

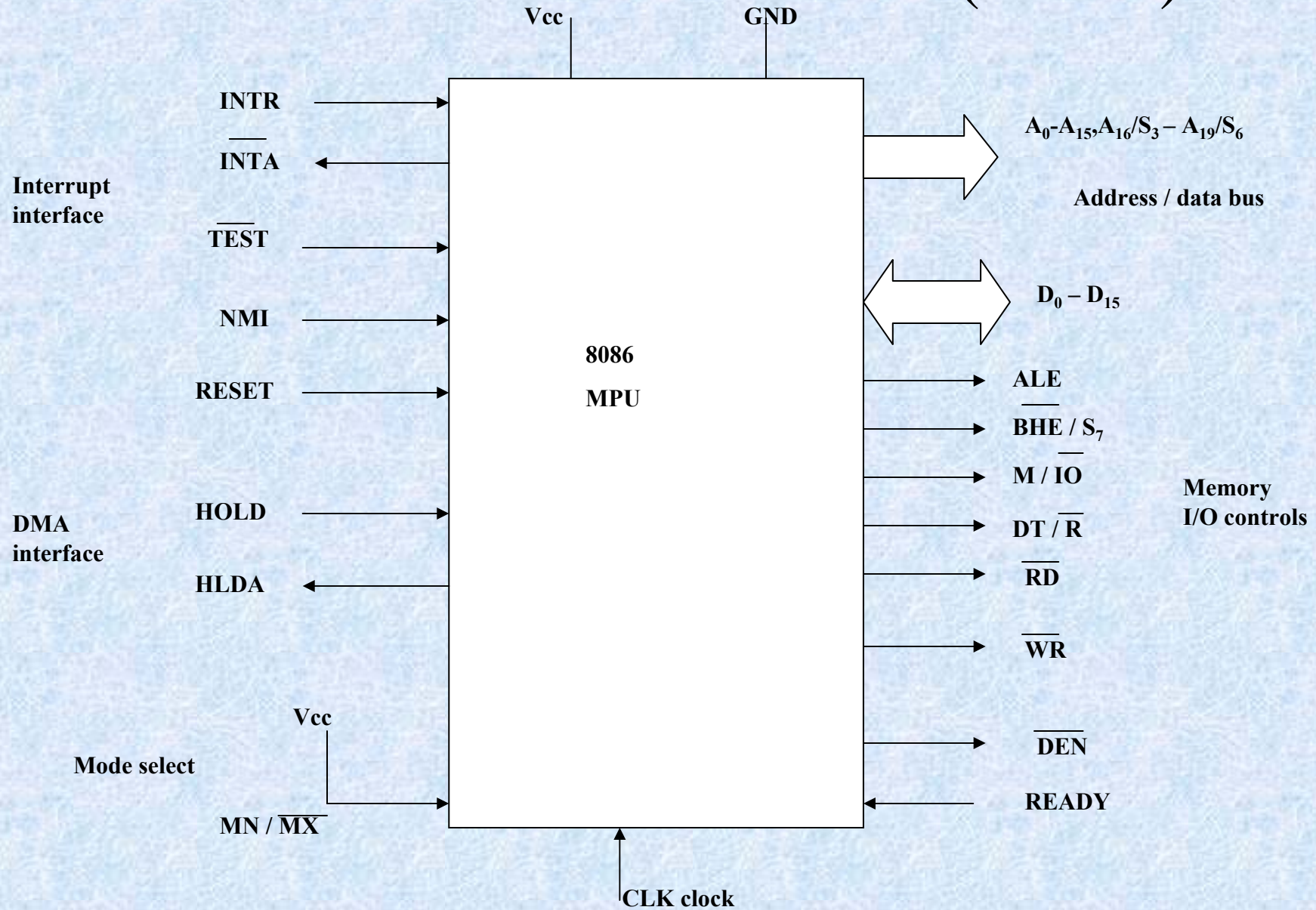
# Minimum Mode Interface

- When the Minimum mode operation is selected, the 8086 provides all control signals needed to implement the memory and I/O interface.
- The minimum mode signal can be divided into the following basic groups : address/data bus, status, control, interrupt and DMA.
- **Address/Data Bus** : these lines serve two functions. As an address bus is 20 bits long and consists of signal lines  $A_0$  through  $A_{19}$ .  $A_{19}$  represents the MSB and  $A_0$  LSB. A 20bit address gives the 8086 a 1Mbyte memory address space. More over it has an independent I/O address space which is 64K bytes in length.

## Minimum Mode Interface ( cont..)

- The 16 data bus lines  $D_0$  through  $D_{15}$  are actually multiplexed with address lines  $A_0$  through  $A_{15}$  respectively. By multiplexed we mean that the bus work as an address bus during first machine cycle and as a data bus during next machine cycles.  $D_{15}$  is the MSB and  $D_0$  LSB.
- When acting as a data bus, they carry read/write data for memory, input/output data for I/O devices, and interrupt type codes from an interrupt controller.

# Minimum Mode Interface ( cont..)



Block Diagram of the Minimum Mode 8086 MPU

## Minimum Mode Interface ( cont..)

- **Status signal** : The four most significant address lines  $A_{19}$  through  $A_{16}$  are also multiplexed but in this case with status signals  $S_6$  through  $S_3$ . These status bits are output on the bus at the same time that data are transferred over the other bus lines.
- Bit  $S_4$  and  $S_3$  together form a 2 bit binary code that identifies which of the 8086 internal segment registers are used to generate the physical address that was output on the address bus during the current bus cycle.
- Code  $S_4S_3 = 00$  identifies a register known as *extra segment register* as the source of the segment address.



## Minimum Mode Interface ( cont..)

$S_4$	$S_3$	Segment Register
0	0	Extra
0	1	Stack
1	0	Code / none
1	1	Data

Memory segment status codes.

## Minimum Mode Interface ( cont..)

- Status line  $S_5$  reflects the status of another internal characteristic of the 8086. It is the logic level of the internal enable flag. The last status bit  $S_6$  is always at the logic 0 level.
- **Control Signals** : The control signals are provided to support the 8086 memory I/O interfaces. They control functions such as when the bus is to carry a valid address in which direction data are to be transferred over the bus, when valid write data are on the bus and when to put read data on the system bus.

## Minimum Mode Interface ( cont..)

- ALE is a pulse to logic 1 that signals external circuitry when a valid address word is on the bus. This address must be latched in external circuitry on the 1-to-0 edge of the pulse at ALE.
- Another control signal that is produced during the bus cycle is BHE bank high enable. Logic 0 on this used as a memory enable signal for the most significant byte half of the data bus  $D_8$  through  $D_1$ . These lines also serves a second function, which is as the  $S_7$  status line.
- Using the M/IO and DT/R lines, the 8086 signals which type of bus cycle is in progress and in which direction data are to be transferred over the bus.

