

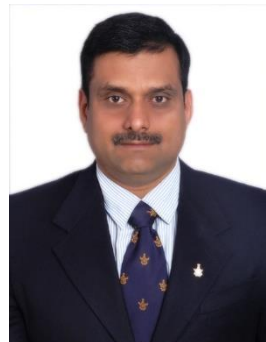


# **Microprocessors and Interfaces: 2021-22**

## **Lecture 3**

### **8086 Pin Diagram and Architecture**

**By Dr. Sanjay Vidhyadharan**

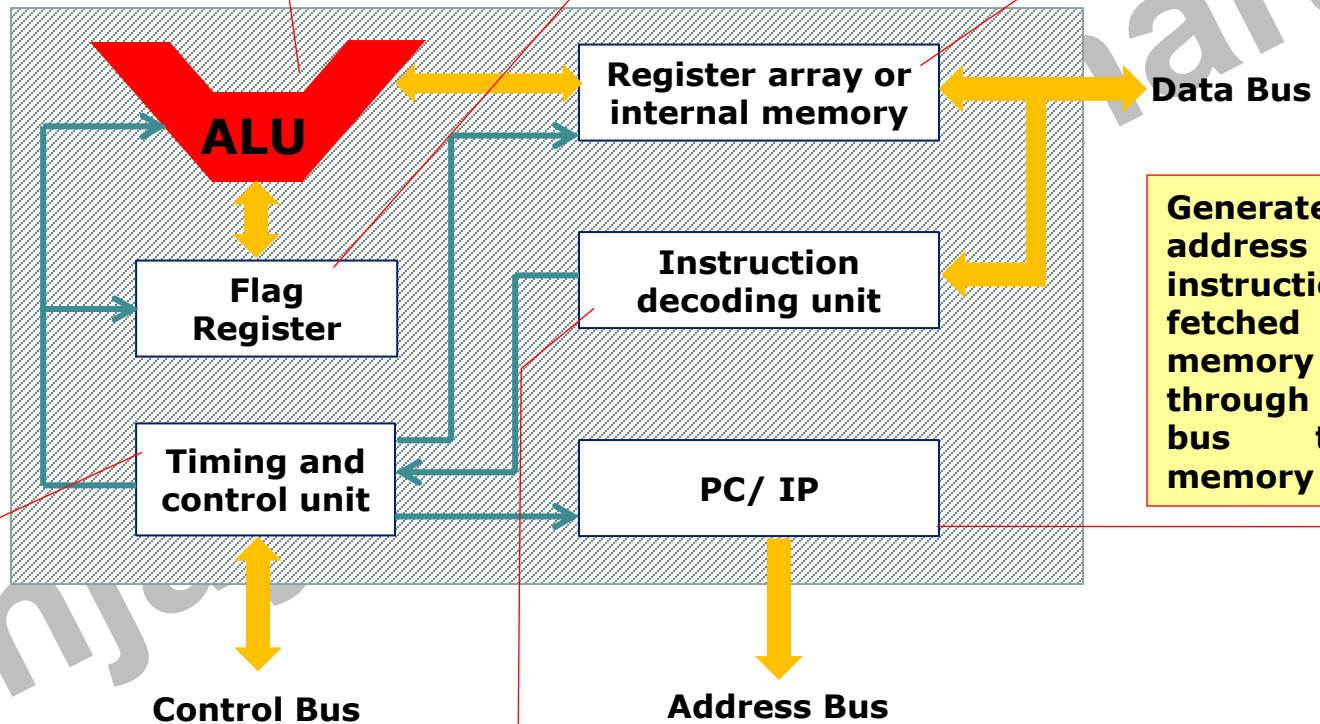


# Functional blocks

Computational Unit; performs arithmetic and logic operations

Various conditions of the results are stored as status bits called flags in flag register

Internal storage of data



Generates the address of the instructions to be fetched from the memory and send through address bus to the memory

Generates control signals for internal and external operations of the microprocessor

Decodes instructions; sends information to the timing and control unit

# Functional blocks

First 16-bit processor released by INTEL in the year 1978

Originally HMOS, now manufactured using HMOS III technique

Approximately 29,000 transistors, 40 pin DIP, 5V supply

Does not have internal clock; external asymmetric clock source with 33% duty cycle

20-bit address to access memory  $\Rightarrow$  can address up to  $2^{20} = 1$  megabytes of memory space.

