

Unit+1

IC Stands for Integrated Circuit fabricated by using active and passive Semiconductor materials.

Types of IC's

Initially IC's are two types

1. Linear IC's → It is Static in nature

2. Digital IC's → The Digital IC is programmable in nature.

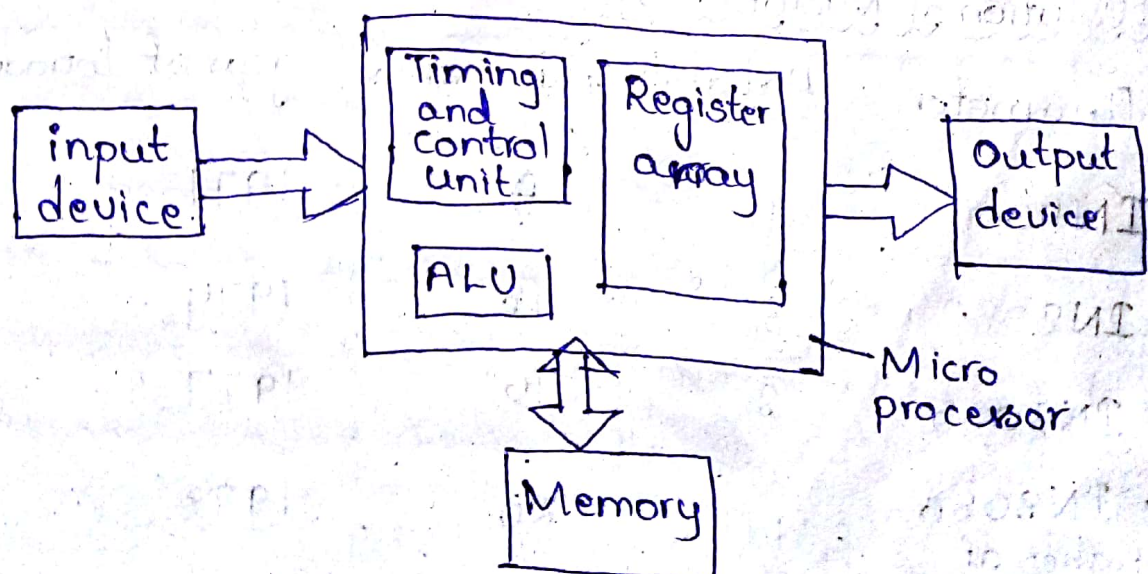
Example of Digital IC is Microprocessor

Example of linear IC is BC-547 (Transistor)

Micro Introduction to Microprocessor

A Microprocessor is a programmable Semiconductor Single chip. It acts like a CPU. Microprocessor is fabricated by using millions of gates includes resistors, capacitors and transistors. It is the heart of the micro Computer.

Block diagram of microprocessor:



Important features of Microprocessor

1. Low Cost - Because of Ic technology the cost is low
2. Low power - Due to the Usage of Metal Oxide Semiconductor its power is low
3. Highly reliable - Because of Semiconductor it takes less time for Switching
4. High Speed - Due to the technology microprocessor executes millions of instruction per Second.

Working Generally microprocessor carries digital data to and from the input to output. All functions of microprocessor are work together in Serial Order.

- 1) fetches the data from the memory, decodes the and executes the instruction.

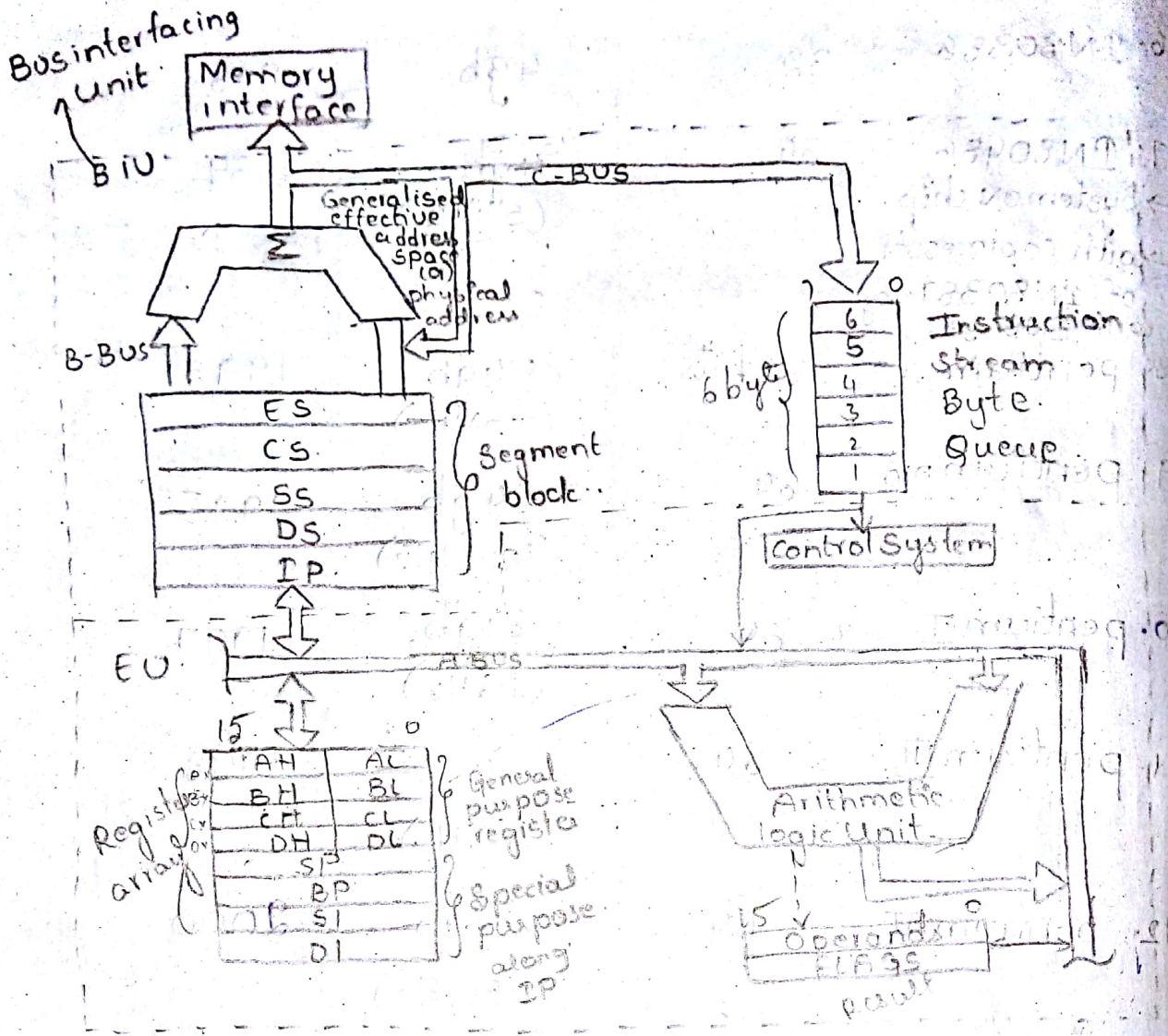
Digital data 4 types 1. Binary data 2. ASCII data 3. Signed number 4. Unsigned number