## Minimum Mode Interface ( cont..)

- The logic level of  $\overline{M}/IO$  tells external circuitry whether a memory or I/O transfer is taking place over the bus. Logic 1 at this output signals a memory operation and logic 0 an I/O operation.
- The direction of data transfer over the bus is signaled by the logic level output at  $DT/\overline{R}$ . When this line is logic 1 during the data transfer part of a bus cycle, the bus is in the transmit mode. Therefore, data are either written into memory or output to an I/O device.
- On the other hand, logic 0 at  $DT/\overline{R}$  signals that the bus is in the receive mode. This corresponds to reading data from memory or input of data from an input port.



## Minimum Mode Interface ( cont..)

- The signal read  $\overline{RD}$  and write  $\overline{WR}$  indicates that a read bus cycle or a write bus cycle is in progress. The 8086 switches  $\overline{WR}$  to logic 0 to signal external device that valid write or output data are on the bus.
- On the other hand,  $\overline{RD}$  indicates that the 8086 is performing a read of data of the bus. During read operations, one other control signal is also supplied. This is  $\overline{DEN}$  ( data enable) and it signals external devices when they should put data on the bus.
- There is one other control signal that is involved with the memory and I/O interface. This is the READY signal.

## Minimum Mode Interface ( cont..)

- **READY** signal is used to insert wait states into the bus cycle such that it is extended by a number of clock periods. This signal is provided by an external clock generator device and can be supplied by the memory or I/O subsystem to signal the 8086 when they are ready to permit the data transfer to be completed.
- **Interrupt signals** : The key interrupt interface signals are interrupt request (INTR) and interrupt acknowledge ( $\overline{\text{INTA}}$ ).
- INTR is an input to the 8086 that can be used by an external device to signal that it need to be serviced.

## Minimum Mode Interface ( cont..)

- Logic 1 at INTR represents an active interrupt request. When an interrupt request has been recognized by the 8086, it indicates this fact to external circuit with pulse to logic 0 at the  $\overline{INTA}$  output.
- The  $\overline{TEST}$  input is also related to the external interrupt interface. Execution of a WAIT instruction causes the 8086 to check the logic level at the  $\overline{TEST}$  input.
- If the logic 1 is found, the MPU suspend operation and goes into the idle state. The 8086 no longer executes instructions, instead it repeatedly checks the logic level of the  $\overline{TEST}$  input waiting for its transition back to logic 0.



## Minimum Mode Interface ( cont..)

- As  $\overline{\text{TEST}}$  switches to 0, execution resume with the next instruction in the program. This feature can be used to synchronize the operation of the 8086 to an event in external hardware.
- There are two more inputs in the interrupt interface: the nonmaskable interrupt NMI and the reset interrupt RESET.
- On the 0-to-1 transition of NMI control is passed to a nonmaskable interrupt service routine. The RESET input is used to provide a hardware reset for the 8086. Switching RESET to logic 0 initializes the internal register of the 8086 and initiates a reset service routine.



# Minimum Mode Interface.

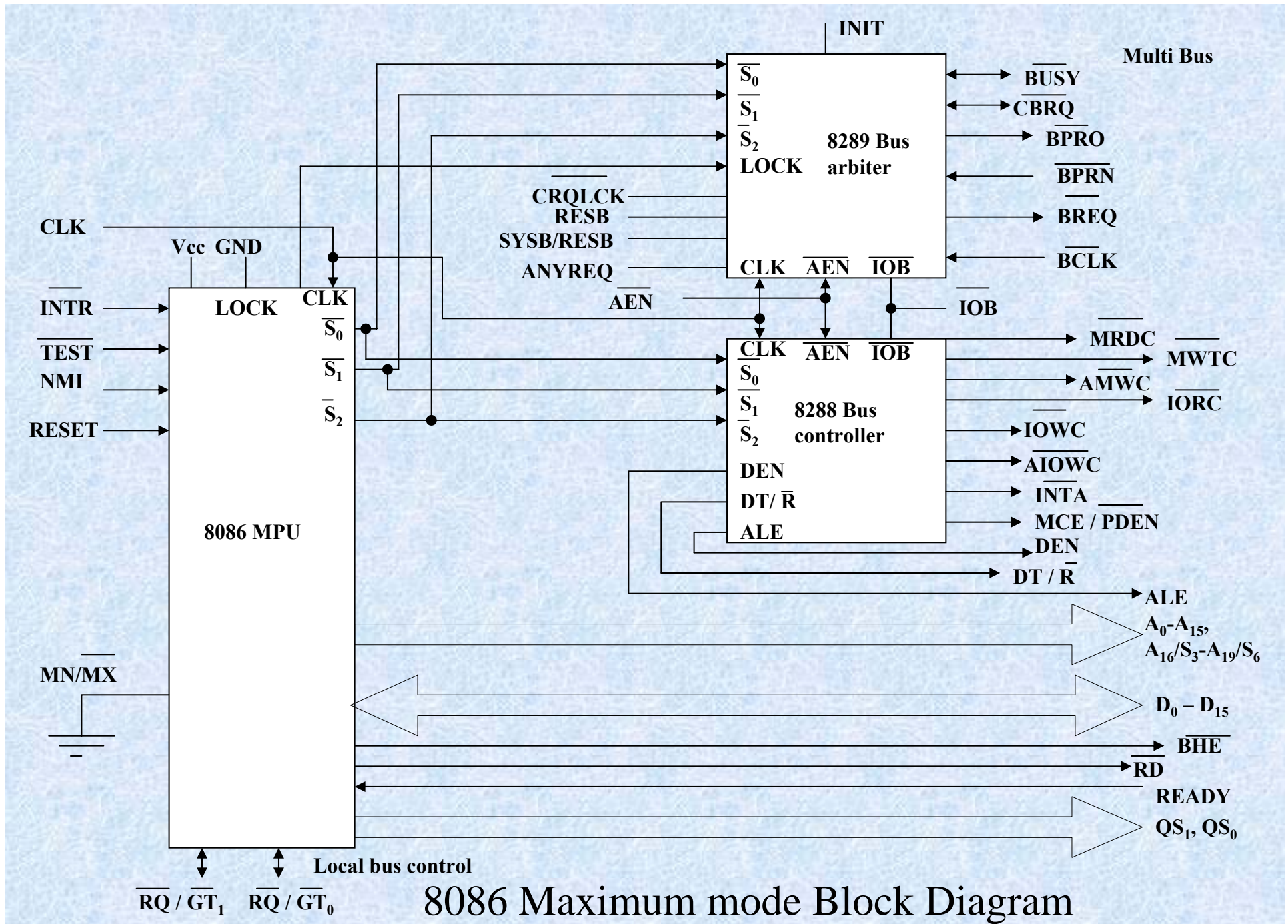
- **DMA Interface signals** :The direct memory access DMA interface of the 8086 minimum mode consist of the HOLD and HLDA signals.
- When an external device wants to take control of the system bus, it signals to the 8086 by switching HOLD to the logic 1 level. At the completion of the current bus cycle, the 8086 enters the hold state. In the hold state, signal lines  $AD_0$  through  $AD_{15}$ ,  $A_{16}/S_3$  through  $A_{19}/S_6$ ,  $\overline{BHE}$ ,  $\overline{M}/IO$ ,  $DT/R$ ,  $RD$ ,  $WR$ ,  $DEN$  and  $INTR$  are all in the high Z state. The 8086 signals external device that it is in this state by switching its HLDA output to logic 1 level.

