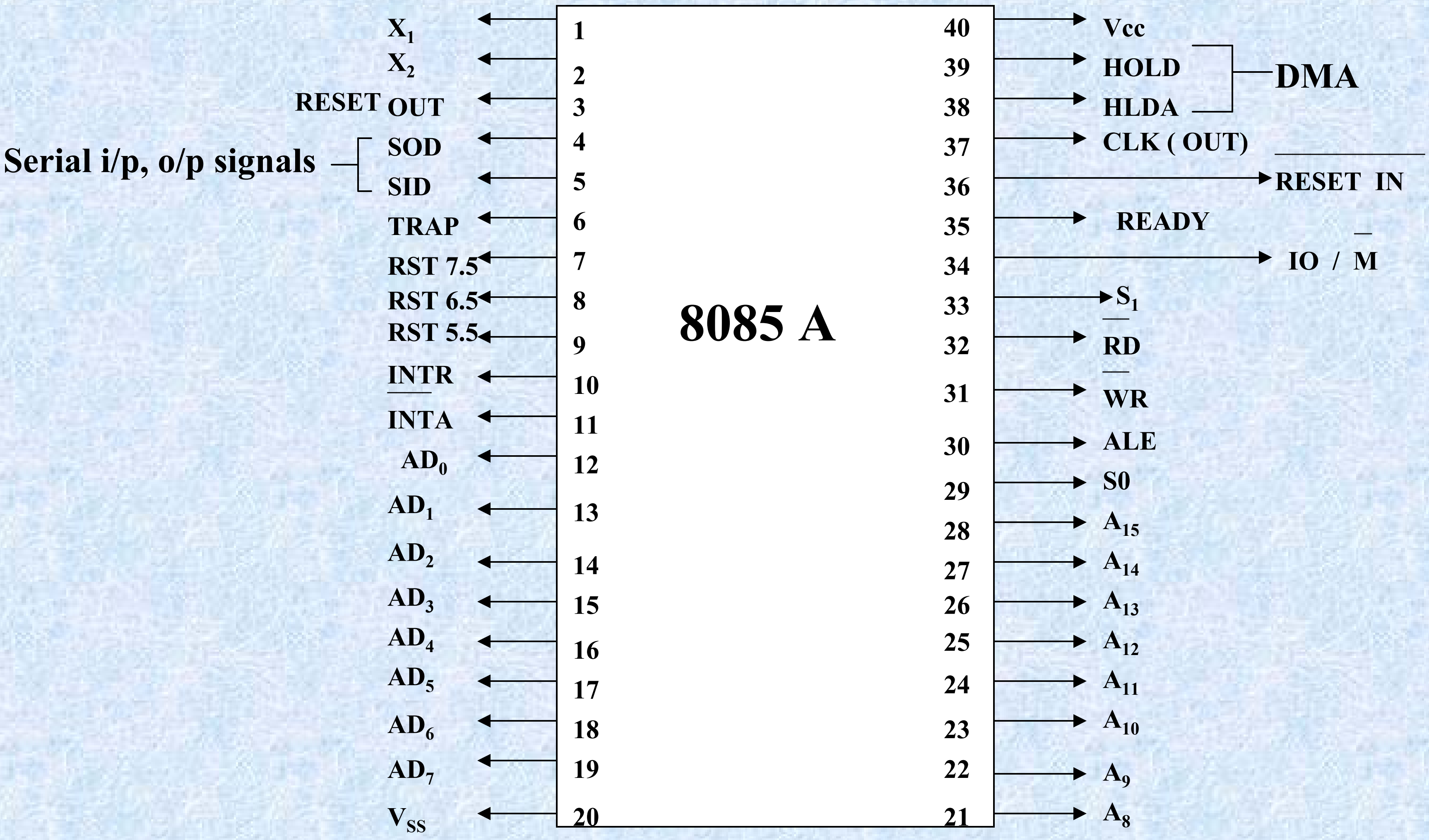
Pin Diagram of 8085



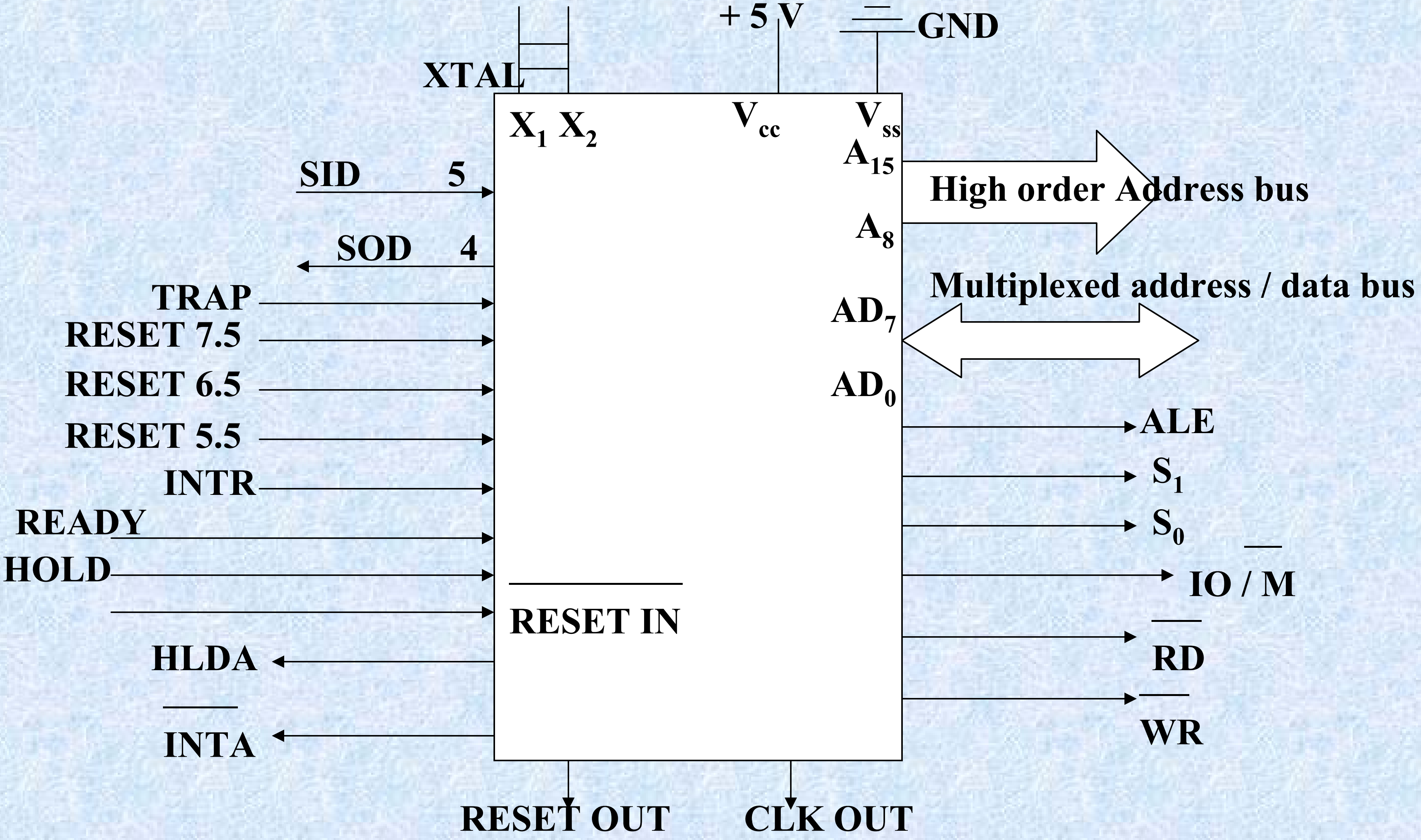
8085 A

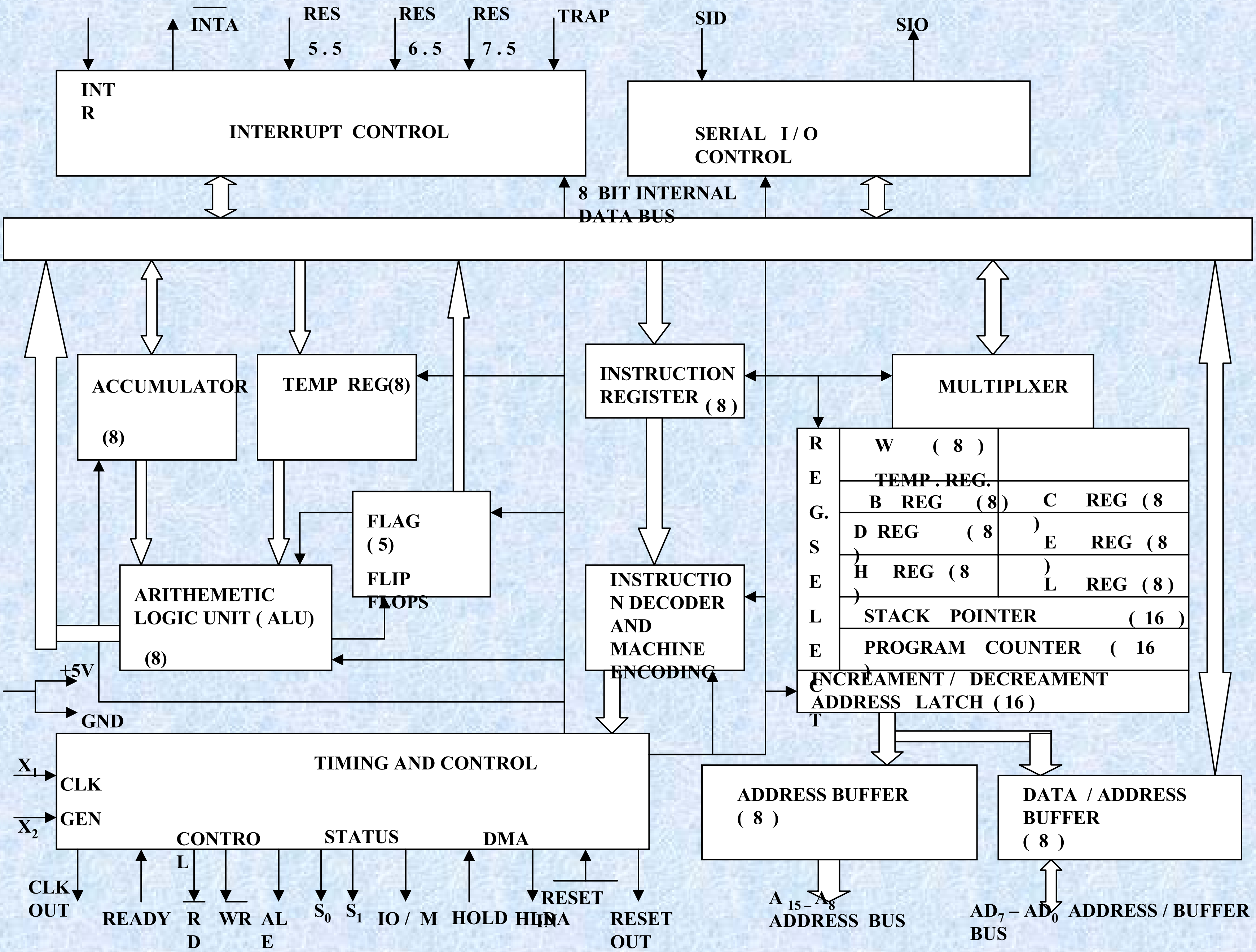
Serial i/p, o/p signals

DMA

X_1	1	40	Vcc
X_2	2	39	HOLD
RESET OUT	3	38	HLDA
SOD	4	37	CLK (OUT)
SID	5	36	$\overline{\text{RESET IN}}$
TRAP	6	35	READY
RST 7.5	7	34	IO / $\overline{\text{M}}$
RST 6.5	8	33	S_1
RST 5.5	9	32	$\overline{\text{RD}}$
INTR	10	31	$\overline{\text{WR}}$
$\overline{\text{INTA}}$	11	30	ALE
AD_0	12	29	S0
AD_1	13	28	A_{15}
AD_2	14	27	A_{14}
AD_3	15	26	A_{13}
AD_4	16	25	A_{12}
AD_5	17	24	A_{11}
AD_6	18	23	A_{10}
AD_7	19	22	A_9
V_{SS}	20	21	A_8

Signal Groups of 8085





INTA

RES 5.5

RES 6.5

RES 7.5

TRAP

SID

SIO

INTR

INTERRUPT CONTROL

SERIAL I / O CONTROL

8 BIT INTERNAL DATA BUS

ACCUMULATOR (8)

TEMP REG(8)

FLAG (5) FLIP FLOPS

INSTRUCTION REGISTER (8)

MULTIPLXER

ARITHEMETIC LOGIC UNIT (ALU) (8)

INSTRUCTION DECODER AND MACHINE ENCODING

REG. SELECT

W (8) TEMP . REG.

B REG (8)

C REG (8)

D REG (8)

E REG (8)

H REG (8)

L REG (8)

STACK POINTER (16)

PROGRAM COUNTER (16)

INCREAMENT / DECREAMENT ADDRESS LATCH (16)

+5V

GND

X1

X2

CLK GEN

TIMING AND CONTROL

CONTROL

STATUS

DMA

CLK OUT

READY

RD

WR

ALE

S_0

S_1

IO / M

HOLD

HLDA

RESET IN

RESET OUT

ADDRESS BUFFER (8)

DATA / ADDRESS BUFFER (8)