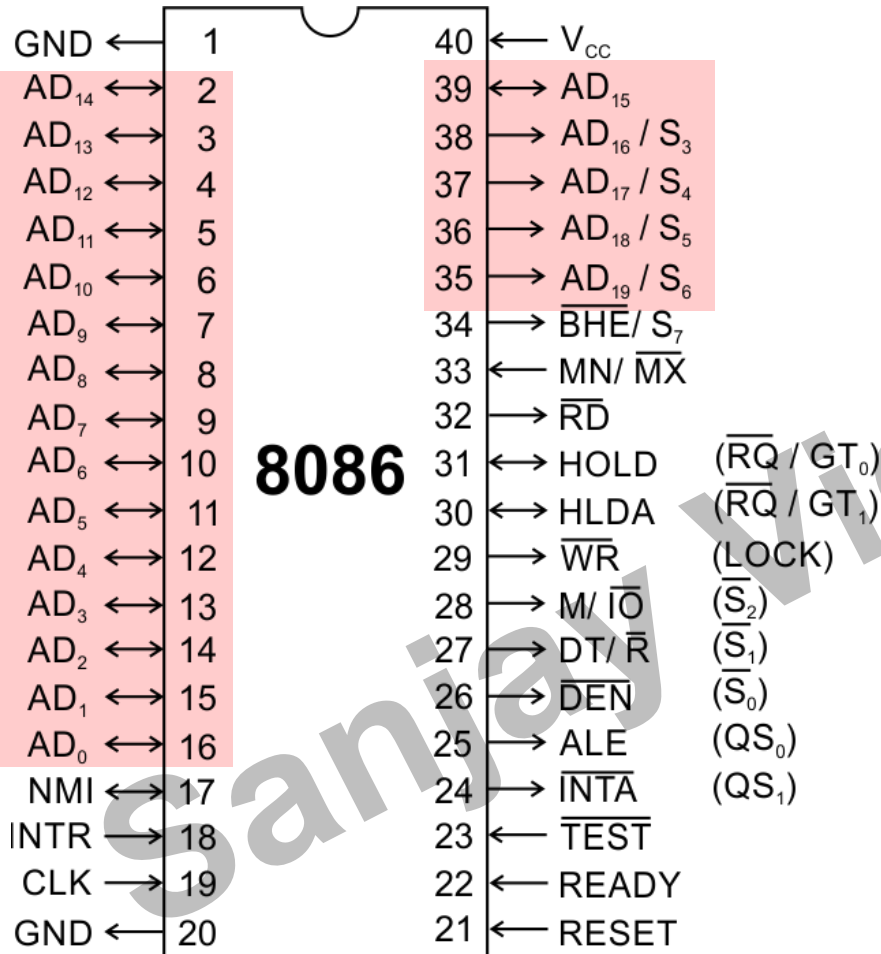
Addressable memory space is organized into two banks of 512 kb each; **Even (or lower) bank** and **Odd (or higher) bank**. Address line  $A_0$  is used to select even/odd bank and control signal  $\overline{BHE}$  to select 16 bit data.

Uses a separate 16 bit address for I/O mapped devices  $\Rightarrow$  can generate  $2^{16} = 64$  k addresses.

Operates in two modes: **minimum mode** and **maximum mode**, decided by the signal at MN and  $\overline{MX}$  pins.

# Pins and Signals

## Common signals



### AD<sub>0</sub>-AD<sub>15</sub> (Bidirectional)

#### Address/Data bus

Low order address bus; these are multiplexed with data.

When AD lines are used to transmit memory address the symbol A is used instead of AD, for example A<sub>0</sub>-A<sub>15</sub>.