# Maximum Mode Interface

- When the 8086 is set for the maximum-mode configuration, it provides signals for implementing a multiprocessor / coprocessor system environment.
- By multiprocessor environment we mean that one microprocessor exists in the system and that each processor is executing its own program.
- Usually in this type of system environment, there are some system resources that are common to all processors.
- They are called as *global resources*. There are also other resources that are assigned to specific processors. These are known as *local* or *private resources*.

## Maximum Mode Interface (cont..)

- Coprocessor also means that there is a second processor in the system. In this two processor does not access the bus at the same time.
- One passes the control of the system bus to the other and then may suspend its operation.
- In the maximum-mode 8086 system, facilities are provided for implementing allocation of global resources and passing bus control to other microprocessor or coprocessor.



8086 Maximum mode Block Diagram



## Maximum Mode Interface (cont..)

- **8288 Bus Controller – Bus Command and Control Signals:** 8086 does not directly provide all the signals that are required to control the memory, I/O and interrupt interfaces.
- Specially the  $\overline{WR}$ ,  $\overline{M/IO}$ ,  $\overline{DT/R}$ ,  $\overline{DEN}$ ,  $\overline{ALE}$  and  $\overline{INTA}$ , signals are no longer produced by the 8086. Instead it outputs three status signals  $\overline{S_0}$ ,  $\overline{S_1}$ ,  $\overline{S_2}$  prior to the initiation of each bus cycle. This 3- bit bus status code identifies which type of bus cycle is to follow.
- $\overline{S_2}\overline{S_1}\overline{S_0}$  are input to the external bus controller device, the bus controller generates the appropriately timed command and control signals.

## Maximum Mode Interface (cont..)

Status Inputs			CPU Cycles	8288 Command
$\overline{S}_2$	$\overline{S}_1$	$\overline{S}_0$		
0	0	0	Interrupt Acknowledge	$\overline{INTA}$
0	0	1	Read I/O Port	$\overline{IORC}$
0	1	0	Write I/O Port	$\overline{IOWC}, \overline{AIOWC}$
0	1	1	Halt	None
1	0	0	Instruction Fetch	$\overline{MRDC}$
1	0	1	Read Memory	$\overline{MRDC}$
1	1	0	Write Memory	$\overline{MWTC}, \overline{AMWC}$
1	1	1	Passive	None

### Bus Status Codes

## Maximum Mode Interface (cont..)

- The 8288 produces one or two of these eight command signals for each bus cycles. For instance, when the 8086 outputs the code  $S_2S_1S_0$  equals 001, it indicates that an *I/O read cycle* is to be performed.
- In the code 111 is output by the 8086, it is signaling that no bus activity is to take place.
- The control outputs produced by the 8288 are DEN, DT/ $\overline{R}$  and ALE. These 3 signals provide the same functions as those described for the minimum system mode. This set of bus commands and control signals is compatible with the Multibus and industry standard for interfacing microprocessor systems.

## Maximum Mode Interface (cont..)

- **8289 Bus Arbiter – Bus Arbitration and Lock Signals :**  
This device permits processors to reside on the system bus. It does this by implementing the Multibus arbitration protocol in an 8086-based system.
- Addition of the 8288 bus controller and 8289 bus arbiter frees a number of the 8086 pins for use to produce control signals that are needed to support multiple processors.
- Bus priority lock ( $\overline{\text{LOCK}}$ ) is one of these signals. It is input to the bus arbiter together with status signals  $S_0$  through  $S_2$ .



## Maximum Mode Interface (cont..)

- *The output of 8289 are bus arbitration signals: bus busy ( $\overline{\text{BUSY}}$ ), common bus request ( $\overline{\text{CBRQ}}$ ), bus priority out ( $\text{BPRO}$ ), bus priority in ( $\text{BPRN}$ ), bus request ( $\text{BREQ}$ ) and bus clock ( $\overline{\text{BCLK}}$ ).*
- They correspond to the bus exchange signals of the Multibus and are used to lock other processor off the system bus during the execution of an instruction by the 8086.
- In this way the processor can be assured of uninterrupted access to common system resources such as *global memory*.

## Maximum Mode Interface (cont..)

- **Queue Status Signals** : Two new signals that are produced by the 8086 in the maximum-mode system are queue status outputs  $QS_0$  and  $QS_1$ . Together they form a 2-bit queue status code,  $QS_1QS_0$ .
- Following table shows the four different queue status.

## Maximum Mode Interface (cont..)

$QS_1$	$QS_0$	Queue Status
<b>0 (low)</b>	<b>0</b>	<b>No Operation. During the last clock cycle, nothing was taken from the queue.</b>
<b>0</b>	<b>1</b>	<b>First Byte. The byte taken from the queue was the first byte of the instruction.</b>
<b>1 (high)</b>	<b>0</b>	<b>Queue Empty. The queue has been reinitialized as a result of the execution of a transfer instruction.</b>
<b>1</b>	<b>1</b>	<b>Subsequent Byte. The byte taken from the queue was a subsequent byte of the instruction.</b>

Queue status codes

## Maximum Mode Interface (cont..)

- **Local Bus Control Signal – Request / Grant Signals:** In a maximum mode configuration, the minimum mode HOLD, HLDA interface is also changed. These two are replaced by request/grant lines  $\overline{RQ}/\overline{GT}_0$  and  $\overline{RQ}/\overline{GT}_1$ , respectively. They provide a prioritized bus access mechanism for accessing the local bus.



# Minimum Mode 8086 System

- In a minimum mode 8086 system, the microprocessor 8086 is operated in minimum mode by strapping its  $\overline{MN}/\overline{MX}$  pin to logic 1.
- In this mode, all the control signals are given out by the microprocessor chip itself. There is a single microprocessor in the minimum mode system.
- The remaining components in the system are latches, transreceivers, clock generator, memory and I/O devices. Some type of chip selection logic may be required for selecting memory or I/O devices, depending upon the address map of the system.

