INTRODUCTION

A microprocessor is a computer processor which incorporates the functions of a computer's Central Processing Unit (CPU) on a single Integrated Circuit (IC), or at most a few integrated circuits. The microprocessor is a multipurpose, clock driven, register based, programmable electronic device which accepts digital or binary data as input, processes it according to instructions stored in its memory, and provides results as output. Microprocessors contain both combinational logic and sequential digital logic.

- Microprocessors operate on numbers and symbols represented in the binary numeral system Microprocessor is the controlling unit or CPU of a micro-computer, fabricated on a very small chip capable or performing ALU operations and communicating with the external word connected to it. It forms a micro-computer when combined with memory and Input/output devices.
- Microprocessors of different word size with varying decrease of capabilities are available. Microprocessor comprises of all the functional components of the central processing unit of a general purpose computer. In other words, functionally it is equivalent to a CPU.
- ***Cost:*** The most important characteristics of a microcomputer is its low cost. Because of the widespread use of microprocessors, the volume of production is very high. That is why, microprocessor chips are available at fairly low prices.
- ***Size:*** The second important features of a microprocessor is its small size. As a result of improvement in fabrication technology, VLSI, electronic circuitry has become so dense that a minute silicon chip can contain hundred and thousands of transistors constituting the microprocessor. Its size does not exceed a few inches on any side, even in the packaged form.
- ***Power Consumption:*** The important characteristics are its low power consumption microprocessors are normally manufactured by Metal-Oxide semiconductor technology.
- ***Versatility:*** The versatility of a microprocessor results from its stored program mode of operation. Keeping the same basic hardware, a microprocessor-based system can be

configured for a number of applications simplify altering the software program. This also makes it very flexible.

- ***Reliability:*** Another important property of VLSI devices which has also been in herniated by microprocessors is extreme reliability. It has been established that the failure rate of an IC is fairly uniform at the package level, regardless of its complexity.

THE 8086 MICROPROCESSOR

The 8086 is a 16-bit microprocessor chip designed by Intel between early 1976 and mid-1978, when it was released. The Intel 8088, released in 1979, was a slightly modified chip with an external 8-bit data bus (allowing the use of cheaper and fewer supporting ICs, and is notable as the processor used in the original IBM PC design, The 8086 gave rise to the x86 architecture which eventually turned out as Intel's most successful line of processors.

Features of 8086 Microprocessor:

- The 8086 is a 16-bit microprocessor which means its data handling capacity is 16-bits per clock. i.e. at any time any resources of 8086 systems can handle up to 16-bit of data for processing.
- It has 16-bit address bus.
- It has 20-bit data bus.
- Direct addressing capability 1MB of memory. (1MB=220 Bytes).
- It has fourteen 16-bit registers.
- 24 operand addressing modes.
- Supports Bit, Byte, Word and Block level operations.
- 8 and 16 bit signed and unsigned arithmetic operations including multiply and divide. (previously the multiply and divide operations were carried out using the iterative looping of addition and subtraction operations respectively).
- Four general purpose registers, each of 16 bit wide. AX, BX, CX and DX. These can be used as 8-bit as well as 16-bit registers.
  - ✓ AX (16-bit register) AL & AH (2 x 8 bit registers)
  - ✓ BX (16-bit register) BL & BH (2 x 8 bit registers)

- ✓ CX (16-bit register) CL & CH (2 x 8 bit registers)
- ✓ DX (16-bit register) DL & DH (2 x 8 bit registers)

- Two Index group registers available Source Index (SI) and Destination index (DI).
- There are four Segment registers in 8086: Code Segment (CS), Data Segment (DS), Stack Segment (SS), Extra Segment (ES).
- Six Status flags and three control flags.
- Memory is Byte addressable each stores and 8-bit value.
- Addresses can be upto 20-bits long, resulting up to 1MB of memory (216 Bytes=1MB)
- Ranges of Clock rates: 5MHz for 8086, 8 MHz for 8086-1 and 10 MHz for 8086-2.
- Multi-bus system compatible interface
- Available as a 40 Pins Plastic-DIP and Lead Cer-DIP.

The internal functions of the 8086 processor are partitioned logically as two functional units as shown in the figure 1.1. They are

- Bus Interface Unit (BIU)
- Execution Unit (EU)

The BIU and EU function independently. The BIU interfaces the 8086 to the outside world. It fetches the instructions, Reads data from memory and ports, and writes data to memory and I/O ports.