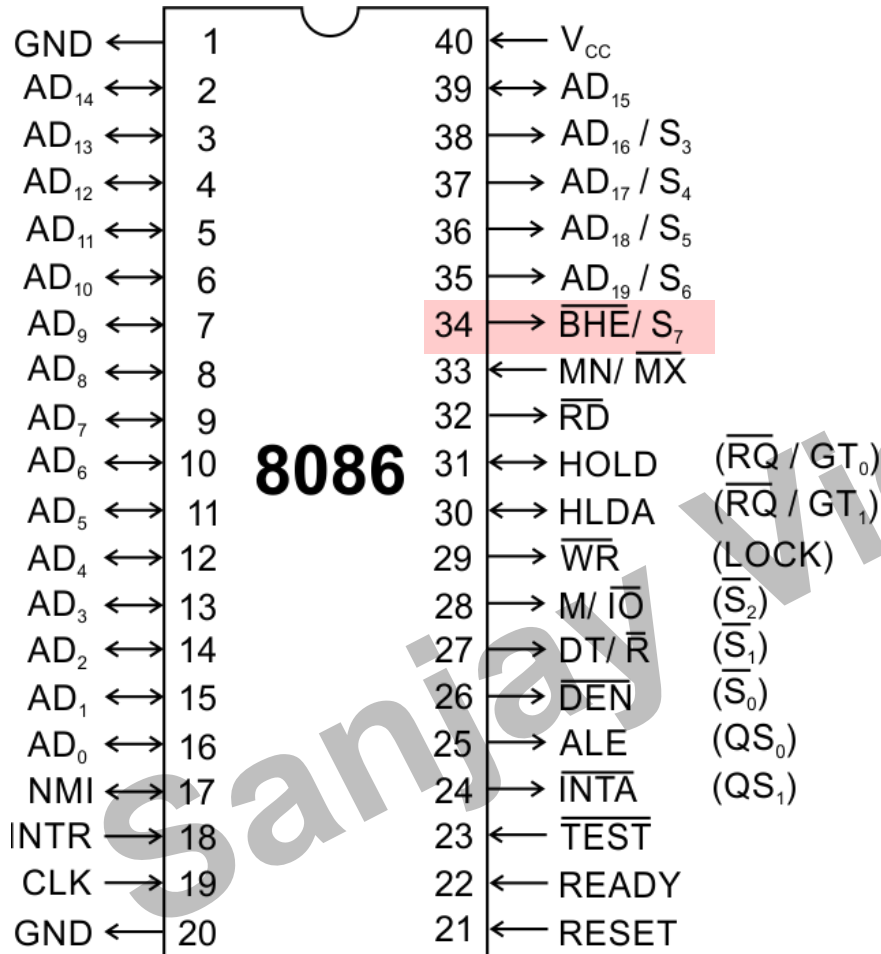
When data are transmitted over AD lines the symbol D is used in place of AD, for example D<sub>0</sub>-D<sub>7</sub>, D<sub>8</sub>-D<sub>15</sub> or D<sub>0</sub>-D<sub>15</sub>.

### A<sub>16</sub>/S<sub>3</sub>, A<sub>17</sub>/S<sub>4</sub>, A<sub>18</sub>/S<sub>5</sub>, A<sub>19</sub>/S<sub>6</sub>

High order address bus. These are multiplexed with status signals

# Pins and Signals

## Common signals



### BHE (Active Low)/S<sub>7</sub> (Output)

#### Bus High Enable/Status

It is used to enable data onto the most significant half of data bus, D<sub>8</sub>-D<sub>15</sub>. 8-bit device connected to upper half of the data bus use BHE (Active Low) signal. It is multiplexed with status signal S<sub>7</sub>.