## Minimum Mode 8086 System (cont..)

- Latches are generally buffered output D-type flip-flops like 74LS373 or 8282. They are used for separating the valid address from the multiplexed address/data signals and are controlled by the ALE signal generated by 8086.
- Transreceivers are the bidirectional buffers and some times they are called as data amplifiers. They are required to separate the valid data from the time multiplexed address/data signals.
- They are controlled by two signals namely,  $\overline{DEN}$  and  $\overline{DT/R}$ .

## Minimum Mode 8086 System (cont..)

- The  $\overline{\text{DEN}}$  signal indicates the direction of data, i.e. from or to the processor. The system contains memory for the monitor and users program storage.
- Usually, EPROM are used for monitor storage, while RAM for users program storage. A system may contain I/O devices.

## Minimum Mode 8086 System (cont..)

- The clock generator generates the clock from the crystal oscillator and then shapes it and divides to make it more precise so that it can be used as an accurate timing reference for the system.
- The clock generator also synchronizes some external signal with the system clock. The general system organisation is as shown in below fig.
- It has 20 address lines and 16 data lines, the 8086 CPU requires three octal address latches and two octal data buffers for the complete address and data separation.



## Minimum Mode 8086 System (cont..)

- The working of the minimum mode configuration system can be better described in terms of the timing diagrams rather than qualitatively describing the operations.
- The opcode fetch and read cycles are similar. Hence the timing diagram can be categorized in two parts, the first is the timing diagram for read cycle and the second is the timing diagram for write cycle.
- The read cycle begins in  $T_1$  with the assertion of address latch enable (ALE) signal and also  $\overline{M}/\overline{IO}$  signal. During the negative going edge of this signal, the valid address is latched on the local bus.

## Minimum Mode 8086 System (cont..)

- The  $\overline{\text{BHE}}$  and  $\overline{\text{A}}_0$  signals address low, high or both bytes. From  $T_1$  to  $T_4$ , the M/IO signal indicates a memory or I/O operation.
- At  $T_2$ , the address is removed from the local bus and is sent to the output. The bus is then tristated. The read ( $\overline{\text{RD}}$ ) control signal is also activated in  $T_2$ .
- The read ( $\overline{\text{RD}}$ ) signal causes the address device to enable its data bus drivers. After  $\overline{\text{RD}}$  goes low, the valid data is available on the data bus.
- The addressed device will drive the READY line high. When the processor returns the read signal to high level, the addressed device will again tristate its bus drivers.

## Minimum Mode 8086 System (cont..)

- A write cycle also begins with the assertion of ALE and the emission of the address. The  $\overline{M/\text{IO}}$  signal is again asserted to indicate a memory or I/O operation. In  $T_2$ , after sending the address in  $T_1$ , the processor sends the data to be written to the addressed location.
- The data remains on the bus until middle of  $T_4$  state. The  $\overline{\text{WR}}$  becomes active at the beginning of  $T_2$  (unlike  $\overline{\text{RD}}$  is somewhat delayed in  $T_2$  to provide time for floating).
- The  $\overline{\text{BHE}}$  and  $A_0$  signals are used to select the proper byte or bytes of memory or I/O word to be read or write.
- The  $\overline{M/\text{IO}}$ ,  $\overline{\text{RD}}$  and  $\overline{\text{WR}}$  signals indicate the type of data transfer as specified in table below.

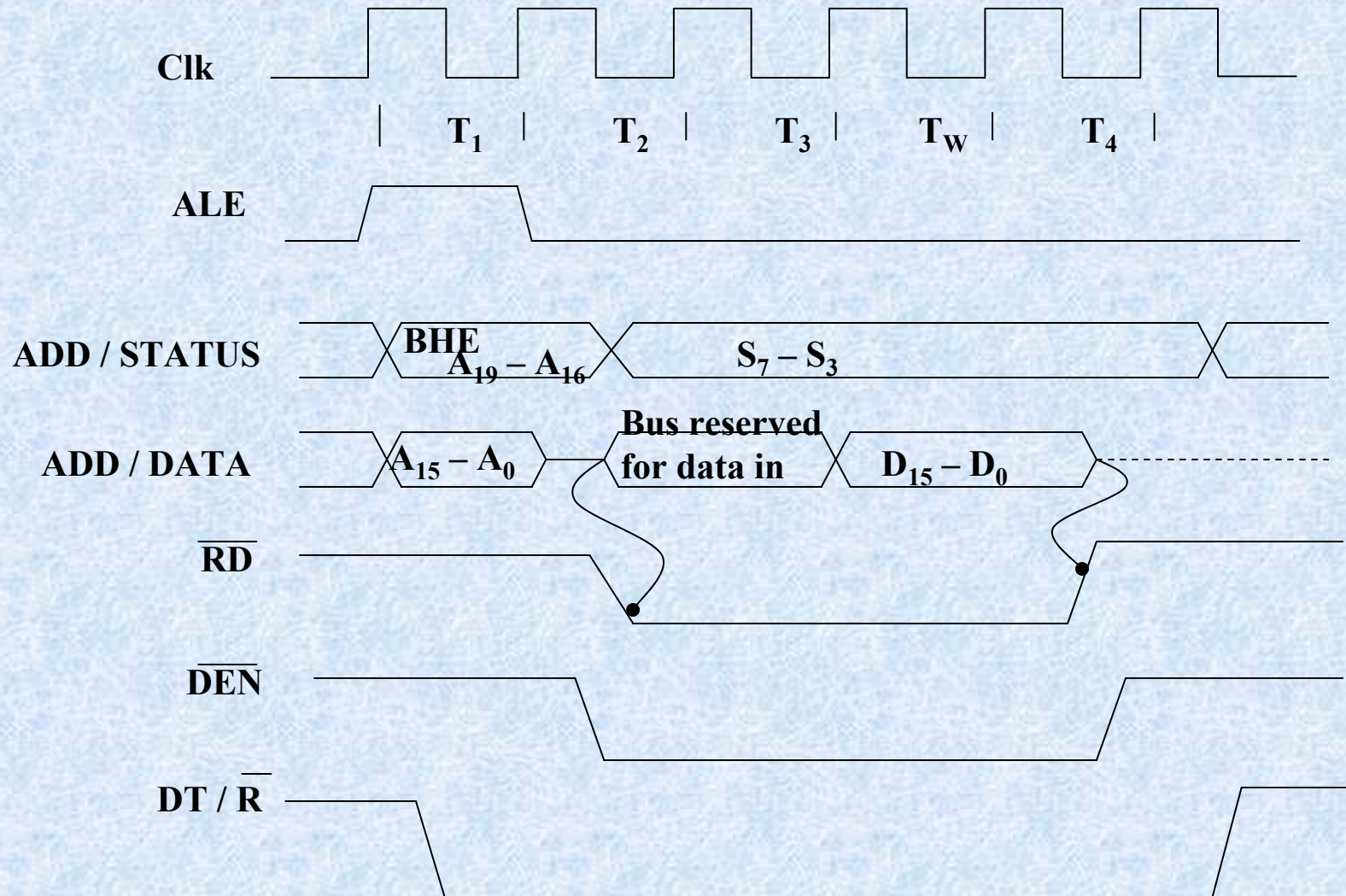
# Minimum Mode 8086 System (cont..)

