**Laboratory Session-1**

**Write-up on Microprocessors, 8086 Functional block diagram, Pin diagram and description.**

**Microprocessors**

* The microprocessor, also known as the Central Processing Unit (CPU), is the brain of all computers and many household and electronic devices.
* The world’s first microprocessor, the Intel 4004, was a 4-bit microprocessor—a programmable controller on a chip.
* The main problems with this early microprocessor were its speed, word width, and memory size. To overcome this updated version of 4004 was evolved named 4040.
* Intel released 8008—an extended 8-bit version of the 4004. The 8008 addressed an expanded memory size (16K bytes) and contained additional instructions.
* In 1973, Intel released 8080, an enhanced version of 8008. Motorola Corp. introduced its MC6800 microprocessor.
* In 1977, Intel Corp. introduced an updated version of the 8080—the 8085. The 8085 was to be the last 8-bit, general-purpose microprocessor developed by Intel.
* 8085 executed software at an even higher speed. An addition that took 2.0µs (500,000 IPs on 8080) required only 1.3µs (769,230 IPs) on 8085.
* The main advantage of 8085 was its internal clock generator, internal system controller, and higher clock frequency.
* In 1978, Intel released the 8086 microprocessor, and an year later it released 8088. Both devices were 16-bit microprocessors, which executed instructions in 400ns (2.5MIPs).
* In addition, the 8086 and 8088 addressed 1M byte of memory, which was 16 times more memory than the 8085.
* The 80286 microprocessor (also a 16-bit microprocessor) was almost identical to the 8086/8088, except it addressed a 16M-byte memory system instead of a 1M-byte system.
* Pentium was released in 1993, was similar to 80386 and 80486 µp and this was originally labeled as P5 or 80586.
* The two introductory versions of the Pentium operated with a clocking frequency of 60MHz and 66MHz.
* Pentium Pro processors were formerly called P6 µp, that contained 21 million transistors, integer units, and a floating-point unit.
* Addition to the internal 16K level-one (L1) cache (8K for data and 8K for instructions) the Pentium Pro also contains a 256K level-two (L2) cache.

The major significant change is that the Pentium Pro uses three execution engines, so it can execute up to three instructions at a time

* Pentium II µp was released in 1997 represents a new direction for Intel.
* Intel placed the Pentium II on a small circuit board. The main reason for the change is that the L2 cache found on the main circuit board of the Pentium was not fast enough to function properly with the Pentium II.
* The P-III µp uses a faster core than the P-II, but it is still a P6 or Pentium Pro processor.
* It is also available in slot 1 version mounted on a plastic cartridge and a socket 370 version called a flip-chip.
* The P4 µp was first available in 2000. The most recent version of P4 is called Core2 by Intel.
* The P4 and Core2, like the Pentium Pro through the P-III use the Intel P6 architecture.

**8086 Functional block diagram**



The 8086 make use of this idle time by pre-fetching the next instruction while the current instruction is being executed. Here the bus is always busy.

The 8088 and 8086 contains two internal units;

1. The ***execution unit (EU)***

2. The ***bus interface unit (BIU).***

**The Execution Unit (EU)**

* The purpose of the EU is to carry out instructions that are fetched from the prefetch queue.
* It contains; an arithmetic and logic unit, an instruction register, and a register array.
* The ALU performs arithmetic and logic operations on memory and or register data.
* The instruction register receives the instructions from the prefetch queue. From here these instructions are decoded to direct the operation of the execution unit.
* The register array holds information temporarily. It also contains pointers or index registers used to address operand data located in the memory

