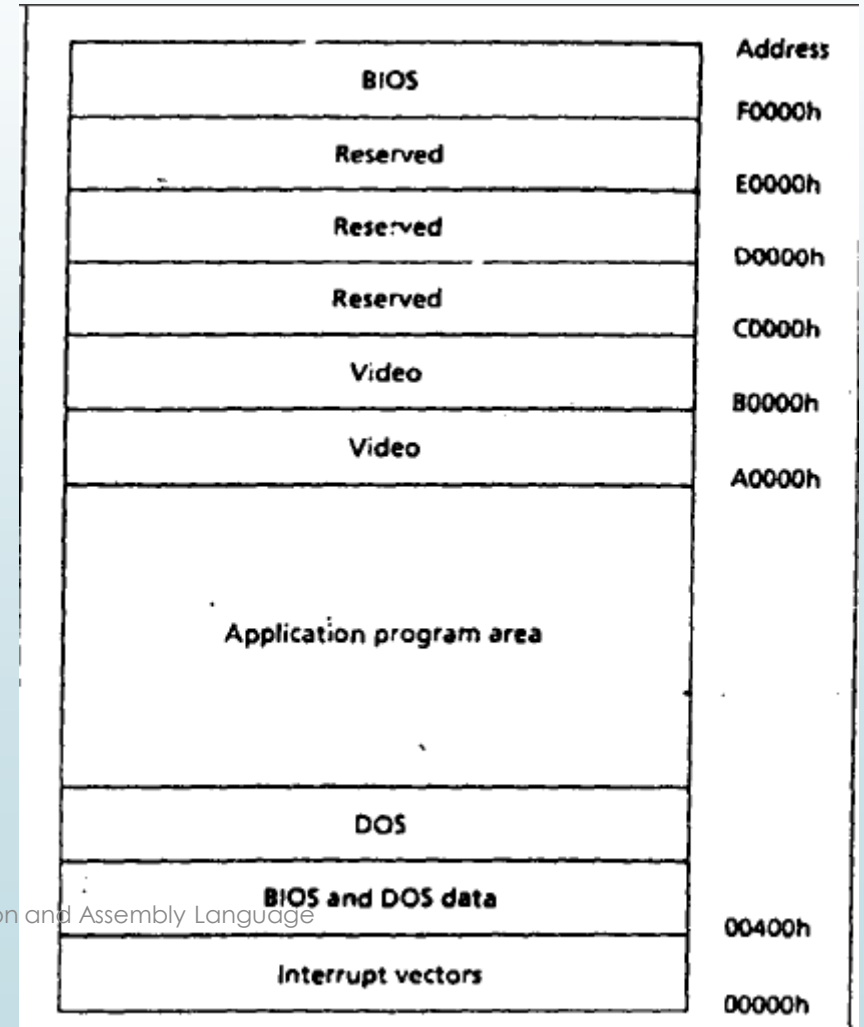
FLAGS Register

- The control flags enable or disable certain operations of the processor; for example, when the IF (interrupt flag) is cleared (set to 0), inputs from the keyboard are ignored by the processor.

Memory Organization of PC

- 8086/6088 processor is capable of addressing 1 megabyte of memory.
- However, not all the memory can be used by an application program.
- Some memory locations have special meaning for the processor.
- For example, the first kilobyte (00000 to 003FFh) is used for interrupt vectors.
- Other memory locations are reserved by IBM for special purposes, such as for BIOS routines and video display memory.
- The display memory holds the data that are being displayed on the monitor.

Memory Map of the PC



SP (Stack Pointer)

- SP (stack pointer) register is used in conjunction with SS for accessing the stack segment.