Evolution of Microprocessor:

Parameter	Data bits	Address bits	year of Introducing
1. IN 4004	4	8	1971
2. IN 8080	8	16	1974
3. IN 8085	8	16	1977
4. IN 8086 father of Microprocessor family	16	20	1978

5. IN 80286	32	24	1983
6. IN 80386	32	4gb	1986.
7. IN 80486 → System on chip → with coprocessor of IN 80387 along with IN 80386	64	4gb (5MHz)	1989.
8. pentium	64 bits	6x 4gb (100MHz)	1993
9. pentium pro	64	64 gb (100MHz)	1995
10. pentium II	64	64gb (450MHz)	1997.
11. pentium III	64	64gb (1GHz)	1998
12. pentium IV	64	64gb (1.5GHz)	2000.

8086 INTERNAL BLOCK DIAGRAM



BIU - Bus interfacing Unit holds and transfers the data

EU - Execution Unit → data is decoded by execution unit

ES - Extra Segment

CS - Code Segment

SS - Stack Segment

DS - Data Segment

IP - Instruction pointer

1. The instruction Queue prefetch instruction to Speed up the Operation.

It follows FIFO fashion.

Generals

BIU Used for transferring data between memory to Input Output and Input Output device to memory.

A → Accumulator → Initialization. The result temporarily hold in Accumulator.

B → Base register - Addressing base Value

C → Count register - loop Operation.

D → Data register - for string Operation.

SP → Stack pointer

BP → Base pointer

SI → Source Index Register.

DI → Destination Index Register.

A → Address bus

C → control bus

Segment registers are used for writing a program.

Working of 8086

Microprocessor is Very much essential in Computer.

And IT (Integrated Technology)

Designed by HMOS technology. It was introduced in 1978.

The microprocessor having important features like instruction queue store 6 bytes of instructions at a time to Speed Up the Operation. It follows FIFO fashion.

→ It is designed by 20 bit address lines 16 bits data lines and 1MB of memory

→ 8086 Comes in different Version 8 based on the

Operating frequencies.

8086 $\rightarrow$ 5 MHz

8086(2) $\rightarrow$ 8 MHz

8086(1) $\rightarrow$ 10 MHz. (commercial)

$\rightarrow$ This is used in military purpose, Industrial purpose, and Commercial purpose.

$\rightarrow$ 8086 is having additional clock generator

$\rightarrow$ Initially microprocessor increases number of instructions per second, because of Bus interfacing unit and Execution Unit.

This architecture is also called as pipe line architecture

Bus Interfacing Unit

$\rightarrow$ Bus interfacing unit is an interface between user and the execution. Its responsibility is to transfer data.

$\rightarrow$ Bus interfacing unit consists of two modules.

1. Segment registers.

2. Instruction queue.

Segment registers are used to hold temporary code and data. The capacity of each segment register is 64 KB.

There are 4 Segment registers

SS - Stack Segment

ES - Extra Segment

CS - Code Segment

DS - Data Segment

They are used to hold program code and

Each Segment register Specially designed with 64kbit memory ($4 \times 64 = 256 \text{ kb}$) for temporarily Code and data. Instruction pointer is Used to addresses the Stored instructions in Order.

Execution Unit

It is Used to execution of program Code along with data.

There are Some important parts work together to perform ALU Operations.

ALU

Arithmetic and logical Unit performs addition, Subtraction, multiplication, division. and also perform logical Operations and shift and rotate.

The program Code is executed by Using registers.

1. e Special purpose registers.

2. 1) pointer registers 2) Index registers

General purpose registers (AX, BX, CX, DX)

Finally the execution result is held into a flag register.

Flag: flag is a Status of result i.e. result of any program.

Flags are the 16 bit registers having 9 individual flags to give a Status.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	OF	DF	IF	TF	SF	ZF	X	AF	X	PF	X	CF

CF $\rightarrow$ Carry flag

PF $\rightarrow$ parity flag

AF $\rightarrow$ Auxiliary carry flag

ZF $\rightarrow$ Zero flag

SF $\rightarrow$ Sign flag

TF $\rightarrow$ Trap or Trace flag

IF $\rightarrow$ Interrupt flag

DF $\rightarrow$ Directional flag

OF $\rightarrow$ Overflow flag

Status flags

Control flags

Status flags

DF Sets Value of 1 when the data shifts from Left to right otherwise it Sets 0.

If IF = 1 interrupt is enable $\rightarrow$ enters to program.
= 0 otherwise $\rightarrow$

TF $\rightarrow$ Single Step flag $\rightarrow$ observes each and every instruction

If TF = 1 Single Step execution

= 0 otherwise.

It is

CF, CF = 1 Carry generates - Carry flag

PF = Parity flag even parity, odd parity

AF = If any ^{5th} bit is present after adding two 4 bit is known as Auxiliary Carry Flag.

ZF = Zero flag

Signed flag - Signed - MSB is 1

Unsigned - MSB is 0

Register Organisation of 8086.

The registers are used for writing a program and to hold the data temporarily.