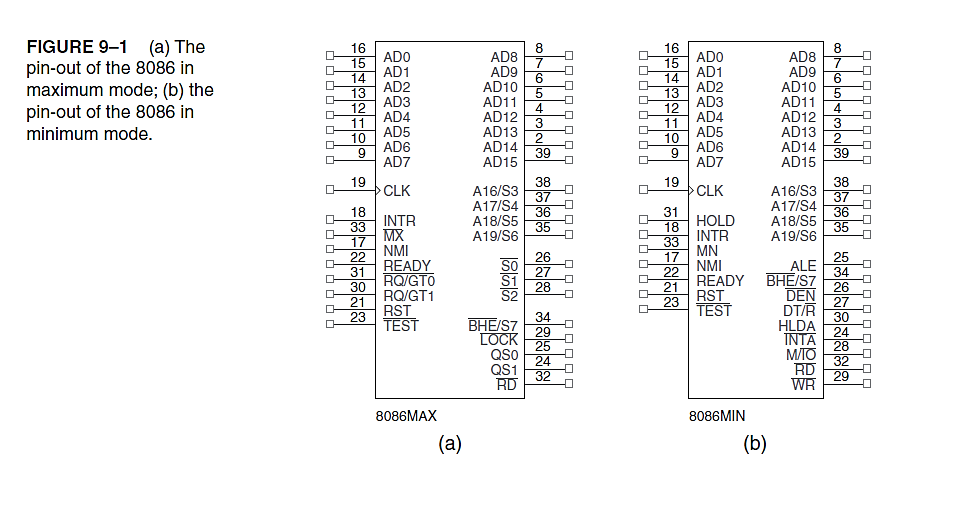
**The Bus Interface Unit (BIU)**

The BIU contains a prefetch queue, a bus controller, segment registers and the instruction pointer.

The main purposes of the BIU are:

* To keep the prefetch queue filled with instructions.
* To generate and accept the signal control signals.
* .To provide the system with a memory address or I/O port number.
* To act as a window between the EU and memory for data.

**8086 Pin diagram and description**

****

**Pin Connections**

**AD7–AD0**

* The 8088 **address/data bus** lines are the multiplexed address data bus of the 8088 contain the rightmost 8 bits of the memory address or I/O port number
* whenever ALE is active (logic 1) or data whenever ALE is inactive (logic 0).
* These pins are at their high-impedance state during a hold acknowledge

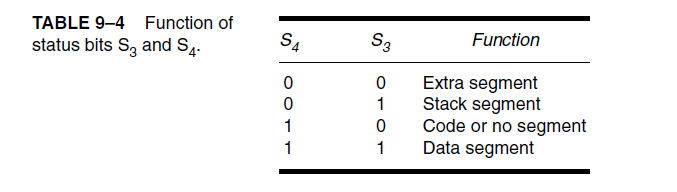
**A15–A8 : address bus:** provides the upper-half memory address bits that are present throughout a bus cycle.

**AD15–AD8** The 8086 **address/data bus**

* These lines compose the upper multiplexed address/data bus on the 8086.
* These lines contain address bits A15–A8 whenever ALE is a logic 1, and data bus connections D15–D8 when ALE is a logic 0.

**A19/S6–A16/S3** The **address/status bus**

* Thesebits are multiplexed to provide address signals A19–A16 and also status bits S6–S3.
* Status bit S6 is always a logic 0, bit S5 indicates the condition of the IF flag bit, and S4 and S3 show which segment is accessed during the current bus cycle.



**RD**

Whenever the **read signal** is a logic 0, the data bus is receptive to data from the memory or I/O devices connected to the system.

**READY**

* The **READY** input is controlled to insert wait states into the timing of the microprocessor.
* If the READY pin is placed at a logic 0 level, the microprocessor enters into wait states and remains idle.
* If the READY pin is placed at a logic 1 level, it has no effect on the operation of the microprocessor.

**INTR: Interrupt request**

It is used to request a hardware interrupt. If INTR is held high when IF = 1, the 8086/8088 enters an interrupt acknowledge cycle (INTA becomes active) after the current instruction has completed execution.

**TEST**

* The **Test** pin is an input that is tested by the WAIT instruction.
* If is a TEST logic 0, the WAIT instruction functions as an NOP and if TEST is a logic 1, the WAIT instruction waits for TEST to become a logic 0.
* The TEST pin is most often connected to the 8087 numeric coprocessor.

**NMI :** The **non-maskable interrupt**

* It is an input , similar to INTR except that the NMI interrupt does not check to see whether the IF flag bit is a logic 1.
* If NMI is activated, this interrupt input uses interrupt vector 2.

**RESET**

* The **reset** input causes the microprocessor to reset itself if this pin is held high for a minimum of four clocking periods.
* Whenever the 8086 or 8088 is reset, it begins executing instructions at memory location FFFFOH and disables future interrupts by clearing the IF flag bit.

**CLK**

The **clock** pin provides the basic timing signal to the microprocessor. The clock signal must have a duty cycle of 33 % (high for one third of the clocking period and low for two thirds) to provide proper internal timing for the 8086/8088.

**VCC**

This **power supply** input provides a +5.0 V, ±10 % signal to the microprocessor.

**GND**

* The **ground** connection is the return for the power supply. Note that the
* 8086/8088 microprocessors have two pins labeled GND—both must be connected to ground for proper operation.