$A_{15} – A_8$ ADDRESS BUS

$AD_7 – AD_0$ ADDRESS / BUFFER BUS

Flag Registers

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S	Z		AC		P		CY

General Purpose Registers

INDIVIDUAL	B, C, D, E, H, L
COMBININATON	B & C, D & E, H & L

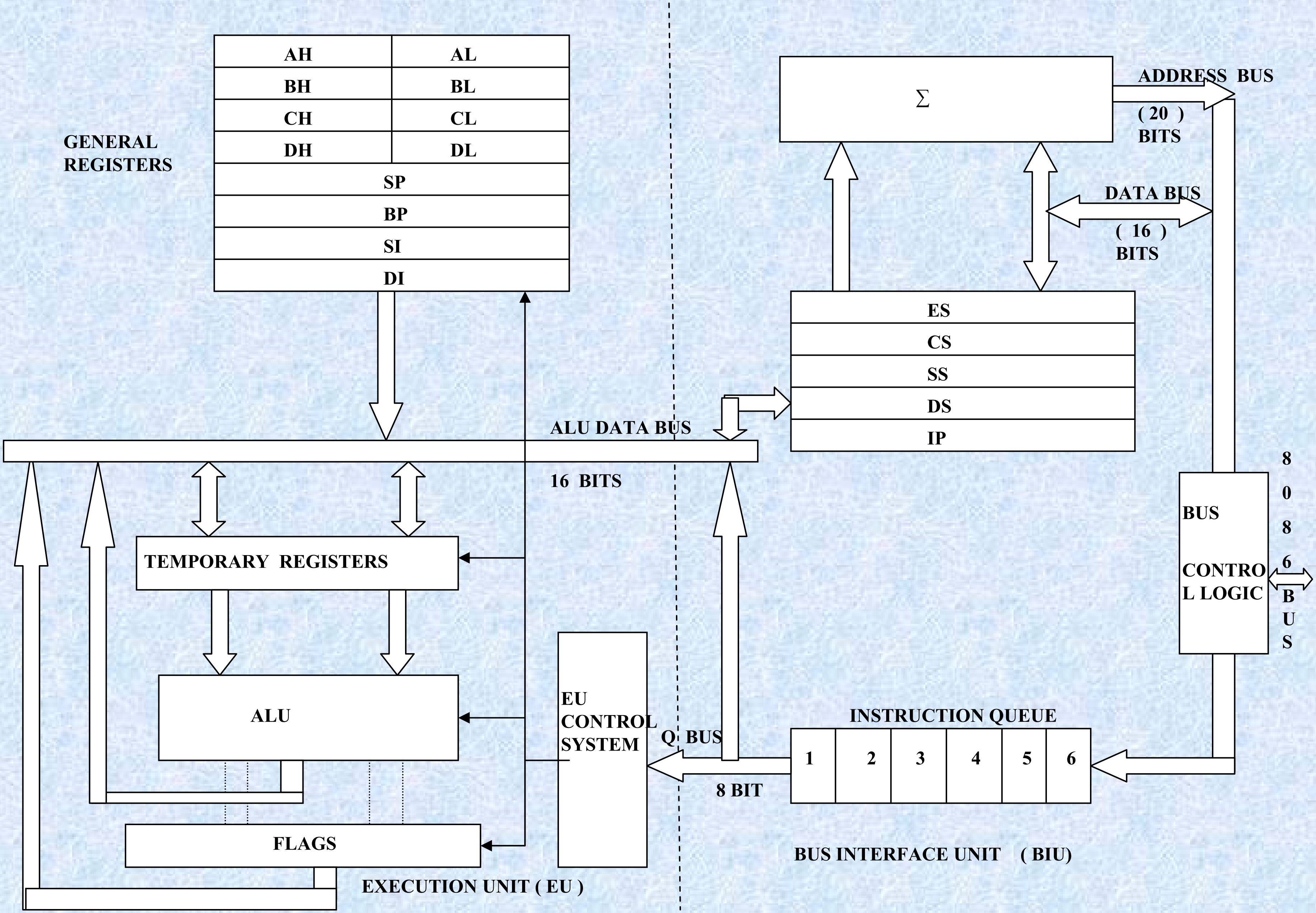
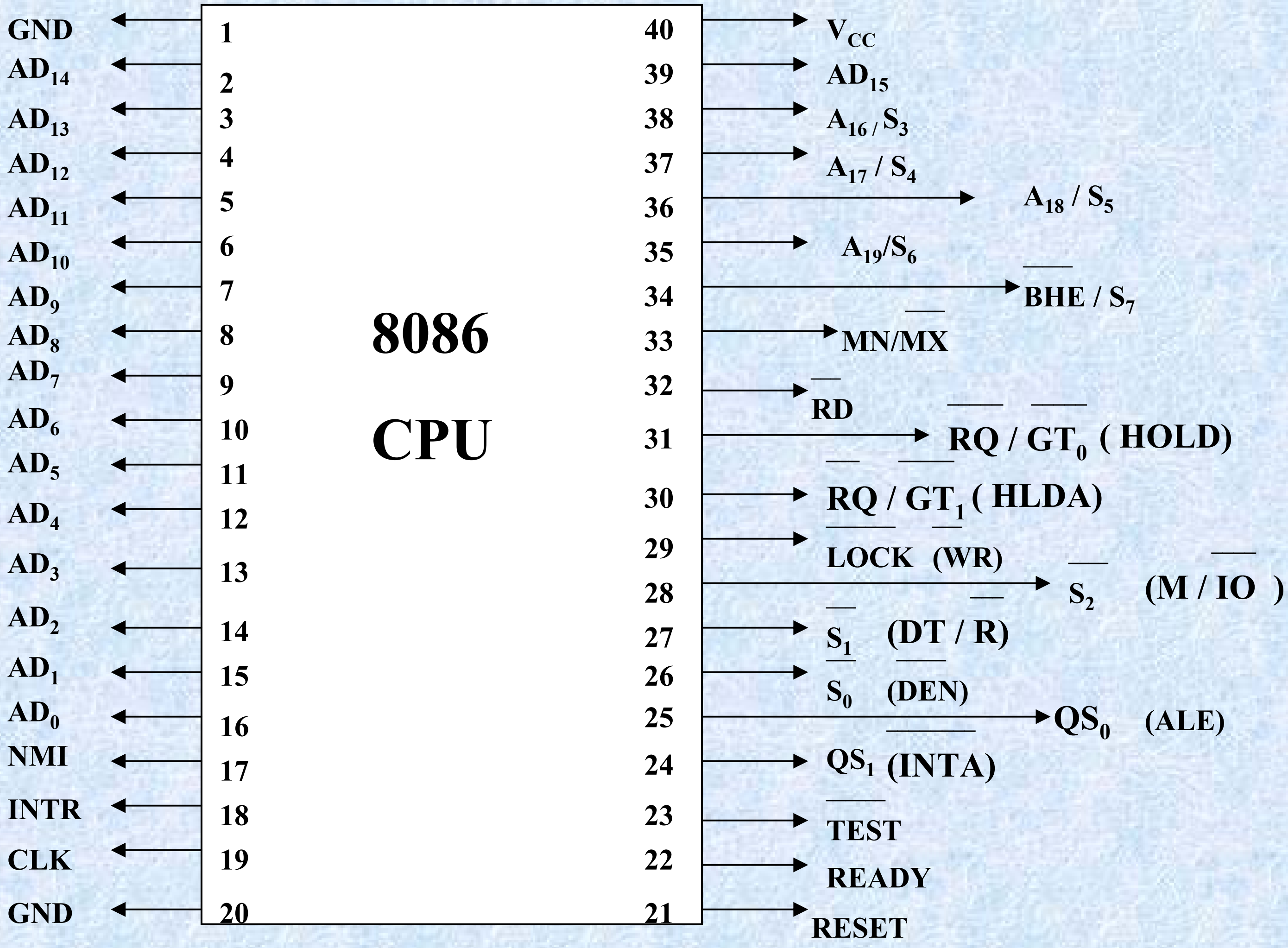
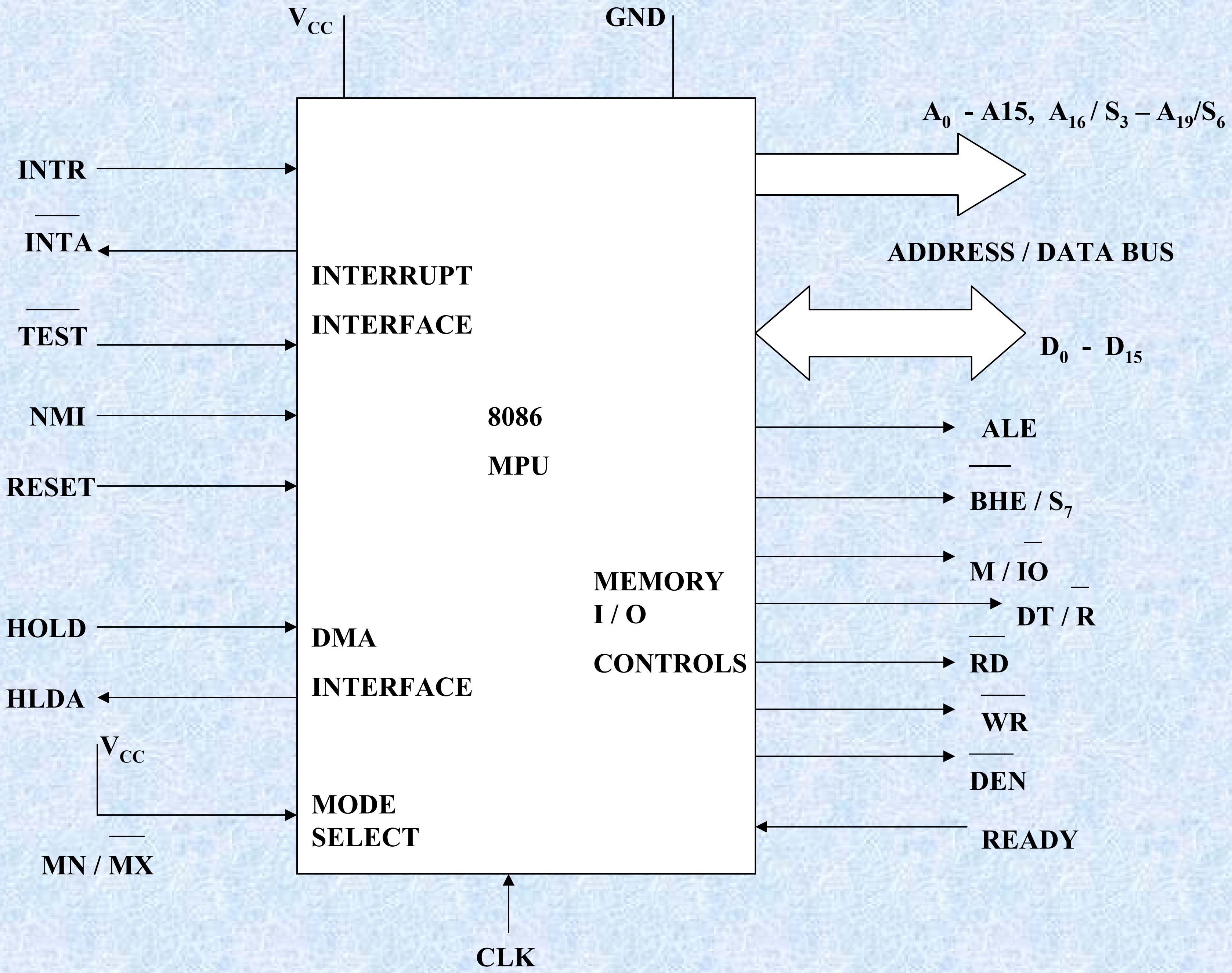