8086 registers are classified into 4 types

1. General purpose register
2. Special purpose register
3. Segment registers
4. Flag registers

1) General purpose registers

15	8,7	0
AH	AL	
BH	BL	
CH	CL	
DH	DL	

2) Segment registers

SS
ES
CS
PS

3) Special purpose registers

15	0
SP	
BP	
IP	
SI	
DI	

4)

15	0
Flags	

AX - Accumulator (16 bit)

again divided into AL and AH

AL - lower byte AH - Higher byte

BX - Base register

To perform any Arith Operation it requires a base register

CX - Count register - loop Operation

DX - Data register - String

→ Segment registers hold the program code and data.
Each SR have 64 kb of memory part of 1 MB memory of 8086 to Speedup the Operation.

SS - Stack.
ES - Extra } String Operation

→ CS - holds Only program Code
DS - Data.

→ SP, BP Used to increases the length of Operation located at top of the Stack.

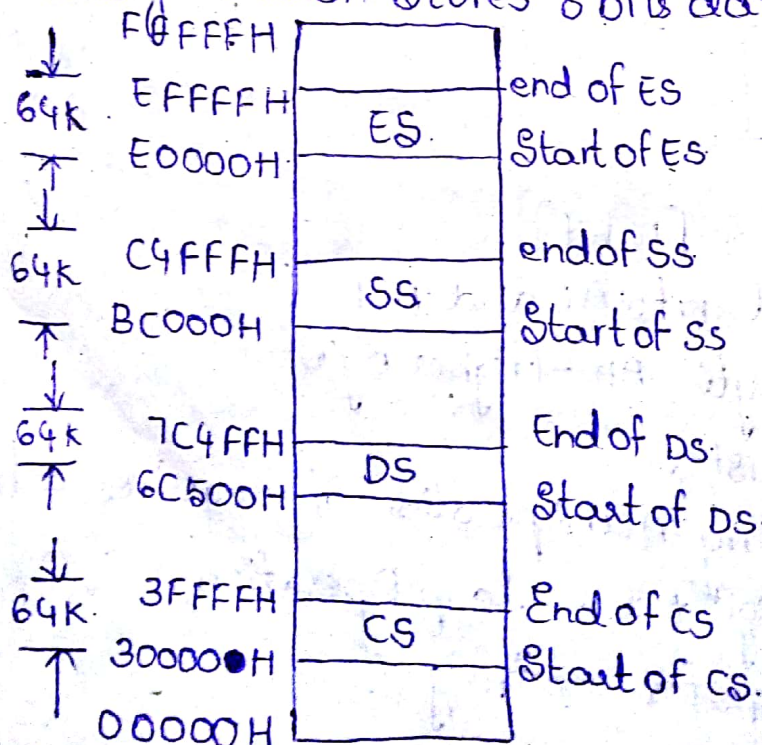
SI, DI, IP is Used to pointing Or addressing instruction in Instruction queue and memory

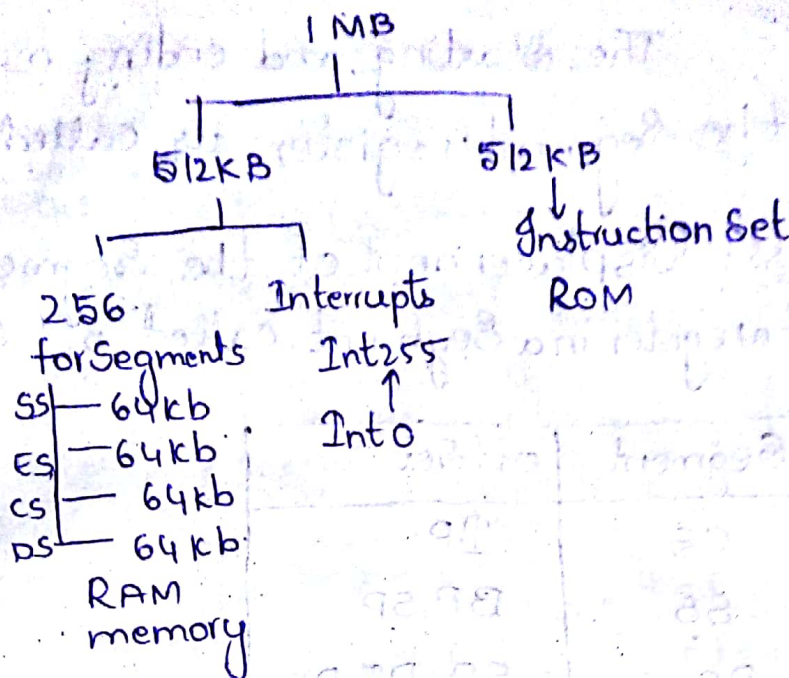
→ SI, DI is Used for performing input Output Operation

Memory Organisation of 8086

The memory Size of 8086 is 1 MB. It is divided into two parts i.e. ^{512 KB} ~~256~~ odd memory 512 KB of even memory.

Each memory location Stores 8 bits data.





In the above fig the 1 MB of memory is logically divided into 4 parts nothing but Segments

Each Segment Occupies 64 kb of memory for temporarily holding program Code and data

To address a memory location within a Specific Segment you need to know Start of the Segment

CS Starts with 30000H and ends with 3FFFFH

DS Starts with 6C500H and ends with 7C4FFH

SS Starts with BC000H and ends with C4FFFH

ES Starts with E0000H and ends with EFFFFH

The Start of the Segment address is called as effective address or offset address

Calculation of physical address

8086 is having 20 bit physical address so it can direct or store 16 bit data anywhere in the 1 MB memory.

The generation of 20 bit physical address consists of base address or effective address

Base address:- The starting and ending memory location of the segment registers is called base address.

Offset address:- Displacement of the segment register. Addresses of a register in a segment called offset address.