<b>M / <math>\overline{\text{IO}}</math></b>	<b><math>\overline{\text{RD}}</math></b>	<b><math>\overline{\text{WR}}</math></b>	<b>Transfer Type</b>
<b>0</b>	<b>0</b>	<b>1</b>	<b>I / O read</b>
<b>0</b>	<b>1</b>	<b>0</b>	<b>I/O write</b>
<b>1</b>	<b>0</b>	<b>1</b>	<b>Memory read</b>
<b>1</b>	<b>1</b>	<b>0</b>	<b>Memory write</b>

Data Transfer table

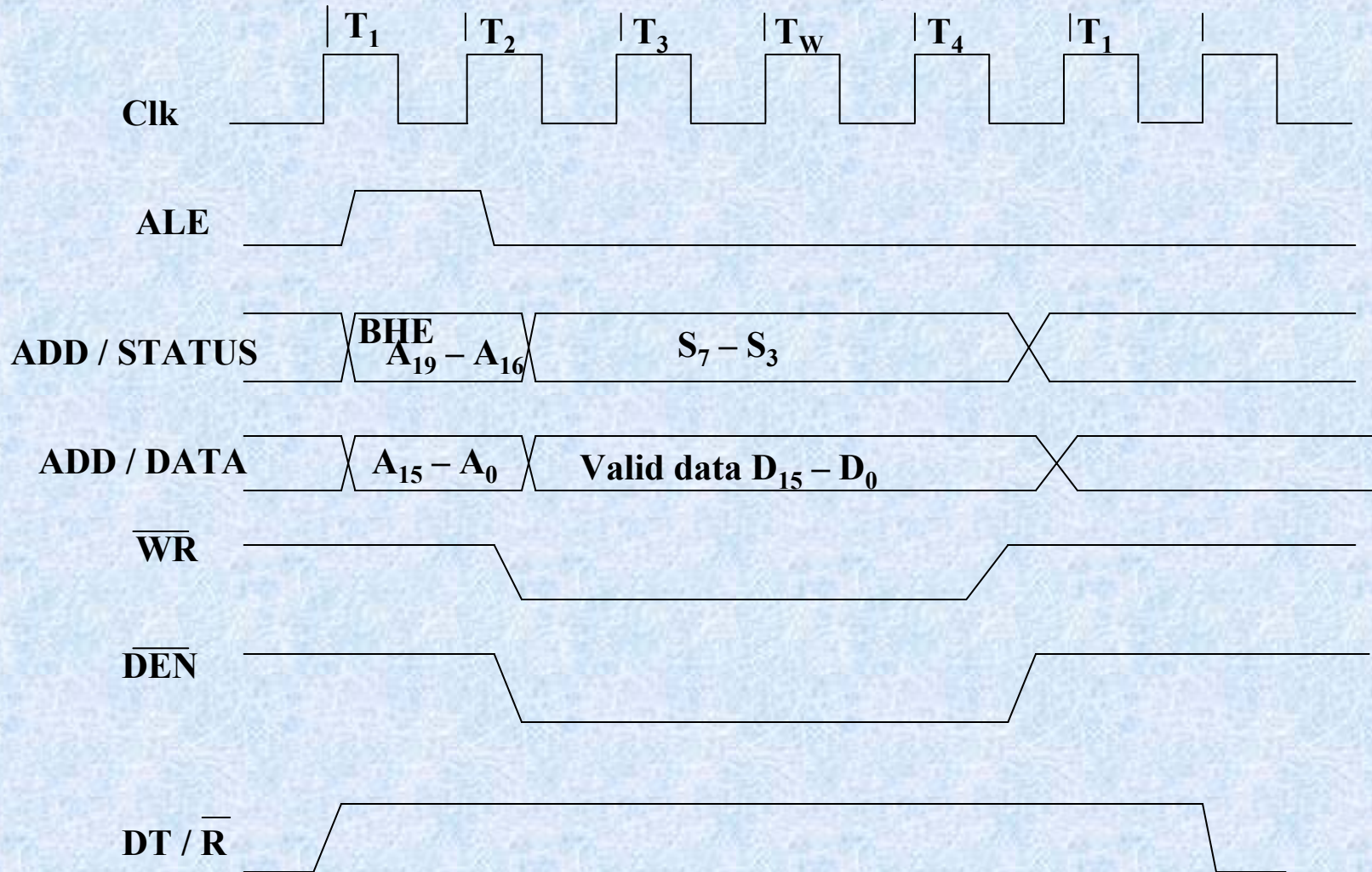


# Minimum Mode 8086 System (cont..)



Read Cycle Timing Diagram for Minimum Mode

# Minimum Mode 8086 System (cont..)

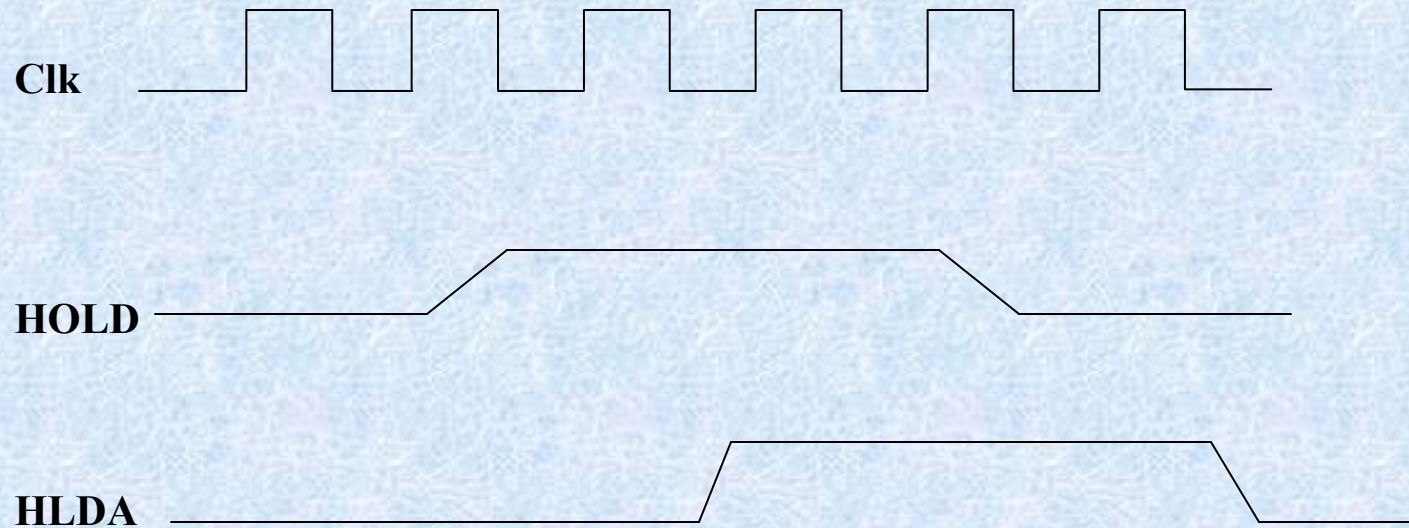


Write Cycle Timing Diagram for Minimum Mode

## Minimum Mode 8086 System (cont..)

- ***Hold Response sequence:*** The HOLD pin is checked at leading edge of each clock pulse. If it is received active by the processor before  $T_4$  of the previous cycle or during  $T_1$  state of the current cycle, the CPU activates HLDA in the next clock cycle and for succeeding bus cycles, the bus will be given to another requesting master.
- The control of the bus is not regained by the processor until the requesting master does not drop the HOLD pin low. When the request is dropped by the requesting master, the HLDA is dropped by the processor at the trailing edge of the next clock.

# Minimum Mode 8086 System (cont..)



Bus Request and Bus Grant Timings in Minimum Mode System



# Maximum Mode 8086 System

- In the maximum mode, the 8086 is operated by strapping the  $\overline{\text{MN}}/\overline{\text{MX}}$  pin to ground.
- In this mode, the processor derives the status signal  $\overline{\text{S}}_2$ ,  $\overline{\text{S}}_1$ ,  $\overline{\text{S}}_0$ . Another chip called bus controller derives the control signal using this status information .
- In the maximum mode, there may be more than one microprocessor in the system configuration.
- The components in the system are same as in the minimum mode system.