### MN / MX

#### MINIMUM / MAXIMUM

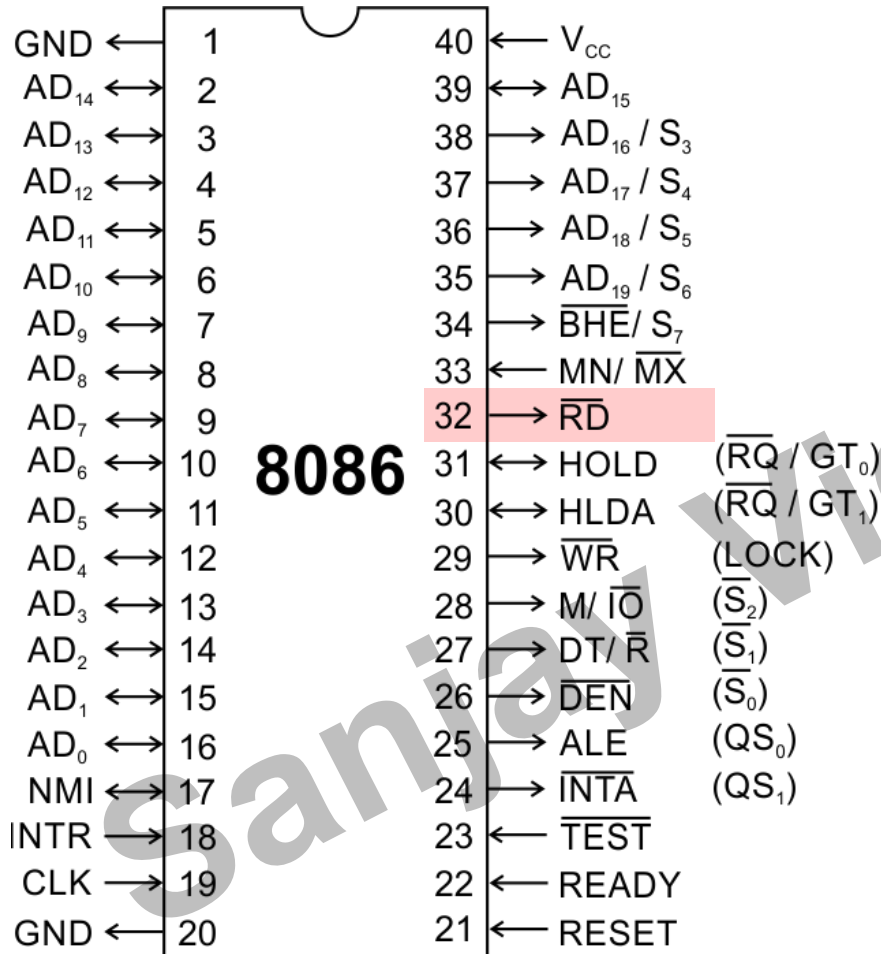
This pin signal indicates what mode the processor is to operate in.

### RD (Read) (Active Low)

The signal is used for read operation.  
It is an output signal.  
It is active when low.

# Pins and Signals

## Common signals



### TEST

**TEST** input is tested by the 'WAIT' instruction.

**8086** will enter a wait state after execution of the WAIT instruction and will resume execution only when the **TEST** is made low by an active hardware.

This is used to synchronize an external activity to the processor internal operation.

### READY

This is the acknowledgement from the slow device or memory that they have completed the data transfer.

The signal made available by the devices is synchronized by the 8284A clock generator to provide ready input to the **8086**.

The signal is active<sup>6</sup>high.

# Pins and Signals

## Min/ Max Pins

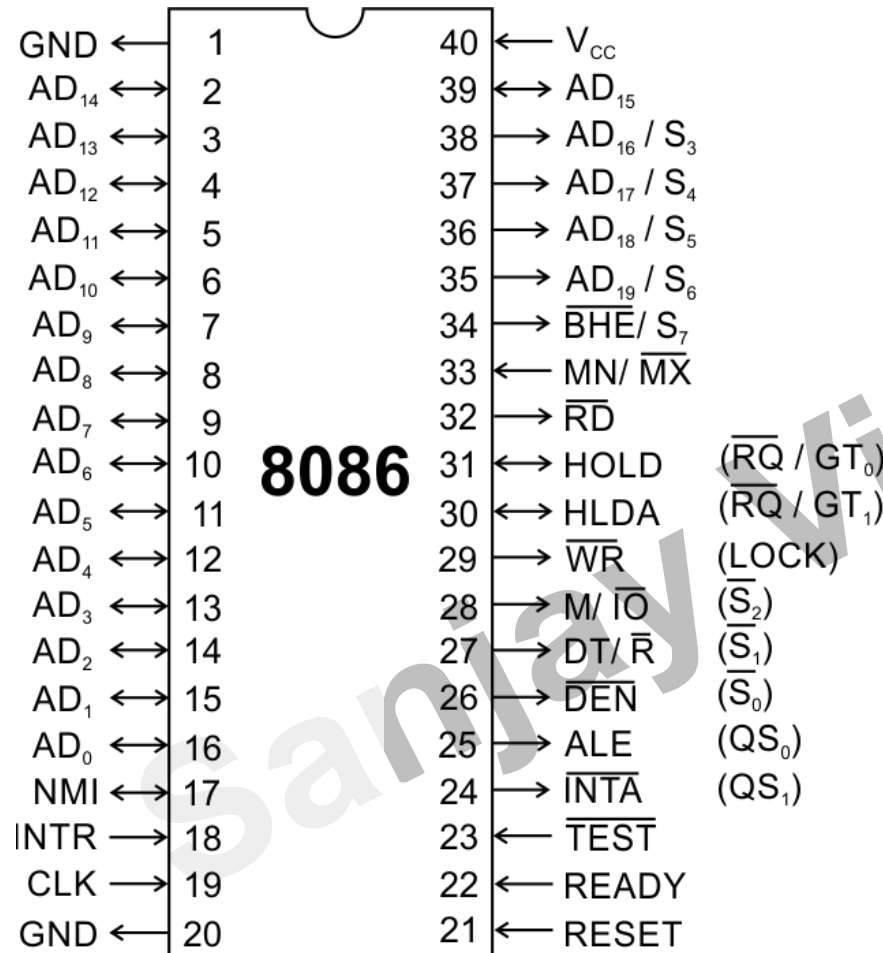
The 8086 microprocessor can work in two modes of operations : **Minimum mode** and **Maximum mode**.