GENERAL REGISTERS
AH
AL
BH
BL
CH
CL
DH
DL
SP
BP
SI
DI
Σ
ADDRESS BUS
(20) BITS
DATA BUS
(16) BITS
ES
CS
SS
DS
IP
ALU DATA BUS
16 BITS
TEMPORARY REGISTERS
ALU
FLAGS
EU CONTROL SYSTEM
Q BUS
8 BIT
INSTRUCTION QUEUE
1
2
3
4
5
6
BUS CONTROL LOGIC
8086 BUS
BUS INTERFACE UNIT (BIU)
EXECUTION UNIT (EU)

Fig:

Pin Diagram of 8086




VCC
GND
A0 - A15, A16 / S3 – A19/S6
INTR
INTA
INTERRUPT
INTERFACE
ADDRESS / DATA BUS
TEST
D0 - D15
NMI
8086
MPU
ALE
RESET
BHE / S7
M / IO
MEMORY
I / O
DT / R
HOLD
DMA
CONTROLS
RD
HLDA
INTERFACE
WR
VCC
DEN
MODE
SELECT
READY
MN / MX
CLK

Signal Description of 8086

- The Microprocessor 8086 is a 16-bit CPU available in different clock rates and packaged in a 40 pin CERDIP or plastic package.
- The 8086 operates in single processor or multiprocessor configuration to achieve high performance. The pins serve a particular function in minimum mode (single processor mode) and other function in maximum mode configuration (multiprocessor mode).
- The 8086 signals can be categorised in three groups. The first are the signal having common functions in minimum as well as maximum mode.

Signal Description of 8086 (cont..)

- The second are the signals which have special functions for minimum mode and third are the signals having special functions for maximum mode.
- The following signal descriptions are common for both modes.
- **AD_{15}-AD_0** : These are the time multiplexed memory I/O address and data lines.
- Address remains on the lines during T_1 state, while the data is available on the data bus during T_2, T_3, T_w and T_4.
- These lines are active high and float to a tristate during interrupt acknowledge and local bus hold acknowledge cycles.