Example:-

Segment	offset
CS	IP
SS	BP, SP
DS	SI, DI, DX
ES	SI, DI, DX

Calculate 20 bit physical address of 8086 where the Code Segment CS = 2500H and IP = 3140H

$$PA = \text{seg} \times 16 + \text{offset address}$$

$$\begin{aligned} PA &= CS + 0 + IP \\ &= 2500 + 0 + IP \\ &= 25000 + 3140 \end{aligned}$$

$$PA = 28140$$

Calculate physical address of 16 bit processor by having Stack Segment 3FFF and base pointer is 1200.

$$PA = \text{Stack Segment} \times 16 + \text{offset address}$$

$$\begin{aligned} &= SS \times 16 + SP, BP \\ &= 3FFF + 1200 \end{aligned}$$

$$\begin{aligned} &= 31110 + 1200 \\ &= 32310 \end{aligned}$$

31110

1200

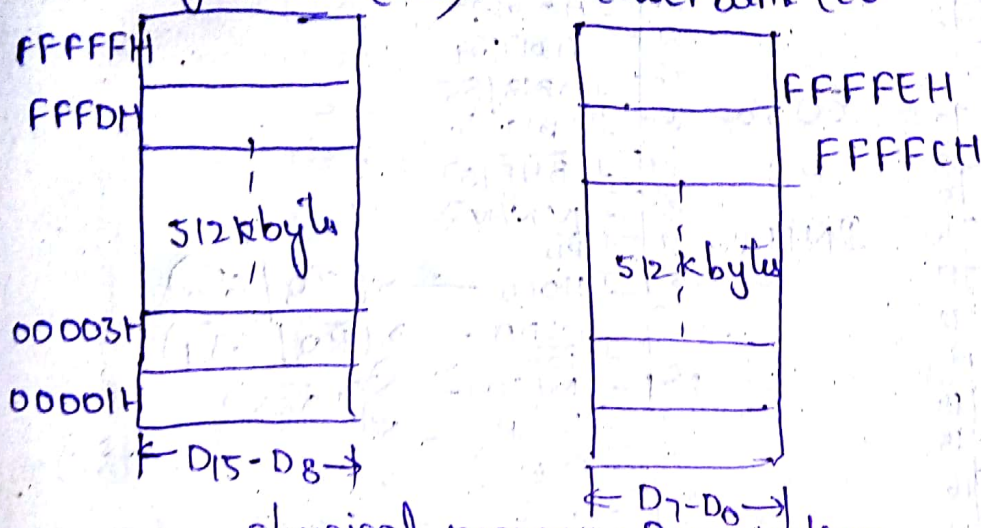
32310

physical memory Organisation of 8086.

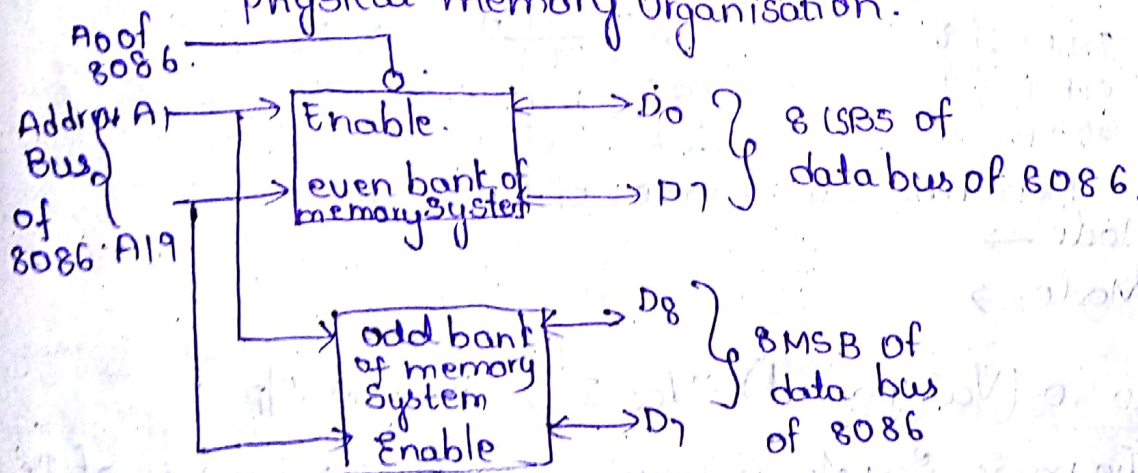
The memory of 8086 is 1MB is calculated by using 20bit address line ($2^{20} \approx 1\text{MB}$). Here 1MB of memory is divided into two types

1) 512KB - odd memory

512KB - even memory as shown in below fig



physical memory Organisation.



$\overline{\text{BHE}}$ of 8086 Interfacing Even and odd memory
 $\overline{\text{BHE}}$ Bus high enable