The EU receives the program instruction codes (OP-codes) and Data from the BIU, executes these instructions and stores the results in general registers/memory or send them out as outputs through ports using BIU. The EU has no connections to the system buses. It receives and outputs all its data through the BIU.

BIU contains

- Segment Registers
- Instruction Pointer
- Instruction Queue

EU contains

- Arithmetic and Logic Unit (ALU)

- General Purpose Registers
- Index Registers
- Pointers
- Flag registers

Bus Interface Unit (BIU)

The function of BIU is to:

- Fetch the instruction or data from memory.
- Write the data to memory.
- Write the data to the port.
- Read data from the port.

Instruction Queue

- The use of this queue is to hold next six instructions to be executed in FIFO manner.
- To increase the execution speed, BIU fetches as many as six instruction bytes ahead to time from memory.
- All six bytes are then held in first in first out 6-byte register called instruction queue.
- Then all bytes have to be given to EU one by one.
- This pre fetching operation of BIU may be in parallel with execution operation of EU, which improves the speed execution of the instruction.

Execution Unit (EU)

The functions of execution unit are:

- To tell BIU where to fetch the instructions or data from.
- To decode the instructions.
- To execute the instructions.
- The EU contains the control circuitry to perform various internal operations. A decoder in EU decodes the instruction fetched memory to generate different internal or external control signals required to perform the operation. EU has 16-bit ALU, which can perform arithmetic and logical operations on 8-bit as well as 16-bit.

General Purpose Registers of 8086:

These registers can be used as 8-bit registers individually as AL-AH, BL-BH, CL-CH and DL-DH or can be used as 16-bit in pair to have AX, BX, CX, and DX.

Note: Any Register RX can be 16-bit register, which can be used as 2 eight bit registers RL and RH, where RL contains lower order byte of that 16-bit word and RH contains the higher order byte of the word.