Signal Description of 8086 (cont..)

- **$A_{19}/S_6, A_{18}/S_5, A_{17}/S_4, A_{16}/S_3$** : These are the time multiplexed address and status lines.
- During T_1 these are the most significant address lines for memory operations.
- During I/O operations, these lines are low. During memory or I/O operations, status information is available on those lines for T_2, T_3, T_w and T_4.
- The status of the interrupt enable flag bit is updated at the beginning of each clock cycle.

Signal Description of 8086 (cont..)

- The S_4 and S_3 combinedly indicate which segment register is presently being used for memory accesses as in below fig.
- These lines float to tri-state off during the local bus hold acknowledge. The status line S_6 is always low .
- The address bit are separated from the status bit using latches controlled by the ALE signal.

S_4	S_3	Indication
0	0	Alternate Data
0	1	Stack
1	0	Code or none
1	1	Data

Signal Description of 8086 (cont..)

- **$\overline{\text{BHE}}$/S$_7$** : The bus high enable is used to indicate the transfer of data over the higher order (D_{15}-D_8) data bus as shown in table. It goes low for the data transfer over D_{15}-D_8 and is used to derive chip selects of odd address memory bank or peripherals. BHE is low during T_1 for read, write and interrupt acknowledge cycles, whenever a byte is to be transferred on higher byte of data bus. The status information is available during T_2, T_3 and T_4. The signal is active low and tristated during hold. It is low during T_1 for the first pulse of the interrupt acknowledge cycle.

BHE	A_0	Indication
0	0	Whole word
0	1	Upper byte from or to even address
1	0	Lower byte from or to even address
1	1	None

Signal Description of 8086 (cont..)

- **$\overline{\mathbf{RD}}$ – Read** : This signal on low indicates the peripheral that the processor is performing s memory or I/O read operation. RD is active low and shows the state for T_2, T_3, T_w of any read cycle. The signal remains tristated during the hold acknowledge.
- **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 high.

Signal Description of 8086 (cont..)

- **INTR-Interrupt Request** : This is a triggered input. This is sampled during the last clock cycles of each instruction to determine the availability of the request. If any interrupt request is pending, the processor enters the interrupt acknowledge cycle.
- This can be internally masked by resulting the interrupt enable flag. This signal is active high and internally synchronized.
- $\overline{\textbf{TEST}}$: This input is examined by a ‘WAIT’ instruction. If the TEST pin goes low, execution will continue, else the processor remains in an idle state. The input is synchronized internally during each clock cycle on leading edge of clock.

Signal Description of 8086 (cont..)

- **NMI- Nonmaskable interrupt** : This is an edge triggered input which causes a Type 2 interrupt. The NMI is not maskable internally by software. A transition from low to high initiates the interrupt response at the end of the current instruction. This input is internally synchronized.
- **RESET** : This input causes the processor to terminate the current activity and start execution from FFF0H. The signal is active high and must be active for at least four clock cycles. It restarts execution when the RESET returns low. RESET is also internally synchronized.
- **Vcc** +5V power supply for the operation of the internal circuit.
- **GND** ground for internal circuit.

Signal Description of 8086 (cont..)

- **CLK-** Clock Input : The clock input provides the basic timing for processor operation and bus control activity. Its an asymmetric square wave with 33% duty cycle.
- **MN/$\overline{\textbf{MX}}$** : The logic level at this pin decides whether the processor is to operate in either minimum or maximum mode.
- *The following pin functions are for the **minimum mode** operation of 8086.*
- **M/$\overline{\textbf{IO}}$ – Memory/IO** : This is a status line logically equivalent to S_2 in maximum mode. When it is low, it indicates the CPU is having an I/O operation, and when it is high, it indicates that the CPU is having a memory operation. This line becomes active high in the previous T_4 and remains active till final T_4 of the current cycle. It is tristated during local bus “hold acknowledge “.

Signal Description of 8086 (cont..)

- **$\overline{\text{INTA}}$ – Interrupt Acknowledge** : This signal is used as a read strobe for interrupt acknowledge cycles. i.e. when it goes low, the processor has accepted the interrupt.
- **ALE – Address Latch Enable** : This output signal indicates the availability of the valid address on the address/data lines, and is connected to latch enable input of latches. This signal is active high and is never tristated.
- **DT/$\overline{\text{R}}$ – Data Transmit/Receive**: This output is used to decide the direction of data flow through the transreceivers (bidirectional buffers). When the processor sends out data, this signal is high and when the processor is receiving data, this signal is low.

Signal Description of 8086 (cont..)

- **DEN – Data Enable** : This signal indicates the availability of valid data over the address/data lines. It is used to enable the transreceivers (bidirectional buffers) to separate the data from the multiplexed address/data signal. It is active from the middle of T_2 until the middle of T_4. This is tristated during ‘ hold acknowledge’ cycle.
- **HOLD, HLDA- Acknowledge** : When the HOLD line goes high, it indicates to the processor that another master is requesting the bus access.
- The processor, after receiving the HOLD request, issues the hold acknowledge signal on HLDA pin, in the middle of the next clock cycle after completing the current bus cycle.

Signal Description of 8086 (cont..)

- At the same time, the processor floats the local bus and control lines. When the processor detects the HOLD line low, it lowers the HLDA signal. HOLD is an asynchronous input, and is should be externally synchronized.
- If the DMA request is made while the CPU is performing a memory or I/O cycle, it will release the local bus during T_4 provided :

1. The request occurs on or before T_2 state of the current cycle.
2. The current cycle is not operating over the lower byte of a word.
3. The current cycle is not the first acknowledge of an interrupt acknowledge sequence.