$\overline{\text{BHE}}$ - A₀

0 0 - Both

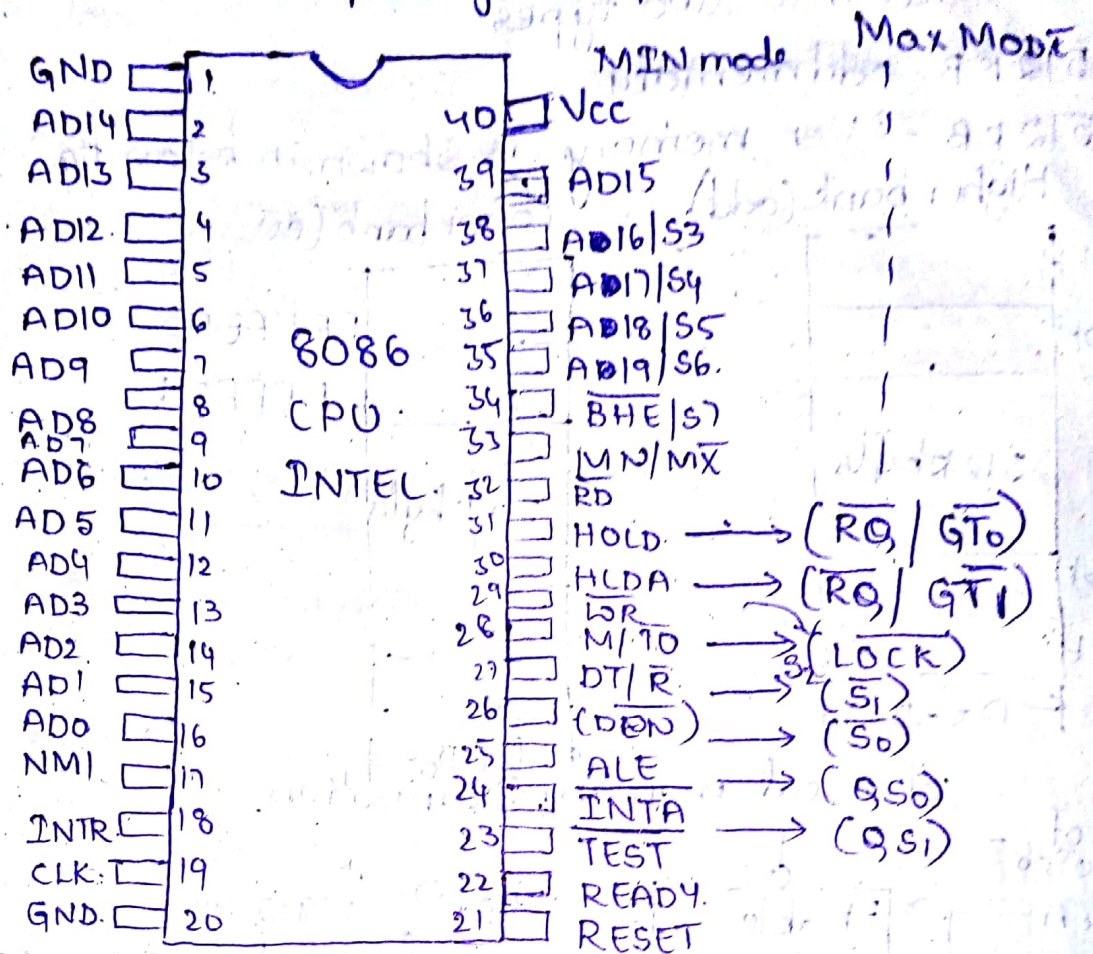
0 1 - odd

1 0 - even

1 1 - No one

Pin diagram of 8086

8086 is time multiplexed hardware device available in 40 pin DIP (Dual in package) 40p



8086 performs two modes of Operation

Min Mode →

Max Mode →

Pin 1, 20, 40 (Vcc, ground) Vcc-I/p Ground-O/p

For The power Supply is required for the activation of IC
It requires +5V. The ground terminal always initiate with 0. It bypass any noisy signals

→ AD0 → AD15 (from pin: 2 to 16, 39) these are multiplexed data and address signal used to hold the data for transmission (Input Output Signals)
→ These signals are used to provide interfacing memory to I/O and I/O to memory for fetching the data.

→ AD16/S3 to AD19/S6 These Signals are Output and time multiplexed. During first clock cycle ALE contains addresses lines separates from the Status lines.

→ S3 and S4 indicates the Segment registers being used for 20 bit physical address.

S3	S4	Segment register
0	0	ES - Extra Segment
0	1	SS - Stack Segment
1	0	CS - Code Segment
1	1	DS - Data Segment

→ AD18/S5 - It indicates Current interrupt flag.
Interrupt - execution pointer.

→ AD19/S6 - No Operation.

→ NMI - Non maskable interrupt
INTR - Interrupt request

} inputs to Microprocessor

NMI, INTR are interrupt signals.

→ clock - To Synchronise microprocessor Operation.

→ BHE/S7 (Bus high enable)

External 5MHz frequency - 8086
15MHz 8284

It is an Output and time multiplexed signals.

During first clock cycle BHE is separated during 2nd clock cycle S7 is set as high. BHE separates address bits and data bits.

→ MN/Mx ($\overline{\text{pin}}+33$)

When it is in MN mode it performs internal operation.

MN → Single processor operation

Mx → Multiprocessor operation

8087 is the Coprocessor to 8086:

when 8086 is in Maximum (MX) mode the 8087 gets linked with 8086.

$\overline{RD}$ (pin 32)

for Read Operation. (fetches the data from i/p)

Min. mode Signals

INTA (Interrupt Acknowledgement)
(active low signal)

ALE (Addresses latch enable) (pin 26)

pin 26 (Data enable) $\overline{DEN}$

Active low signal -- when the input is zero.

If data is present the signal is high.

This signal is used to active the data for transmission

pin 27 $DT/\overline{R}$ (

It is multiplex data transmitter and receiver.
It is like pin 33.

$DT/\overline{R} = 1$ transmitter

$DT/\overline{R} = 0$ receiver

pin 28 $M/\overline{IO}$

M - Memory

IO - Input Output

If the signal is 0 it fetches the instruction from IO

If the signal is 1 it fetches the instruction from memory.

Memory / Input device Output device.

It Selects memory or I/O operation

$\overline{WR}$ (Write)

This Signal is Used to indicate the Value of data to Write Or read after completion of program

Pin: 13 (HLDA), (HOLD)

Hold Acknowledgement - HLDA

These are Supporting devices.

ALE is connected with BHE

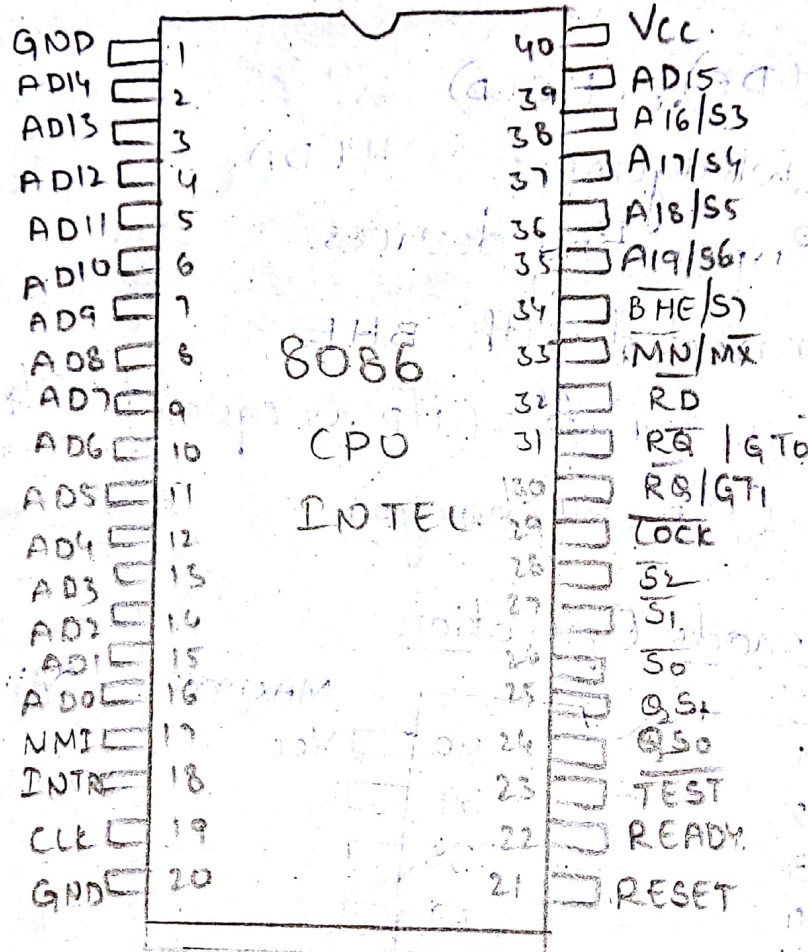
Latches acts as a buffer (i/p is equal to O/p)