## Maximum Mode 8086 System (cont..)

- The basic function of the bus controller chip IC8288, is to derive control signals like RD and WR ( for memory and I/O devices),  $\overline{DEN}$ ,  $DT/\overline{R}$ , ALE etc. using the information by the processor on the status lines.
- The bus controller chip has input lines  $\overline{S}_2$ ,  $\overline{S}_1$ ,  $\overline{S}_0$  and CLK. These inputs to 8288 are driven by CPU.
- It derives the outputs ALE,  $\overline{DEN}$ ,  $DT/\overline{R}$ ,  $\overline{MRDC}$ ,  $\overline{MWTC}$ ,  $\overline{AMWC}$ ,  $\overline{IORC}$ ,  $\overline{IOWC}$  and  $\overline{AIOWC}$ . The  $\overline{AEN}$ , IOB and CEN pins are specially useful for multiprocessor systems.

## Maximum Mode 8086 System ( cont..)

- AEN and  $\overline{\text{IOB}}$  are generally grounded. CEN pin is usually tied to +5V. The significance of the MCE/ $\overline{\text{PDEN}}$  output depends upon the status of the IOB pin.
- If IOB is grounded, it acts as master cascade enable to control cascade 8259A, else it acts as peripheral data enable used in the multiple bus configurations.
- $\overline{\text{INTA}}$  pin used to issue two interrupt acknowledge pulses to the interrupt controller or to an interrupting device.

## Maximum Mode 8086 System ( cont..)

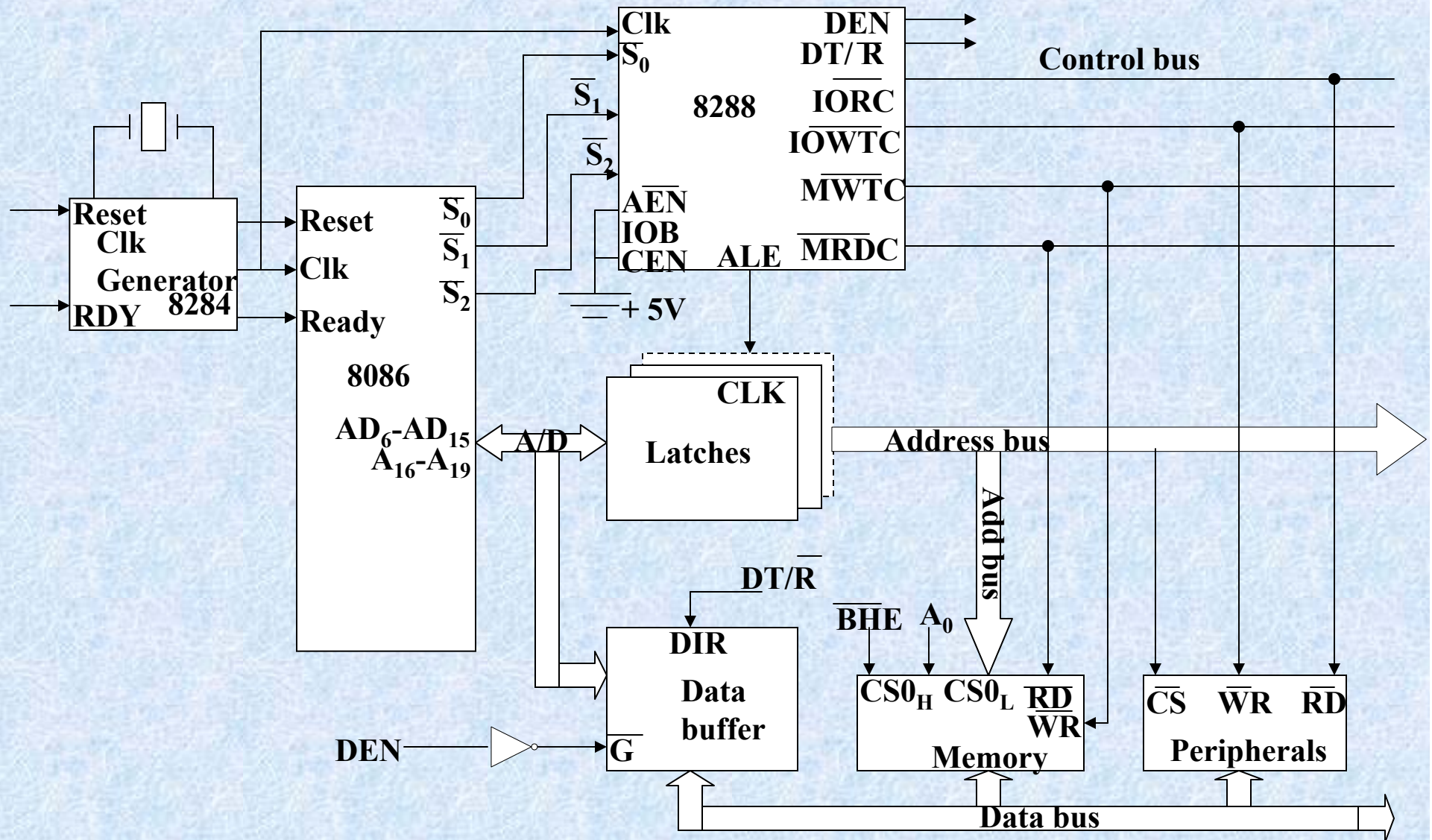
- $\overline{\text{IORC}}$ ,  $\overline{\text{IOWC}}$  are I/O read command and I/O write command signals respectively . These signals enable an IO interface to read or write the data from or to the address port.
- The  $\overline{\text{MRDC}}$ ,  $\overline{\text{MWTC}}$  are memory read command and memory write command signals respectively and may be used as memory read or write signals.
- All these command signals instructs the memory to accept or send data from or to the bus.
- For both of these write command signals, the advanced signals namely  $\overline{\text{AIOWC}}$  and  $\overline{\text{AMWTC}}$  are available.