In the minimum mode of operation the microprocessor do not associate with any co-processors and can not be used for multiprocessor systems.

In the maximum mode the 8086 can work in multi-processor or co-processor configuration.

Minimum or maximum mode operations are decided by the pin MN/ MX(Active low).

When this pin is high 8086 operates in minimum mode otherwise it operates in **Maximum mode**.



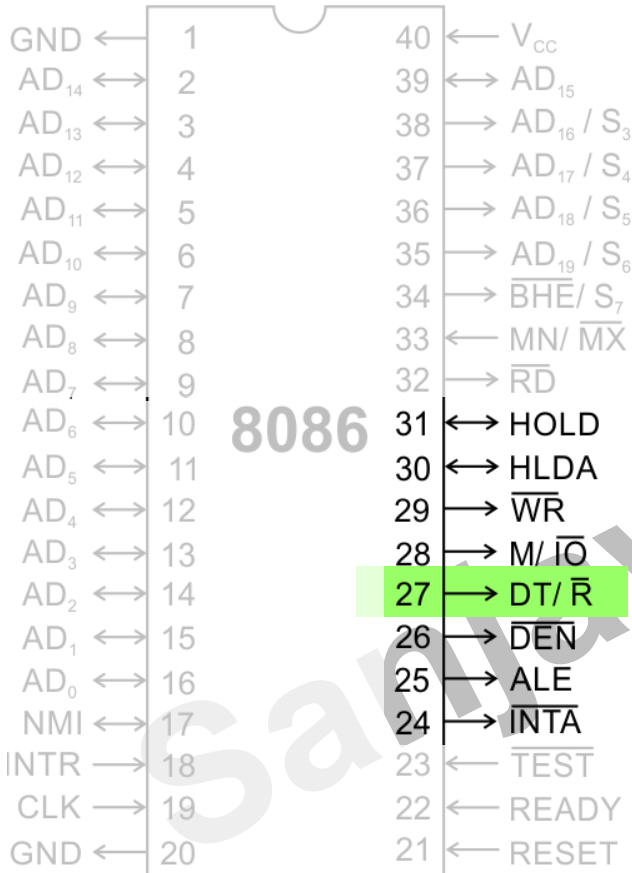
# Pins and Signals

## Minimum mode signals

### Pins 24 -31

For minimum mode operation, the  $\overline{MN}/\overline{MX}$  is tied to VCC (logic high)

8086 itself generates all the bus control signals



**DT/ $\overline{R}$**

**(Data Transmit/ Receive)** Output signal from the processor to control the direction of data flow through the data transceivers

**$\overline{DEN}$**

**(Data Enable)** Output signal from the processor used as out put enable for the transceivers

**ALE**

**(Address Latch Enable)** Used to demultiplex the address and data lines using external latches

**M/ $\overline{IO}$**

Used to differentiate memory access and I/O access. For memory reference instructions, it is **high**. For IN and OUT instructions, it is **low**.