In flip flop (O/p will change)

Qs₀ and Qs₁

This Signals are used to indicate the Status of instruction queue during the clock Cycle:

Maximum mode Operation



Qs ₀	Qs ₁	Operation
0	0	No Operation
0	1	first byte of an op code from the queue
1	0	Empty the Queue
1	1	Subset byte from the queue

50, 51, 52

These are called Status Signals to generate bus timing and control signals

$\overline{S_0}$	$\overline{S_1}$	$\overline{S_2}$	Status
0	0	0	Interrupt Acknowledgement
0	0	1	Read I/O port
0	1	0	Read, ~~and~~ write I/O port
0	1	1	Halt
1	0	0	Cache Access
1	0	1	Read I/O memory
1	1	0	Write memory
1	1	1	Inactive

LOCK:

It is an Output Signal activated by lock prefix instruction given by 8086.

$\overline{RQ}/\overline{GT_0}$, $\overline{RQ}/\overline{GT_1}$

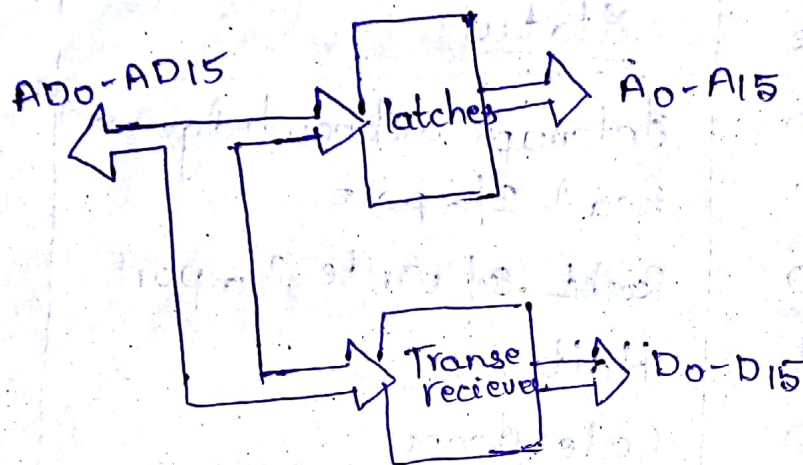
Bus request / Bus grant

General bus Operation of 8086

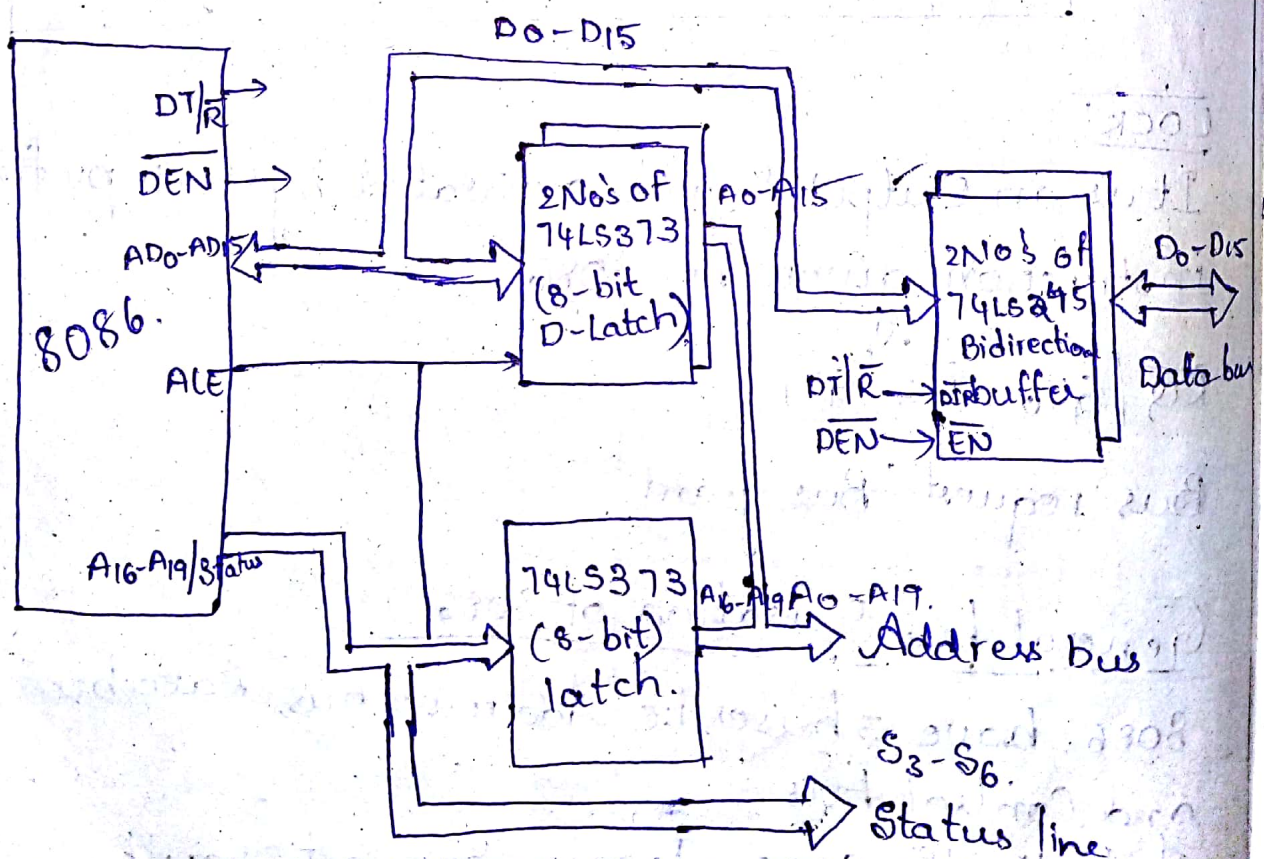
8086 have 3 buses i.e Address bus, Data bus and Control bus.

