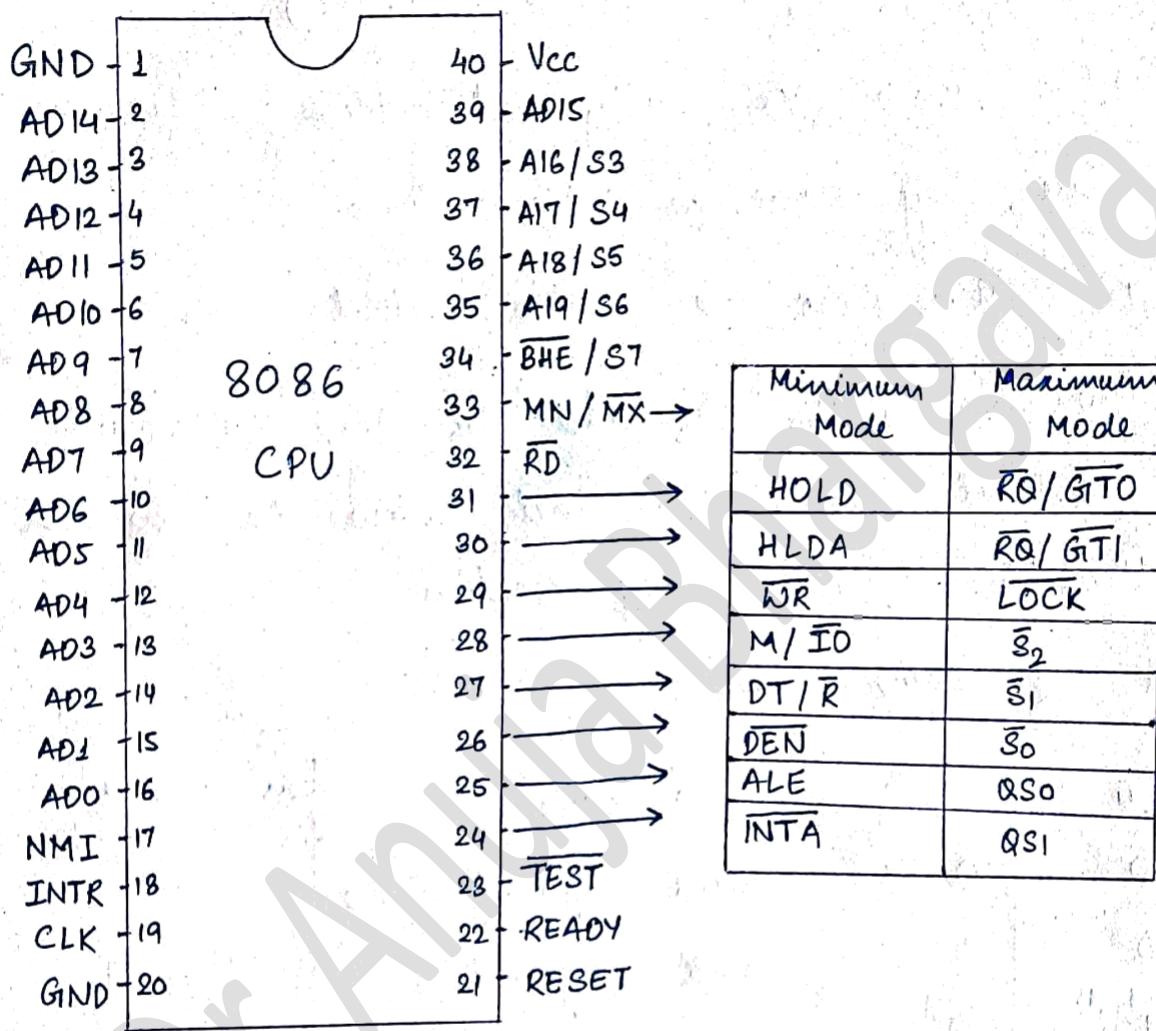


# ~~8086~~ 8086 Pin Configuration

- The 8086 have a 20-bit address bus, 16-bit data bus, three power supply pins and 17 pins devoted to control and timing functions.



- 8086 has ability to operate in two modes

- Minimum Mode (Unprocessor System)
- Maximum Mode (Multiprocessor System)

Minimum mode is used for small system in which 8086 generates all necessary signals. Maximum mode is used for medium to large system which includes two or more processor

## 1) Address / Data Bus ( $AD_0 - AD_{15}$ )

- $AD_0 - AD_{15}$  lines are 16-bit multiplexed address.
- During  $T_1$  state,  $A_0 - A_{15}$  contain address.

## 2) Address / Status lines

( $A_{16}/S_3, A_{17}/S_4, A_{18}/S_5, A_{19}/S_6$ )

- $A_{16}/S_3 - A_{19}/S_6$  lines are multiplexed address / status bus.
- During  $T_1$  state, they seen as address lines.  
During  $T_2 - T_4$  state, these are status output lines.

Note : The 20 pins ( $AD_0 - AD_{15}$  and  $A_{16} - A_{19}$ ) corresponds to the CPU's 20 bit address bus and allow the processor to access 1,048,576 unique memory location.

## 3) Characteristics of Status lines

$S_4$	$S_3$	Characteristic
0	0	Extra Segment Access
0	1	Stack Segment Access
1	0	Code Segment Access
1	1	Data Segment Access

$S_5$  - Indicates the status of interrupt enable flag.