## Maximum Mode 8086 System ( cont..)

- They also serve the same purpose, but are activated one clock cycle earlier than the  $\overline{\text{IOWC}}$  and  $\overline{\text{MWTC}}$  signals respectively.
- The maximum mode system timing diagrams are divided in two portions as read (input) and write (output) timing diagrams.
- The address/data and address/status timings are similar to the minimum mode.
- ALE is asserted in  $T_1$ , just like minimum mode. The only difference lies in the status signal used and the available control and advanced command signals.

# Maximum Mode 8086 System ( cont..)



Maximum Mode 8086 System.

## Maximum Mode 8086 System ( cont..)

- Here the only difference between in timing diagram between minimum mode and maximum mode is the status signals used and the available control and advanced command signals.
- $\overline{R}_0$ ,  $\overline{S}_1$ ,  $\overline{S}_2$  are set at the beginning of bus cycle. 8288 bus controller will output a pulse as on the ALE and apply a required signal to its DT / R pin during  $T_1$ .

## Maximum Mode 8086 System ( cont..)

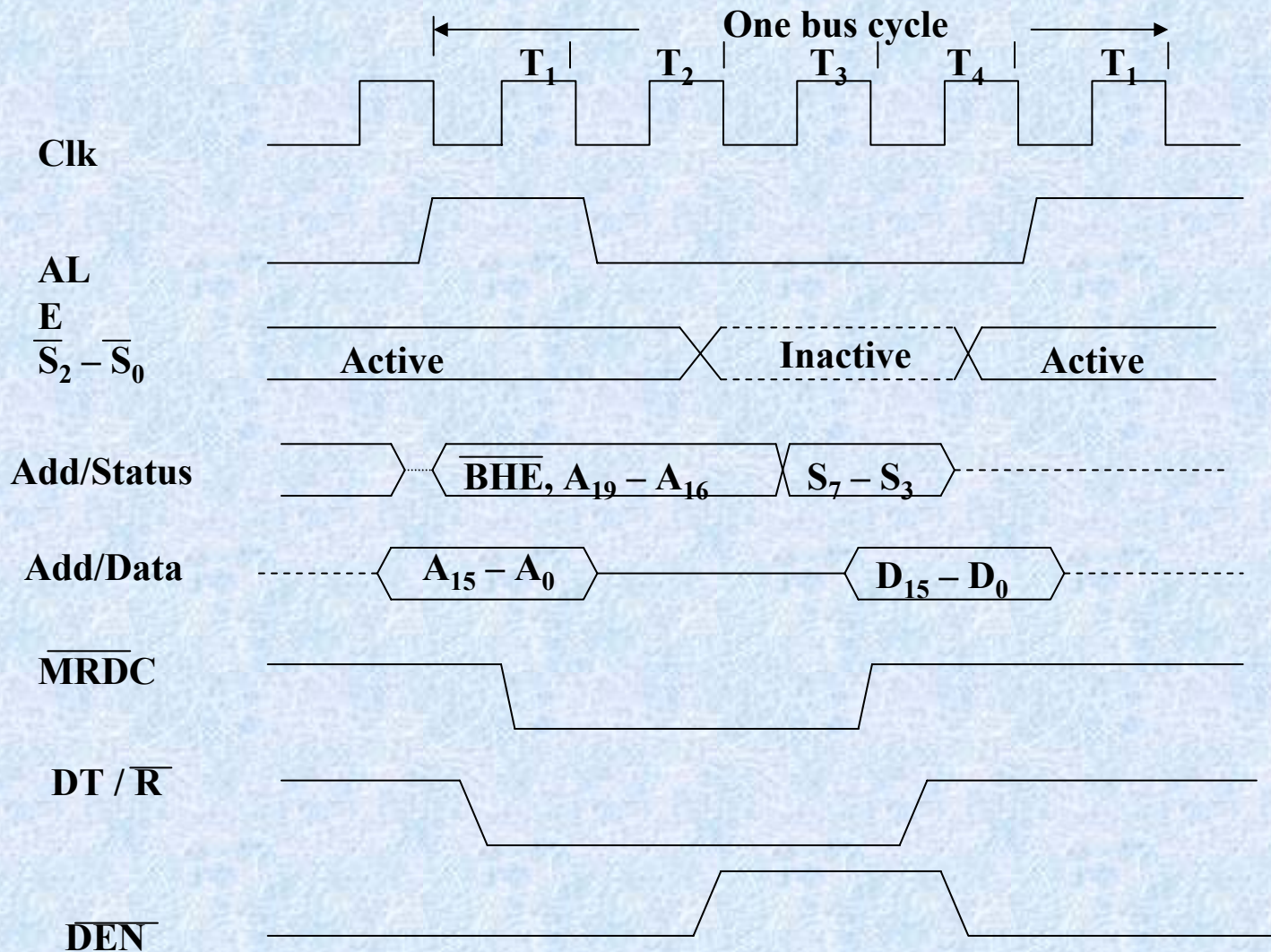
- In  $T_2$ , 8288 will set  $\overline{DEN}=1$  thus enabling transceivers, and for an input it will activate  $\overline{MRDC}$  or  $\overline{IORC}$ . These signals are activated until  $T_4$ . For an output, the  $\overline{AMWC}$  or  $\overline{AIOWC}$  is activated from  $T_2$  to  $T_4$  and  $\overline{MWTC}$  or  $\overline{IOWC}$  is activated from  $T_3$  to  $T_4$ .
- The status bit  $S_0$  to  $S_2$  remains active until  $T_3$  and become passive during  $T_3$  and  $T_4$ .
- If reader input is not activated before  $T_3$ , wait state will be inserted between  $T_3$  and  $T_4$ .