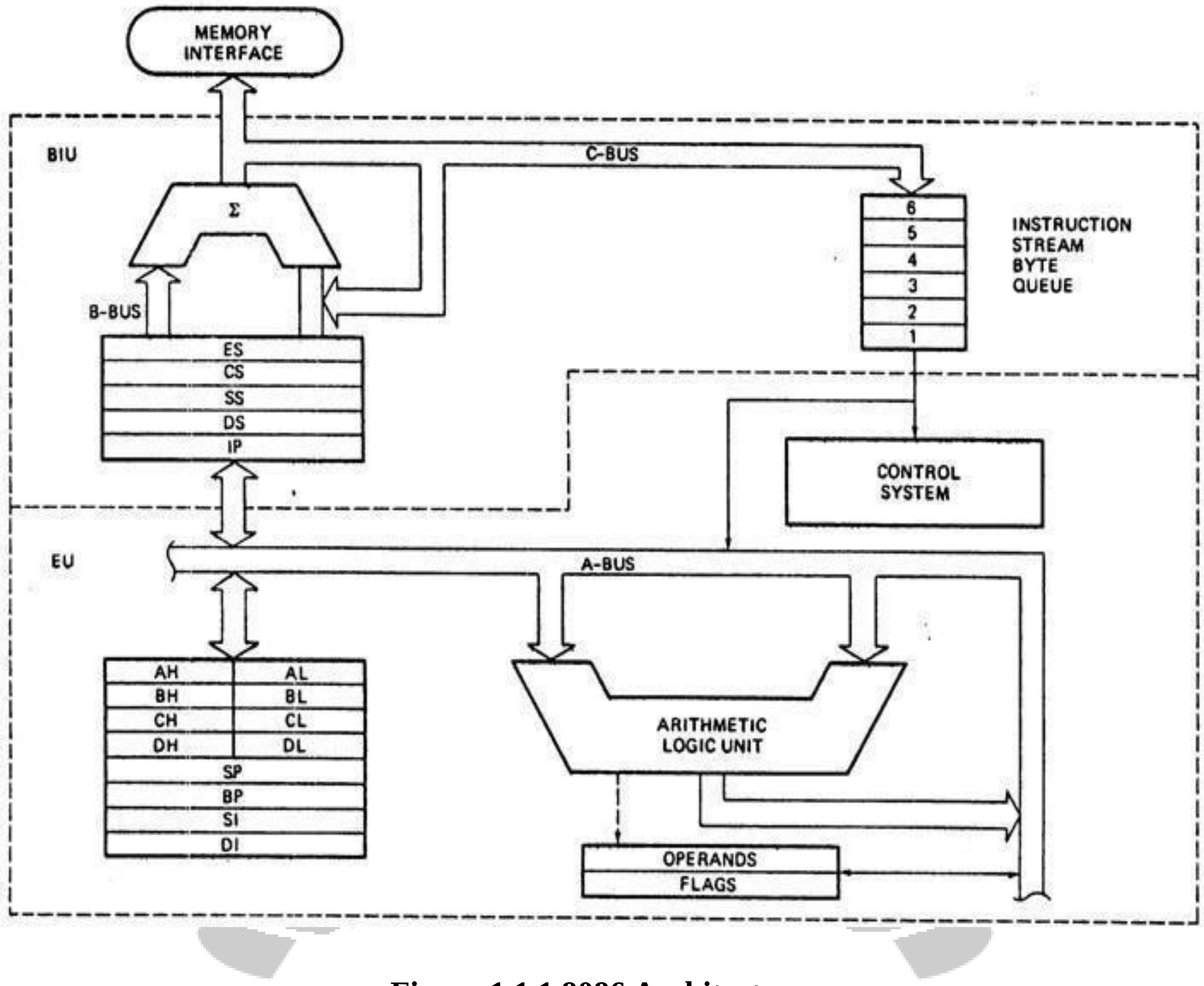

Figure 1.1.1 8086 Architecture

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

Physical Address Formation: Generation of 20-bit Address

The 8086 addresses a segmented memory. The complete physical address which is 20-bits long is generated using segment and offset registers each of the size 16-bit. The content of a segment register also called as segment address, and content of an offset register also called as offset address. To get total physical address, put the lower nibble0H to segment address and add offset address. The content of segment register is multiplied by 10H. i.e. shifted by 4 positions to the left by inserting 4 zero bits and then the offset. i.e. the contents of IP register are added to the shifted contents of CS to generate physical address.

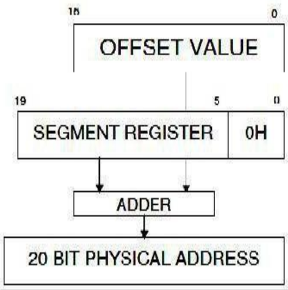

Figure 1.1.2 Physical Address Formation

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

The Figure 1.1.2 shows formation of 20-bit physical address.

Example: The contents of CS register are 348AH, therefore the shifted contents of CS register are 348A0H. When the BIU adds the offset of 4214H in IP to this starting address, we get 38AB4H as a 20-bit physical address of memory.

Register Organization of 8086:

All the registers of 8086 are 16-bit registers. The general purpose registers, can be used either 8- bit registers or 16-bit registers used for holding the data, variables and intermediate results temporarily or for other purpose like counter or for storing offset address for some particular addressing modes etc. The special purpose registers are used as segment registers, pointers, index registers or as offset storage registers for particular Addressing modes. Register organization of 8086 is shown in Figure 1.1.3.

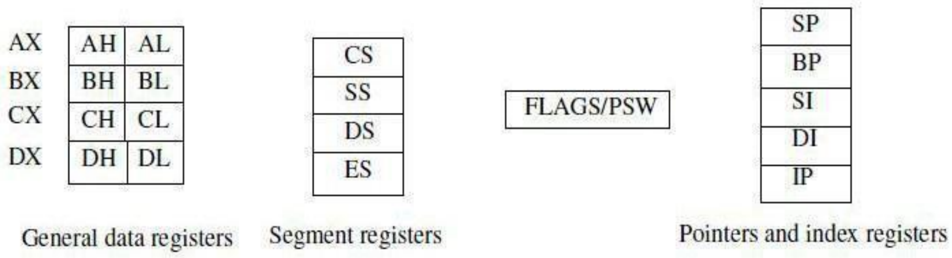

Figure 1.1.3 Register Organization of 8086

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

AX Register: Accumulator register consists of two 8-bit registers AL and AH, which can be combined together and used as a 16- bit register AX. AL in this case contains the low-order byte of the word, and AH contains the high-order byte. Accumulator can be used for I/O operations, rotate and string manipulation.

BX Register: This register is mainly used as a **base register**. It holds the starting base location of a memory region within a data segment. It is used as offset storage for forming physical address in case of certain addressing mode.

CX Register: It is used as default counter - **count register** in case of string and loop instructions.

DX Register: Data register can be used as a port number in I/O operations and implicit operand or destination in case of few instructions. In integer 32-bit multiply and divide instruction the DXregister contains high-order word of the initial or resulting number.

Segment Registers:

1Mbyte memory is divided into 16 logical segments. The complete 1Mbyte memory segmentation is as shown in fig 1.4. Each segment contains 64Kbyte of memory. There are foursegment registers.