$S_6$  - It is always zero, and no significance is attached to it.

### 3) Bus High Enable / Status ( $\overline{BHE} / S_7$ )

- The 8086 outputs a low on this pin during read / write and interrupt acknowledge cycles in which data are to be transferred from an odd address in higher order byte ( $AD_{15} - AD_8$ ) of the data bus or word transfer take place from even address on  $AD_{15} - AD_0$ .
- $\overline{BHE}$  can be used with  $AD_0$  to select even / odd memory or I/O ports.

Operation	$\overline{BHE}$	$AD_0$
1. Write / Read a word at even address	0	0
2. Write / Read a byte at even address	1	0
3. Write / Read a byte at odd address	0	1
4. Write / Read a word at odd address	0	1

### 4) NMI : Non Maskable Interrupt

- It is positive edge triggered interrupt.
- The NMI has higher priority than INTR.
- It is a vectored interrupt, that means 8086 knows where to branch, to service NMI interrupt.

### 5) INTR : Interrupt:

- It is a maskable interrupt input.
- It can be enable by STI (Set Intercept flag) and disabled by CLI (Clear Intercept flag) instruction.

### 6) CLK : Clock:

- The 8086 comes in different clock frequency in the range between 4 to 8 MHz.

Version	Frequency
8086 - 1	10 MHz
8086 - 2	8 MHz
8086	5 MHz

### 7) Reset :

- It terminates the processor activity.
- It can be done by reset button or power on button.

### 8) Ready :

- This pin is used for interfacing peripherals with processor.
- It is acknowledgement signal from memory or I/O interface that the CPU can complete the current bus cycle.

9) TEST :

- It is used to synchronize the 8086 with external hardware such as coprocessor.
- It is examined by the CPU during the 'wait' instruction in multiprocessor environment.

10) RD : (Read)

- It is active low signal when 8086 is reading data from memory or I/O location.

11) V<sub>cc</sub> :

Supply Voltage  $\pm 5V \pm 10\%$ .

12) GND (Ground)

2 Pins are used for ground.

13) MN / MX (Minimum / Maximum Mode)

It is high for minimum mode and low for maximum (uniprocessor)

(Multiprocessor) mode operation.

Pins for Minimum Mode :-

1) INTA (Interrupt Acknowledge)

- It ~~gives~~ indicates recognition of an interrupt request.

2) ALE (Address latch Enable)

- It is used to demultiplex AD<sub>0</sub> - AD<sub>15</sub> into A<sub>0</sub> - A<sub>15</sub> and D<sub>0</sub> - D<sub>15</sub>. It indicate address is available on the pin.

### 3) $\overline{DEN}$ (Data Enable)

- It is low during memory or I/O read / write.
- It becomes low, little after ALE is made 0, so that the data on the data pins are settled.

### 4) $DT/R$ (Data Transmit / Receive)

- It is an output signal controls the direction of data flow through transceivers.
- If it is 1, the buffer is used to transmit the data.
- If it is 0, the buffer is used to receive the data.

### 5) $M/\overline{IO}$ : (Memory / I/O Operation)

- For memory operation  $M/\overline{IO} = 1$
- I/O operation  $M/\overline{IO} = 0$

### 6) $\overline{WR}$ (Write):

- It is used to indicate that processor is performing write Memory or write I/O operation.

### 7) HOLD & HLDA (Hold Acknowledge)

- The 8086 processor is in HOLD situation whenever DMA is required to access directly.
- The microprocessor indicates to the DMA controller that it has entered the hold state, by sending HLDA signal.

## Pins for Maximum Mode

47

1)  $QS_1, QS_0$  (Queue Status)

- It reflect the status of the instruction queue.

$QS_1$	$QS_0$	Interpretation
0	0	No operation
0	1	First byte of instruction is executed.
1	0	Queue is empty.
1	1	Next byte of instruction is executed.

2)  $\bar{S}_0, \bar{S}_1, \bar{S}_2$  (Status Signal)

- These are used for timing and control signals.

$\bar{S}_2$	$\bar{S}_1$	$\bar{S}_0$	Function
0	0	0	Interrupt Acknowledge
0	0	1	Read I/O Port
0	1	0	Write I/O Port
0	1	1	Halt
1	0	0	Instruction fetch
1	0	1	Read Memory
1	1	0	Write Memory
1	1	1	Inactive

### 3) LOCK :

- It is active low signal used to prevent other processor from getting access to the system bus.

### 4) $\overline{RQ}/\overline{GT}_0$ & $\overline{RQ}/\overline{GT}_1$ : (Request & Grant lines)

- It is used to force the processor to release the local bus at the end of processor cycle.
- $\overline{RQ}/\overline{GT}_0$  having higher priority than  $\overline{RQ}/\overline{GT}_1$ .

• The request / grant function of 8086 are as follows

- (i) A pulse from another local bus master ( $\overline{RQ}/\overline{GT}_0$  or  $\overline{RQ}/\overline{GT}_1$  pins) indicates a local bus request to the 8086.
- (ii) At the end of 8086 current bus cycle, a pulse from 8086 to the requesting master indicates that the 8086 has the system bus. and tri-stated the outputs.

Then the new bus master sends a low signal on  $\overline{RQ}/\overline{GT}_0$  or  $\overline{RQ}/\overline{GT}_1$  pins.

The 8086 then regains bus control.