**MN/MX**

* The **minimum/maximum** mode pin selects either minimum mode or maximum mode operation for the microprocessor.
* If minimum mode is selected, the MN/MX pin must be connected directly to +5.0 V.

**BHE / S7**

* The **bus high enable** pin is used in the 8086 to enable the most-significant data bus bits (D15–D8) during a read or a write operation.
* The state of S7 is always a logic 1.

**Minimum Mode Pins**

Minimum mode operation of the 8086/8088 is obtained by connecting the MN/MX pin directly to +5.0 V.

**IO/ M or M/IO**

* The **IO**/**M**  (8088) or the **M/ IO** (8086) pin selects memory or I/O.
* This pin indicates that the microprocessor address bus contains either a memory address or an I/O port address.

**WR**

* The **write line** is a strobe that indicates that the 8086/8088 is outputting data to a memory or I/O device.
* During the time that the WR is a logic 0, the data bus contains valid data for memory or I/O.

**INTA**

* The **interrupt acknowledge** signal is a response to the INTR input pin.
* The INTA pin is normally used to gate the interrupt vector number onto the data bus in response to an interrupt request.

**ALE**

* **Address latch enable** shows that the 8086/8088 address/data bus contains address information.
* This address can be a memory address or an I/O port number.
* Note that the ALE signal does not float during a hold acknowledge

**DT/ R**

* The **data transmit/receive** signal shows that the microprocessor data bus is transmitting (**DT/R =1**) or receiving (**DT/R=0** ) data.
* This signal is used to enable external data bus buffers.

**DEN : Data bus enable** activates external data bus buffers.

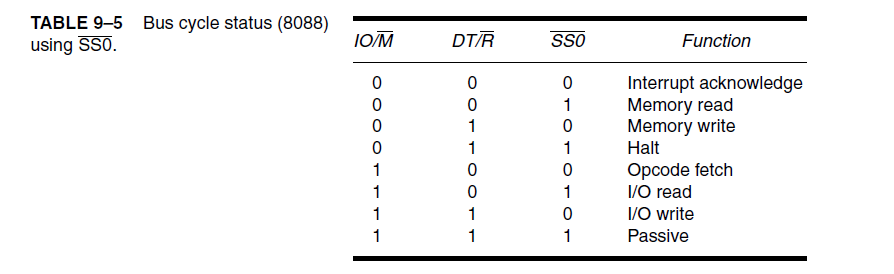
**HOLD:** The **hold input** requests a direct memory access (DMA).

* If the HOLD signal is a logic 1, the microprocessor stops executing software and places its address, data, and control bus at the high-impedance state.
* If the HOLD pin is a logic 0, the microprocessor executes software normally.

**HLDA: Hold acknowledge** indicates that the 8086/8088 has entered the hold state

S**S0** :

* The status line is equivalent to the S0 pin in maximum mode operation of the microprocessor.
* This signal is combined with **IO/M** and **DT/R** to decode the function of the current bus cycle

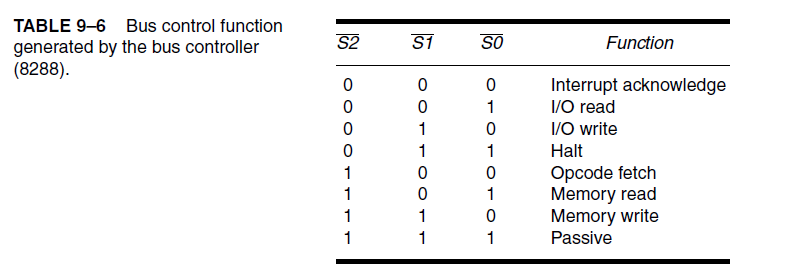


**Maximum Mode Pins**

In order to achieve maximum mode for use with external coprocessors, connect the MN/MX pin to ground.

**S2,S1 and S0**

* The **status bits** indicate the function of the current bus cycle.
* These signals are normally decoded by the 8288 bus controller.
* Table 9–6 shows the function of these three status bits in the maximum mode.



**RQ / GT1 and RQ** / **GT0**

* The **request/grant** pins request direct memory accesses (DMA) during maximum mode operation.
* These lines are bidirectional and are used to both request and grant a DMA operation.

**LOCK**

* The **lock** output is used to lock peripherals off the system.
* This pin is activated by using the LOCK: prefix on any instruction.

**QS1 and QS0**

* The **queue status** bits show the status of the internal instruction queue.
* These pins are provided for access by the numeric coprocessor (8087).
* Table 9–7 for the operation of the queue status bits.