## Maximum Mode 8086 System ( cont..)

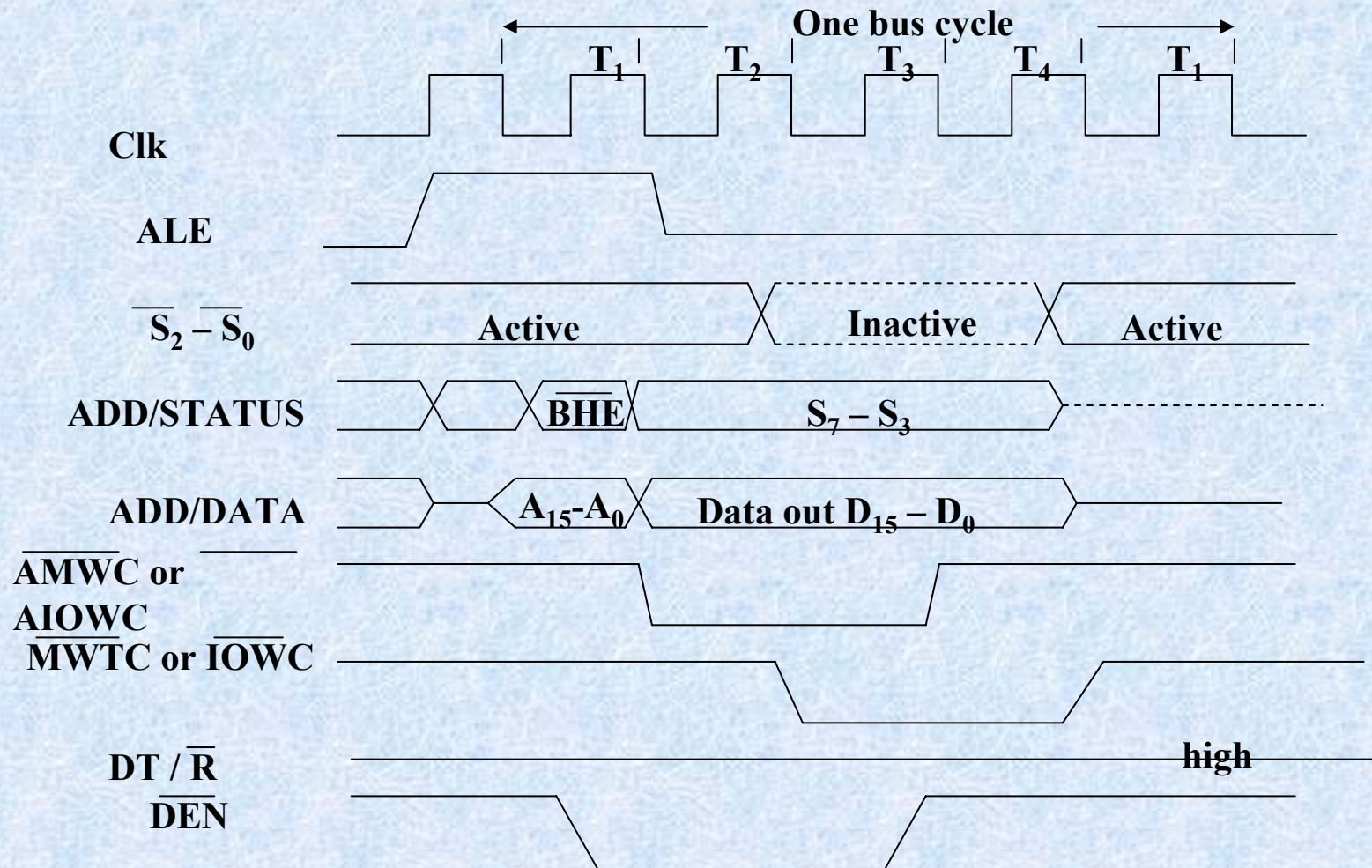
- **Timings for  $\overline{RQ}/\overline{GT}$  Signals** :The request/grant response sequence contains a series of three pulses. The request/grant pins are checked at each rising pulse of clock input.
- When a request is detected and if the condition for HOLD request are satisfied, the processor issues a grant pulse over the  $\overline{RQ}/\overline{GT}$  pin immediately during  $T_4$  (current) or  $T_1$  (next) state.
- When the requesting master receives this pulse, it accepts the control of the bus, it sends a release pulse to the processor using  $\overline{RQ}/\overline{GT}$  pin.

# Maximum Mode 8086 System ( cont..)



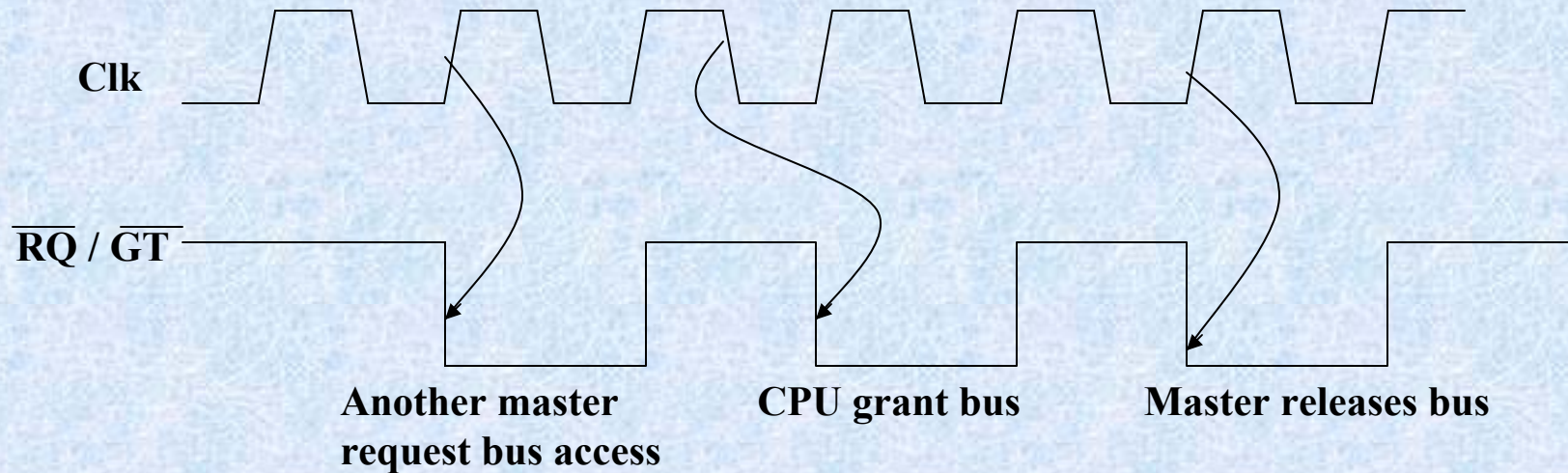
## Memory Read Timing in Maximum Mode

# Maximum Mode 8086 System ( cont..)



Memory Write Timing in Maximum mode.

# Maximum Mode 8086 System ( cont..)



**$\overline{RQ} / \overline{GT}$  Timings in Maximum Mode.**