**$\overline{WR}$**

Write control signal; asserted **low** Whenever processor writes data to memory or I/O port

**$\overline{INTA}$**

**(Interrupt Acknowledge)** When the interrupt request is accepted by the processor, the output is **low** on this line.



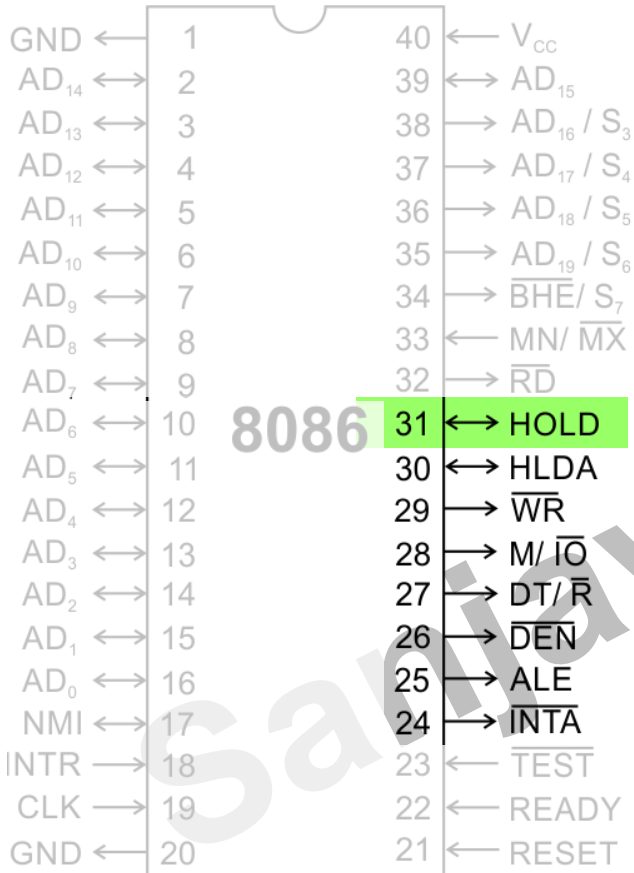
# Pins and Signals

## Minimum mode signals

### Pins 24 -31

For minimum mode operation, the MN/  $\overline{\text{MX}}$  is tied to VCC (logic high)

**8086 itself generates all the bus control signals**



#### HOLD

Input signal to the processor from the bus masters as a request to grant the control of the bus.

Usually used by the DMA controller to get the control of the bus.

#### HLDA

(Hold Acknowledge) Acknowledge signal by the processor to the bus master requesting the control of the bus through HOLD.

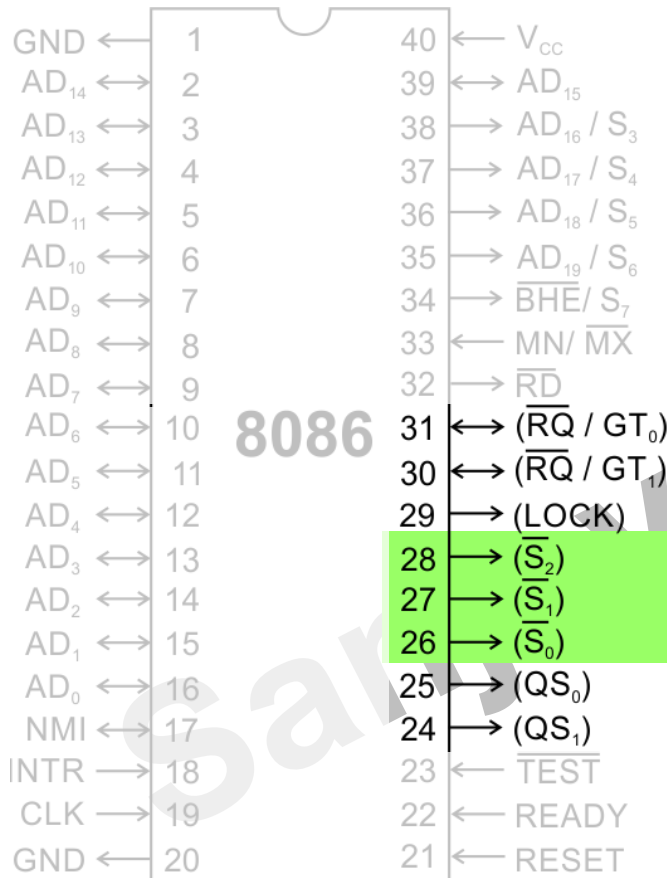
The acknowledge is asserted high, when the processor accepts HOLD.