Signal Description of 8086 (cont..)

4. A Lock instruction is not being executed.

- ***The following pin function are applicable for maximum mode operation of 8086***.
- **$\overline{S}_2$, $\overline{S}_1$, $\overline{S}_0$ – Status Lines** : These are the status lines which reflect the type of operation, being carried out by the processor. These become activity during T_4 of the previous cycle and active during T_1 and T_2 of the current bus cycles.

S_2	S_1	S_0	Indication
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

Signal Description of 8086 (cont..)

- $\overline{\textbf{LOCK}}$: This output pin indicates that other system bus master will be prevented from gaining the system bus, while the $\overline{\text{LOCK}}$ signal is low.
- The $\overline{\text{LOCK}}$ signal is activated by the ‘LOCK’ prefix instruction and remains active until the completion of the next instruction. When the CPU is executing a critical instruction which requires the system bus, the LOCK prefix instruction ensures that other processors connected in the system will not gain the control of the bus.
- The 8086, while executing the prefixed instruction, asserts the bus lock signal output, which may be connected to an external bus controller.

Signal Description of 8086 (cont..)

- **QS_1, QS_0 – Queue Status**: These lines give information about the status of the code-prefetch queue. These are active during the CLK cycle after while the queue operation is performed.
- This modification in a simple fetch and execute architecture of a conventional microprocessor offers an added advantage of pipelined processing of the instructions.
- The 8086 architecture has 6-byte instruction prefetch queue. Thus even the largest (6-bytes) instruction can be prefetched from the memory and stored in the prefetch. This results in a faster execution of the instructions.
- In 8085 an instruction is fetched, decoded and executed and only after the execution of this instruction, the next one is fetched.

Signal Description of 8086 (cont..)

- By prefetching the instruction, there is a considerable speeding up in instruction execution in 8086. This is known as ***instruction pipelining***.
- At the starting the CS:IP is loaded with the required address from which the execution is to be started. Initially, the queue will be empty an the microprocessor starts a fetch operation to bring one byte (the first byte) of instruction code, if the CS:IP address is odd or two bytes at a time, if the CS:IP address is even.
- The first byte is a complete opcode in case of some instruction (one byte opcode instruction) and is a part of opcode, in case of some instructions (two byte opcode instructions), the remaining part of code lie in second byte.

Signal Description of 8086 (cont..)

- But the first byte of an instruction is an opcode. When the first byte from the queue goes for decoding and interpretation, one byte in the queue becomes empty and subsequently the queue is updated.
- The microprocessor does not perform the next fetch operation till at least two bytes of instruction queue are emptied. The instruction execution cycle is never broken for fetch operation. After decoding the first byte, the decoding circuit decides whether the instruction is of single opcode byte or double opcode byte.
- If it is single opcode byte, the next bytes are treated as data bytes depending upon the decoded instruction length, otherwise, the next byte in the queue is treated as the second byte of the instruction opcode.

Signal Description of 8086 (cont..)

- The second byte is then decoded in continuation with the first byte to decide the instruction length and the number of subsequent bytes to be treated as instruction data.
- The queue is updated after every byte is read from the queue but the fetch cycle is initiated by BIU only if at least two bytes of the queue are empty and the EU may be concurrently executing the fetched instructions.
- The next byte after the instruction is completed is again the first opcode byte of the next instruction. A similar procedure is repeated till the complete execution of the program.

Signal Description of 8086 (cont..)

- The fetch operation of the next instruction is overlapped with the execution of the current instruction. As in the architecture, there are two separate units, namely Execution unit and Bus interface unit.
- While the execution unit is busy in executing an instruction, after it is completely decoded, the bus interface unit may be fetching the bytes of the next instruction from memory, depending upon the queue status.

QS_1	QS_0	Indication
0	0	No operation
0	1	First byte of the opcode from the queue
1	0	Empty queue
1	1	Subsequent byte from the queue

Signal Description of 8086 (cont..)

- **$\overline{RQ}/\overline{GT}_0$, $\overline{RQ}/\overline{GT}_1$ – Request/Grant :** These pins are used by the other local bus master in maximum mode, to force the processor to release the local bus at the end of the processor current bus cycle.
- Each of the pin is bidirectional with $\overline{RQ}/\overline{GT}_0$ having higher priority than $\overline{RQ}/\overline{GT}_1$.
- $\overline{RQ}/\overline{GT}$ pins have internal pull-up resistors and may be left unconnected.
- Request/Grant sequence is as follows:

1. A pulse of one clock wide from another bus master requests the bus access to 8086.

Signal Description of 8086 (cont..)

2. During T_4(current) or T_1(next) clock cycle, a pulse one clock wide from 8086 to the requesting master, indicates that the 8086 has allowed the local bus to float and that it will enter the ‘hold acknowledge’ state at next cycle. The CPU bus interface unit is likely to be disconnected from the local bus of the system.
3. A one clock wide pulse from the another master indicates to the 8086 that the hold request is about to end and the 8086 may regain control of the local bus at the next clock cycle. Thus each master to master exchange of the local bus is a sequence of 3 pulses. There must be at least one dead clock cycle after each bus exchange.

Signal Description of 8086

- The request and grant pulses are active low.
- For the bus request those are received while 8086 is performing memory or I/O cycle, the granting of the bus is governed by the rules as in case of HOLD and HLDA in minimum mode.