Initially the 8086 always supports multiplexed addresses and data signal for read and write operation. But peripheral devices not necessary

to carry multiplex addresses and data always so the addresses and data signals are demultiplexed by using latches and transceiver.



Interfacing of 8086 with latches and transceivers

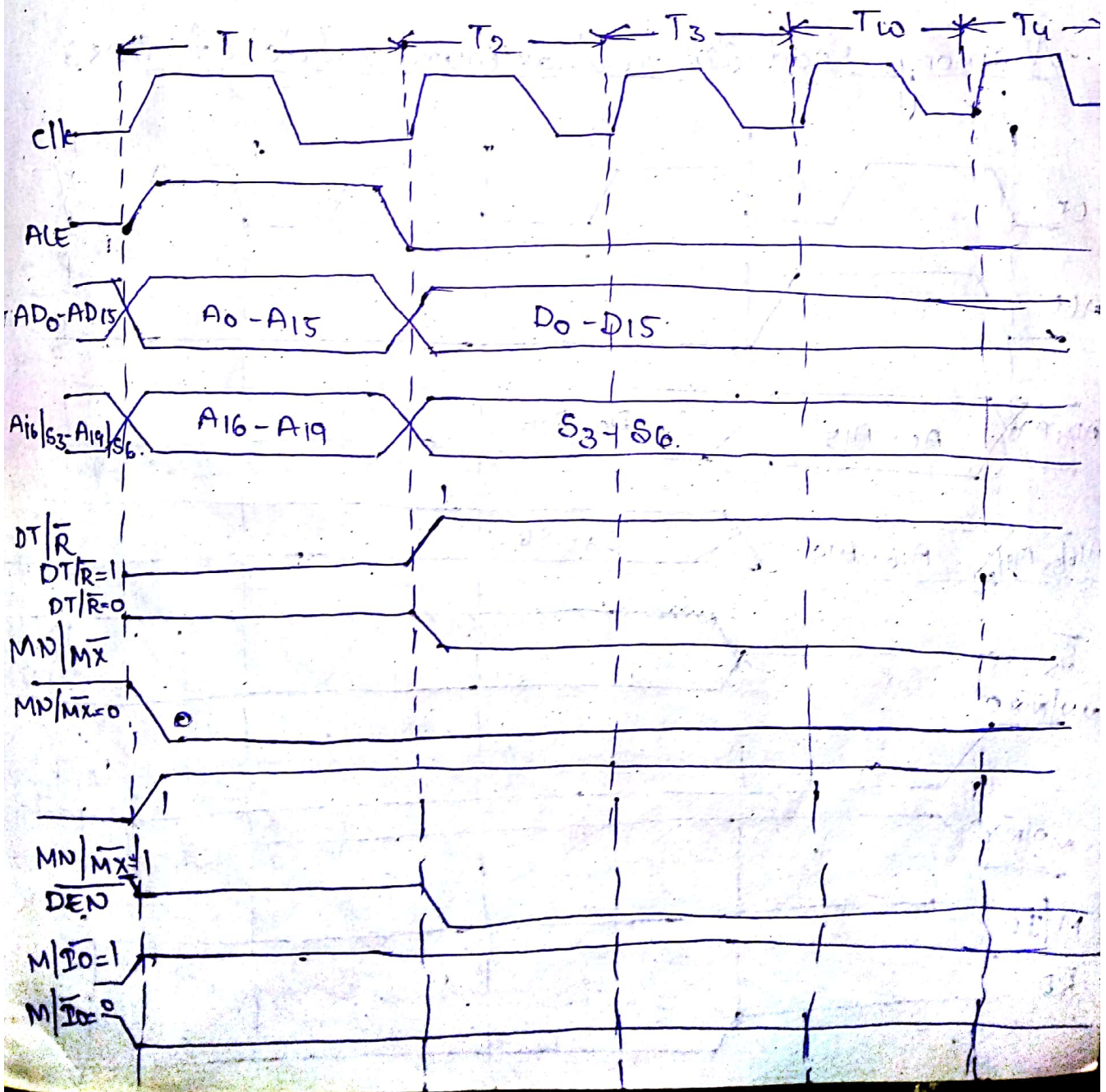


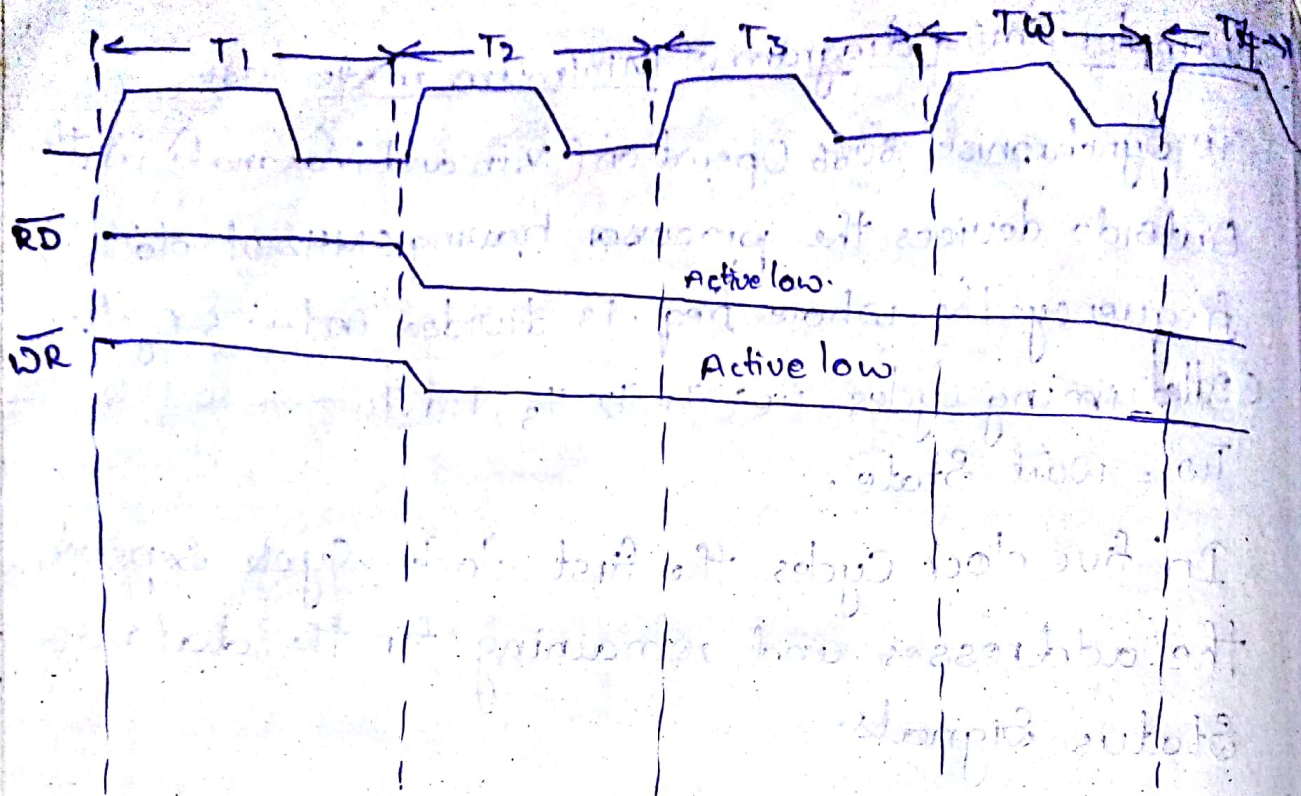