# Pins and Signals

## Maximum mode signals

During maximum mode operation, the  $\overline{MN}/\overline{MX}$  is grounded (logic low)

Pins 24 -31 are reassigned



$\overline{S}_0, \overline{S}_1, \overline{S}_2$

**Status signals;** used by the 8086 bus controller to generate bus timing and control signals. These are decoded as shown.

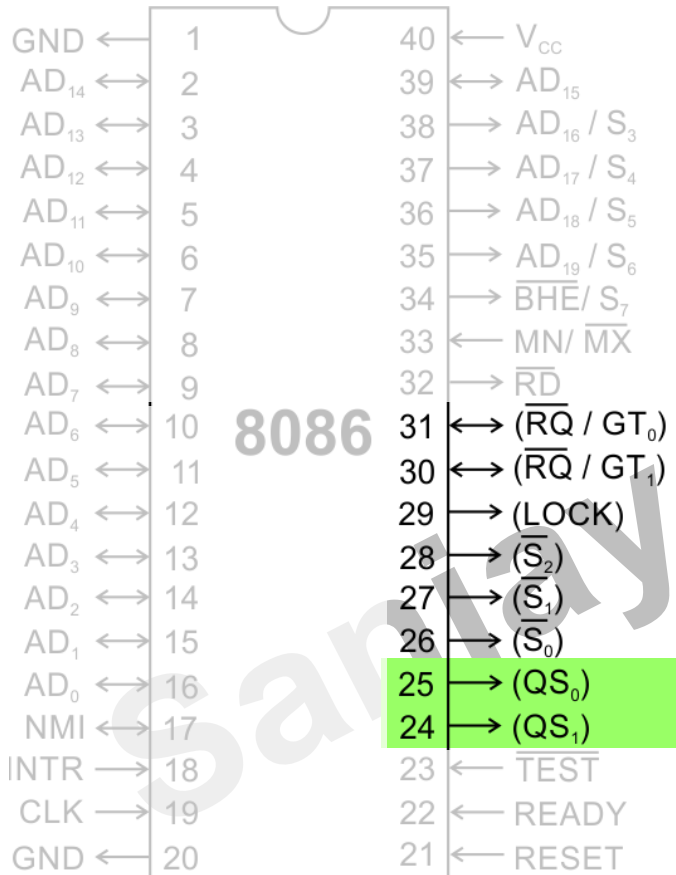
Status Signal			Machine Cycle
$\overline{S}_2$	$\overline{S}_1$	$\overline{S}_0$	
0	0	0	Interrupt acknowledge
0	0	1	Read I/O port
0	1	0	Write I/O port
0	1	1	Halt
1	0	0	Code access
1	0	1	Read memory
1	1	0	Write memory
1	1	1	Passive/Inactive

# Pins and Signals

## Maximum mode signals

During maximum mode operation, the  $MN/\overline{MX}$  is grounded (logic low)

Pins 24 -31 are reassigned



$QS_0, QS_1$

**(Queue Status)** The processor provides the status of queue in these lines.

The queue status can be used by external device to track the internal status of the queue in 8086.

The output on  $QS_0$  and  $QS_1$  can be interpreted as shown in the table.

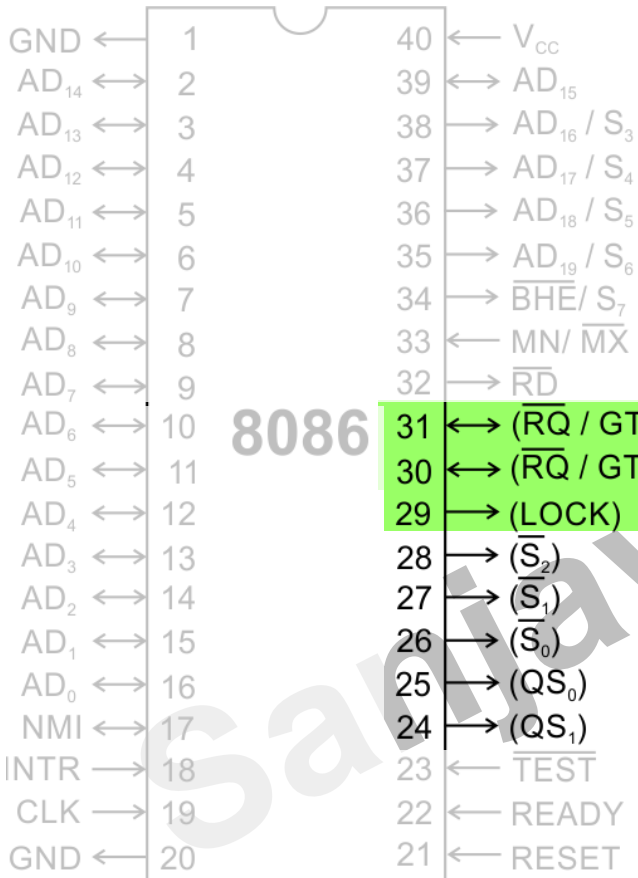
Queue status		Queue operation
$QS_1$	$QS_0$	
0	0	No operation
0	1	First byte of an opcode from queue
1	0	Empty the queue
1	1	Subsequent byte from queue