Code Segment (CS) is a 16-bit register containing address of 64 KB segment with processor instructions. The processor uses CS segment for all accesses to instructions referenced by Instruction pointer (IP) register. CS register cannot be changed directly. The CS register is automatically updated during far jump, far call and far return instructions. It is used for addressing a memory location in the code segment of the memory, where the executableprogram is stored.

Stack Segment (SS) is a 16-bit register containing address of 64KB segment with program stack. By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment. SS register can be changed directly using POP instruction. It is used for addressing stack segment of memory. The stack segment is that segment of memory, which is used to store stack data.

Data Segment (DS) is a 16-bit register containing address of 64KB segment with program data. By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment. DS register can be changed directly using POP and LDS instructions. It points to the data segment memory where the data is resided.

Extra Segment (ES) is a 16-bit register containing address of 64KB segment, usually with program data. By default, the processor assumes that the DI register references the ES segment in string manipulation instructions. ES register can be changed directly using POP and LES instructions. It also refers to segment which essentially is another data segment of the memory. It also contains data.

Pointers and Index Registers:

The pointers contain within the particular segments. The pointers IP, BP, SP usually containoffsets within the code, data and stack segments respectively:

Stack Pointer (SP) is a 16-bit register pointing to program stack in stack segment.

Base Pointer (BP) is a 16-bit register pointing to data in stack segment. BP register is usuallyused for based, based indexed or register indirect addressing.

Source Index (SI) is a 16-bit register. SI is used for indexed, based indexed and register indirectaddressing, as well as a source data addresses in string manipulation instructions.

Destination Index (DI) is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data address in string manipulation instructions.

Flag Register:

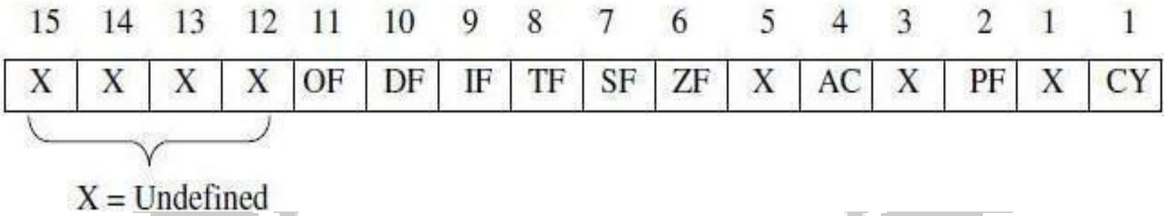

Figure 1.1.4 Flag Register of 8086

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

Flags Register determines the current state of the processor. They are modified automatically by CPU after mathematical operations, this allows to determine the type of the result, and to determine conditions to transfer control to other parts of the program. The 8086 flag register as shown in the figure 1.1.4.

8086 has 9 active flags and they are divided into two categories:

- Conditional Flags
- Control Flags

Conditional Flags

Carry Flag (CY): This flag indicates an overflow condition for unsigned integer **a**rithmetic. It isalso used in multiple-precision arithmetic.

Auxiliary Flag (AC): If an operation performed in ALU generates a carry/barrow from lower nibble (i.e. D0 – D3) to upper nibble (i.e. D4 – D7), the AC flag is set i.e. carry given by D3 bit to D4 is AC flag. This is not a general-purpose flag, it is used internally by the Processor to perform Binary to BCD conversion.

Parity Flag (PF): This flag is used to indicate the parity of result. If lower order 8-bits of the result contains even number of 1's, the Parity Flag is set and for odd number of 1's, the Parityflag is reset.

Zero Flag (ZF): It is set; if the result of arithmetic or logical operation is zero else it is reset.

Sign Flag (SF): In sign magnitude format the sign of number is indicated by MSB bit. If the result of operation is negative, sign flag is set.

Control Flags

Control flags are set or reset deliberately to control the operations of the execution unit. Control flags are as follows:

Trap Flag (TF): It is used for single step control. It allows user to execute one instruction of a program at a time for debugging. When trap flag is set, program can be run in single step mode.

Interrupt Flag (IF): It is an interrupt enable/disable flag. If it is set, the maskable interrupt of 8086 is enabled and if it is reset, the interrupt is disabled. It can be set by executing instruction sit and can be cleared by executing CLI instruction.

Direction Flag (DF): It is used in string operation. If it is set, string bytes are accessed from higher memory address to lower memory address. When it is reset, the string bytes are accessed from lower memory address to higher memory address.