General Bus Operation

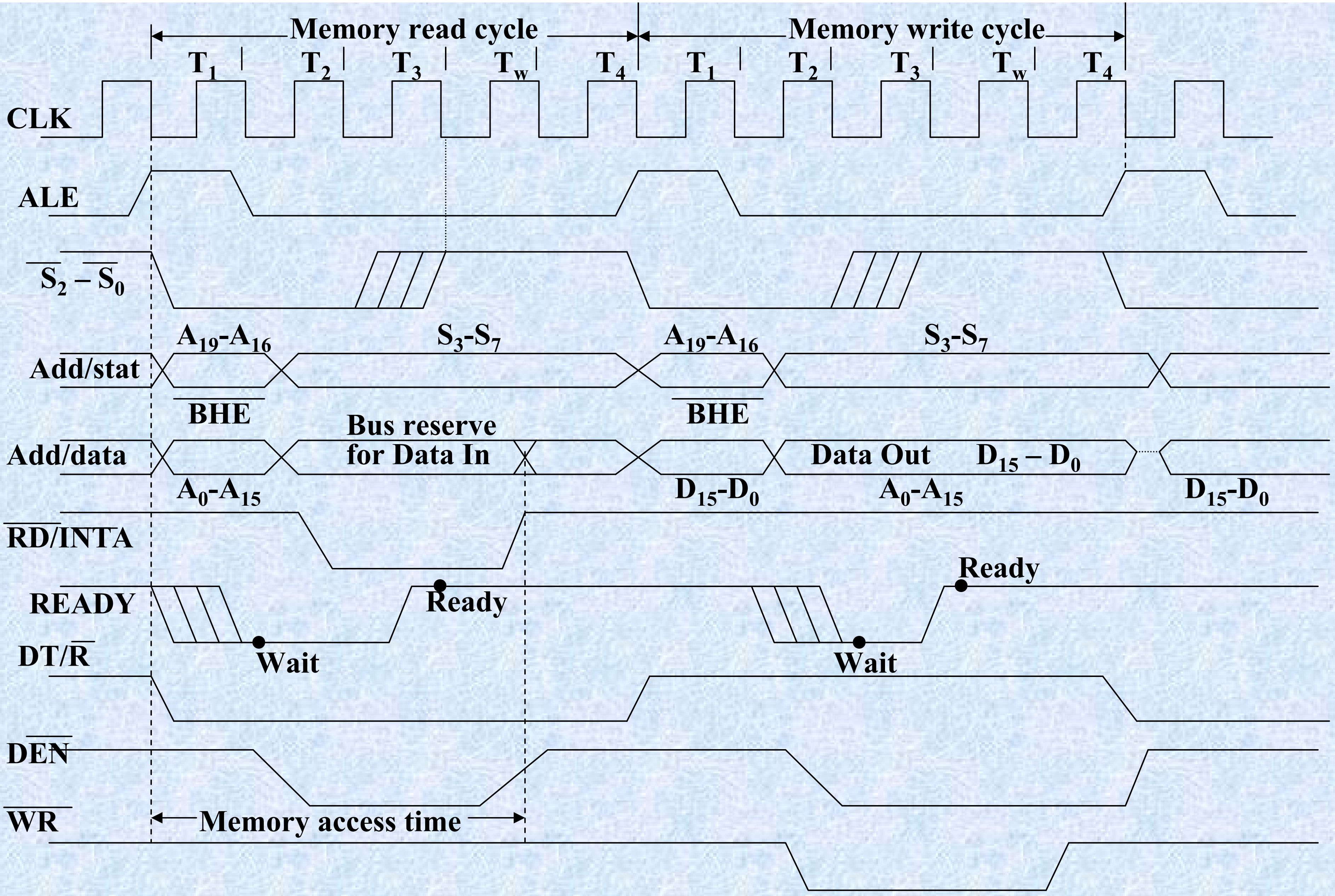
- The 8086 has a combined address and data bus commonly referred as a time multiplexed address and data bus.
- The main reason behind multiplexing address and data over the same pins is the maximum utilisation of processor pins and it facilitates the use of 40 pin standard DIP package.
- The bus can be demultiplexed using a few latches and transreceivers, when ever required.
- Basically, all the processor bus cycles consist of at least four clock cycles. These are referred to as T_1, T_2, T_3, T_4. The address is transmitted by the processor during T_1. It is present on the bus only for one cycle.

General Bus Operation (cont..)

- During T_2, i.e. the next cycle, the bus is tristated for changing the direction of bus for the following data read cycle. The data transfer takes place during T_3, T_4.
- In case, an address device is slow ‘NOT READY’ status the wait status T_w are inserted between T_3 and T_4. These clock states during wait period are called ***idle states*** (T_i), ***wait states*** (T_w) or ***inactive states***. The processor used these cycles for internal housekeeping.
- The address latch enable (ALE) signal is emitted during T_1 by the processor (minimum mode) or the bus controller (maximum mode) depending upon the status of the $MN/\overline{MX}$ input.

General Bus Operation (cont..)

- The negative edge of this ALE pulse is used to separate the address and the data or status information. In maximum mode, the status lines $\overline{S}_0$, $\overline{S}_1$ and $\overline{S}_2$ are used to indicate the type of operation.
- Status bits $\overline{S}_3$ to $\overline{S}_7$ are multiplexed with higher order address bits and the BHE signal. Address is valid during T_1 while status bits S_3 to S_7 are valid during T_2 through T_4.



General Bus Operation Cycle in Maximum Mode