# Pins and Signals

## Maximum mode signals

During maximum mode operation, the MN/  $\overline{\text{MX}}$  is grounded (logic low)

Pins 24 -31 are reassigned



$\overline{\text{RQ}}/\text{GT}_0$ ,  
 $\overline{\text{RQ}}/\text{GT}_1$

(**Bus Request/ Bus Grant**) These requests are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle.

These pins are bidirectional.

The request on  $\overline{\text{GT}}_0$  will have higher priority than  $\overline{\text{GT}}_1$

$\overline{\text{LOCK}}$

An output signal activated by the LOCK prefix instruction.

Remains active until the completion of the instruction prefixed by LOCK.

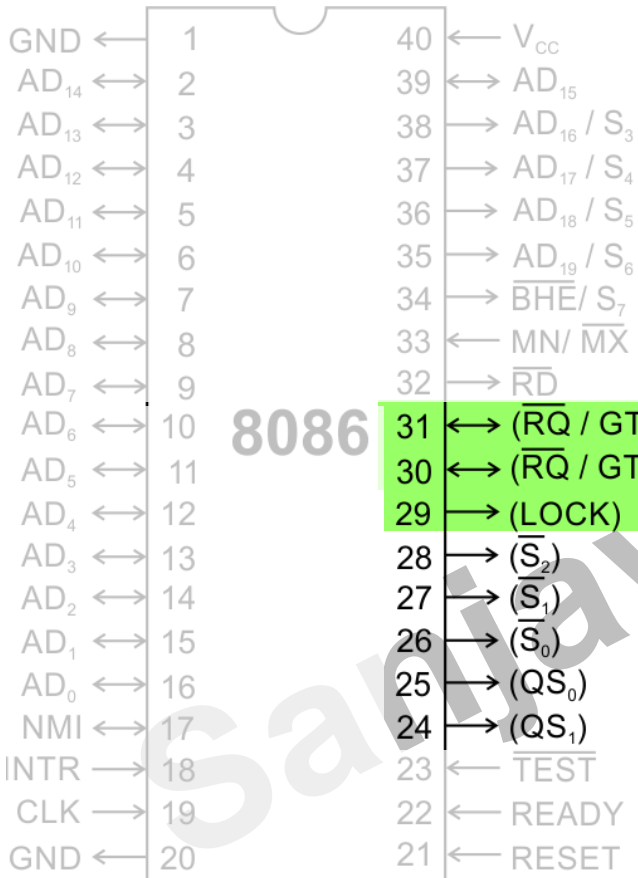
The 8086 output low on the  $\overline{\text{LOCK}}$  pin while executing an instruction prefixed by LOCK to prevent other bus masters from gaining control of the system bus.

# Pins and Signals

## Maximum mode signals

During maximum mode operation, the MN/  $\overline{\text{MX}}$  is grounded (logic low)

Pins 24 -31 are reassigned



$\overline{\text{RQ}}/\text{GT}_0$ ,  
 $\overline{\text{RQ}}/\text{GT}_1$

(**Bus Request/ Bus Grant**) These requests are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle.

These pins are bidirectional.

The request on  $\overline{\text{GT}}_0$  will have higher priority than  $\overline{\text{GT}}_1$

$\overline{\text{LOCK}}$

An output signal activated by the LOCK prefix instruction.

Remains active until the completion of the instruction prefixed by LOCK.

The 8086 output low on the  $\overline{\text{LOCK}}$  pin while executing an instruction prefixed by LOCK to prevent other bus masters from gaining control of the system bus.

---

**Thankyou**

Sanjay Vidhyadharan