4.1 The 8086 Hardware Specifications

The 8086 microprocessor is enclosed in a 40 pin package as shown in figure 4.1:

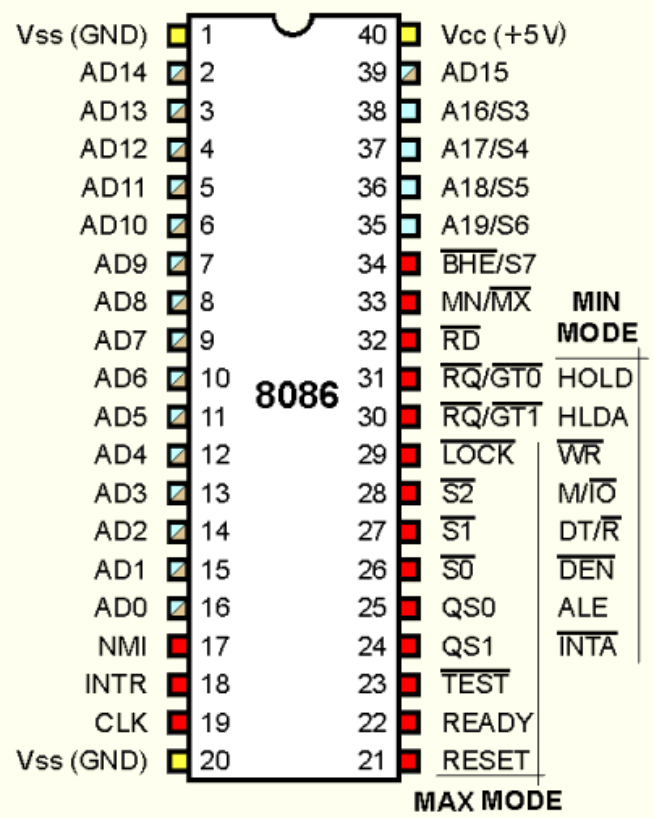

Fig. (4.1) ***The 8086 MP Pin Configuration***

Notice from figure shown above that many of its pins have multiple functions. The 8086 microprocessor can be configured to work either of two modes: the minimum mode and maximum mode.

4.1.1 Minimum Mode

- This mode is selected by making the pin MN/$\overline{MX}$ equal to logic 1.
- Typically smaller systems and contains a single microprocessor.
- Cheaper since all control signals for memory and I/O are generated by the microprocessor.

4.1.2 Maximum Mode

- This mode is selected by making the pin MN/$\overline{MX}$ equal to logic 0.

- Larger systems with more than one processor (designed to be used when a coprocessor exists in the system).
- Some of the control signals must be externally generated. This requires the addition of an external bus controller (8288).

4.2 Minimum Mode Interface Signals

Figure (4.2) illustrates the block diagram of a minimum mode control signal of the 8086 MP. The minimum mode signals can be divided into the following groups:

1. Address/Data Bus.
2. Status Signals.
3. Control Signals.
4. Interrupt Signals.
5. DMA Signals.
6. Power Signals.

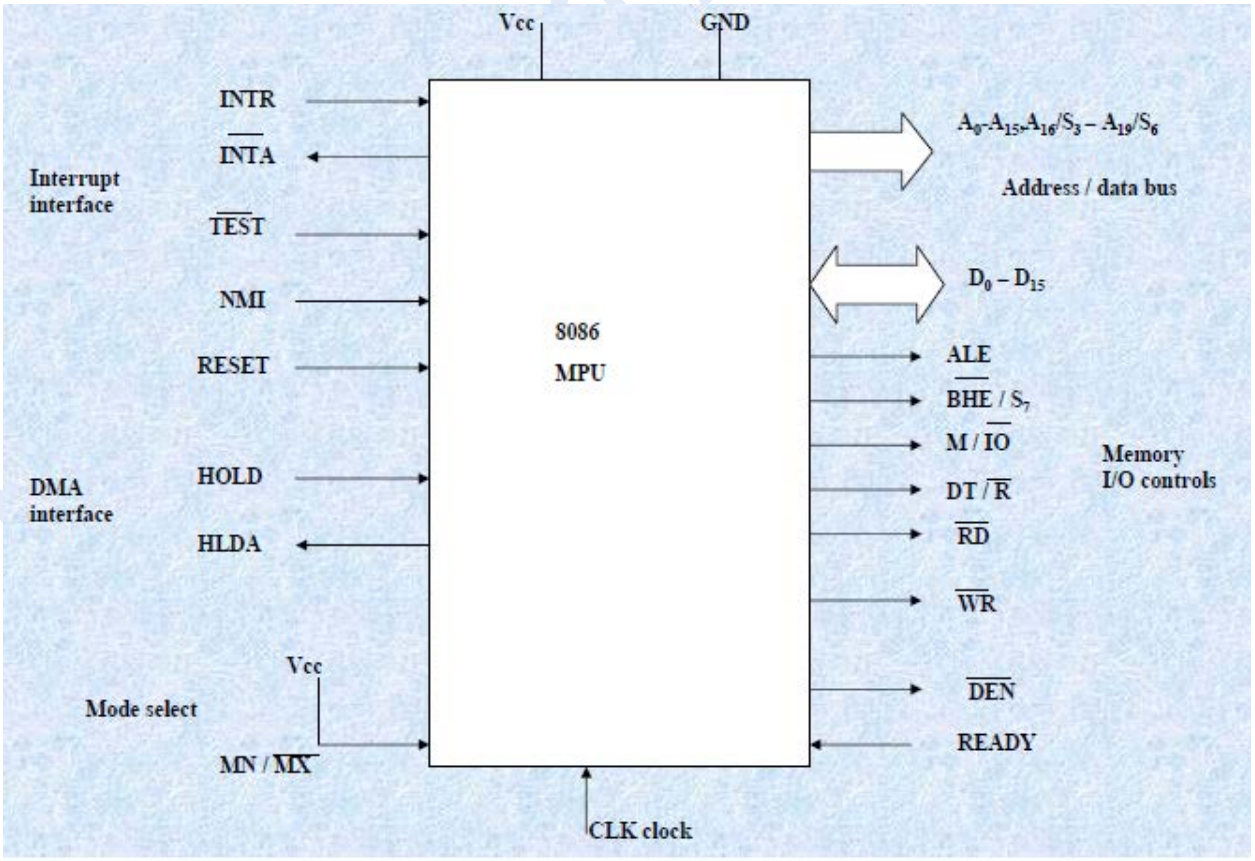

Figure (4.2) ***Block Diagram of the 8086 Minimum Mode***

4.2.1 Address/Data Bus

The address bus is 20 bits long ($A_0 - A_{19}$). The first 16 bits of the address bus are multiplexed with the data bus ($D_0 - A_{15}$) and that results in signals ($AD_0 - A_{D15}$). The last four bits of the address bus ($A_{16} - A_{19}$) are multiplexed with some status signals and that results in signals ($A_{16}/S_3 - A_{19}/S_6$).

The **ALE** signal (pin 25) is set to indicate that valid physical-address is present in the address-bus ($A_0 - A_{19}$) and set to (logic 0) to indicate that the multiplexed lines contains a data and status signals.

4.2.2 Status Signals

As mentioned previously, the status signals ($S_3 - S_6$) are multiplexed with the four most significant bits of the address bus ($A_{16} - A_{19}$) and one of the status signals is multiplexed with control signal **BHE/S_7 .**

S_3 and S_4: a 2 bit binary code that identifies which of the internal segment registers was used to generate the physical address that was output on the address bus during the current bus cycle. The four cases for the S_3 and S_4 signals are shown in table (4.1):

S_5: is the logic level of the internal interrupt enable flag (**IF**).

S_6: is always logic 0.

S_7: is a logic 1 signal.

S_4	S_3	Active SR
0	0	Extra Seg.
0	1	Stack Seg.
1	0	Code Seg.
1	1	Data Seg.

Table (4.1) ***Cases of S_3 and S_4 Signals***

4.2.3 Control Signals

These signals are used to support memory and I/O interfaces to the 8086 microprocessor. These are:

- **ALE (Address Latch Enable)**

  This signal is used to demultiplex the address bus from the data bus and the status signals.

- **$\overline{BHE}/S_7$ (Bank High Enable)**

  This signal is used to enable data onto the most significant half of the data bus, pins ($D_8 - A_{15}$) during a read or write operation.

- **M/$\overline{IO}$ (Memory/ Input-Output)**

  This pin indicates whether the address bus is holding a memory address or an I/O address.

- **DT/$\overline{R}$ (Data Transmit Receive)**

  Specifies that the data bus is either transmitting data (logic 1) to memory or output port (writing) or receiving data (logic 0) from memory or input port (reading).

- **$\overline{DEN}$ (Data Bus Enable)**

  Enables the external devices to supply data to the processor. Used when sharing memory with another processor.

- **$\overline{RD}$ (Read)**

  Indicates that the processor is performing a read memory or read I/O cycle, depending on the state of the M/$\overline{IO}$ signal.

- **$\overline{WR}$ (Write)**

  Indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the M/$\overline{IO}$ signal.

- **READY (Ready)**

  In 8086 microprocessor, pin 22 accepts **READY** signal from external circuits to prolong the BUS-Cycle, by inserting wait-status if required. This enhances the compatibility between high-speed CPU with relatively slower peripheral devices.