Example+ Timing diagram of Minimum mode 8086

To Synchronise 8086 Operation (Min and Max mode) with Outside devices the processor having 5MHz of clock frequency. The whole freq is divided into 5 cycles called Timing Cycles i.e. T_1, T_2, T_3, T_4, T_5 .

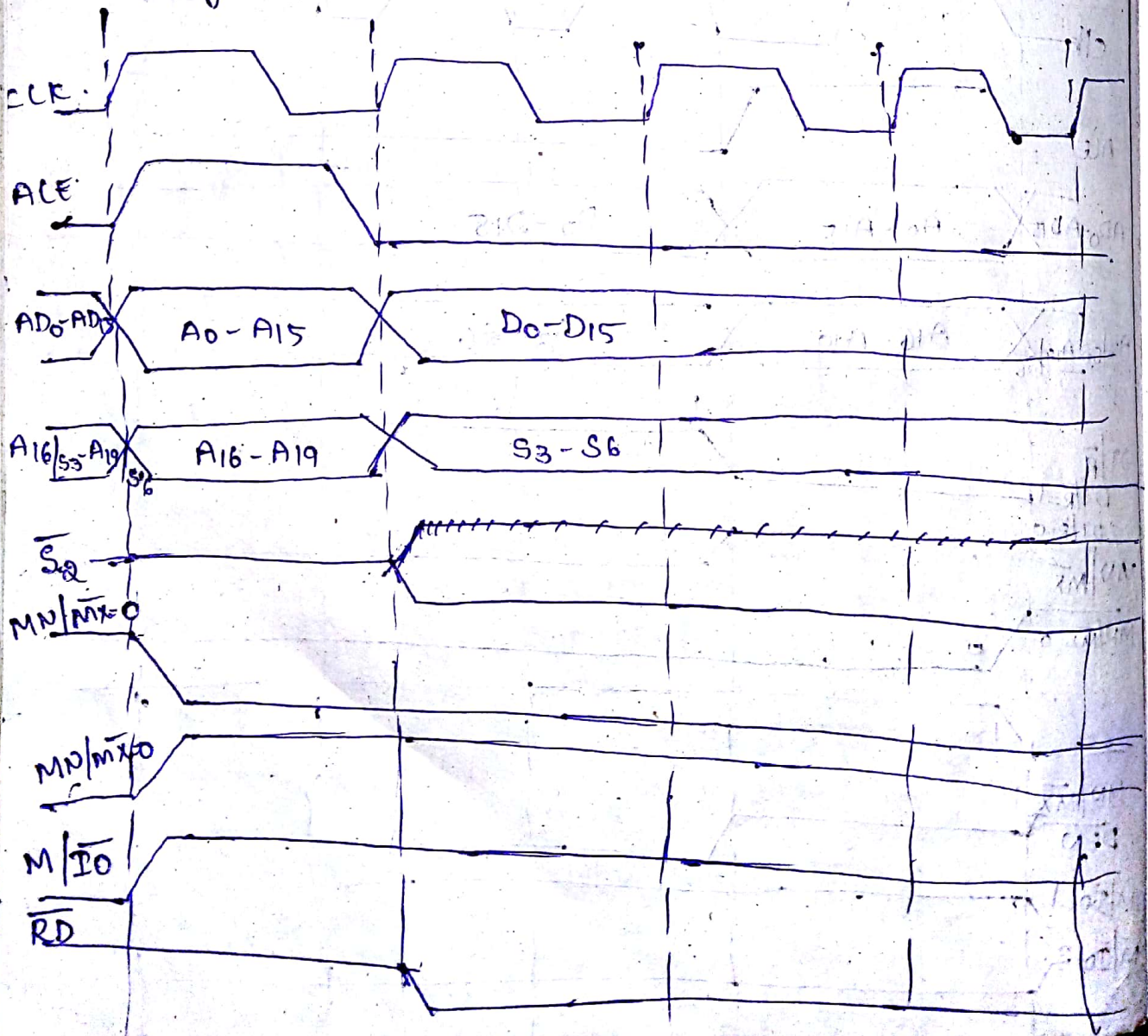
T_5 = wait State.

In five clock Cycles the first clock Cycle Separates the addresses and remaining for the data and Status Signals.

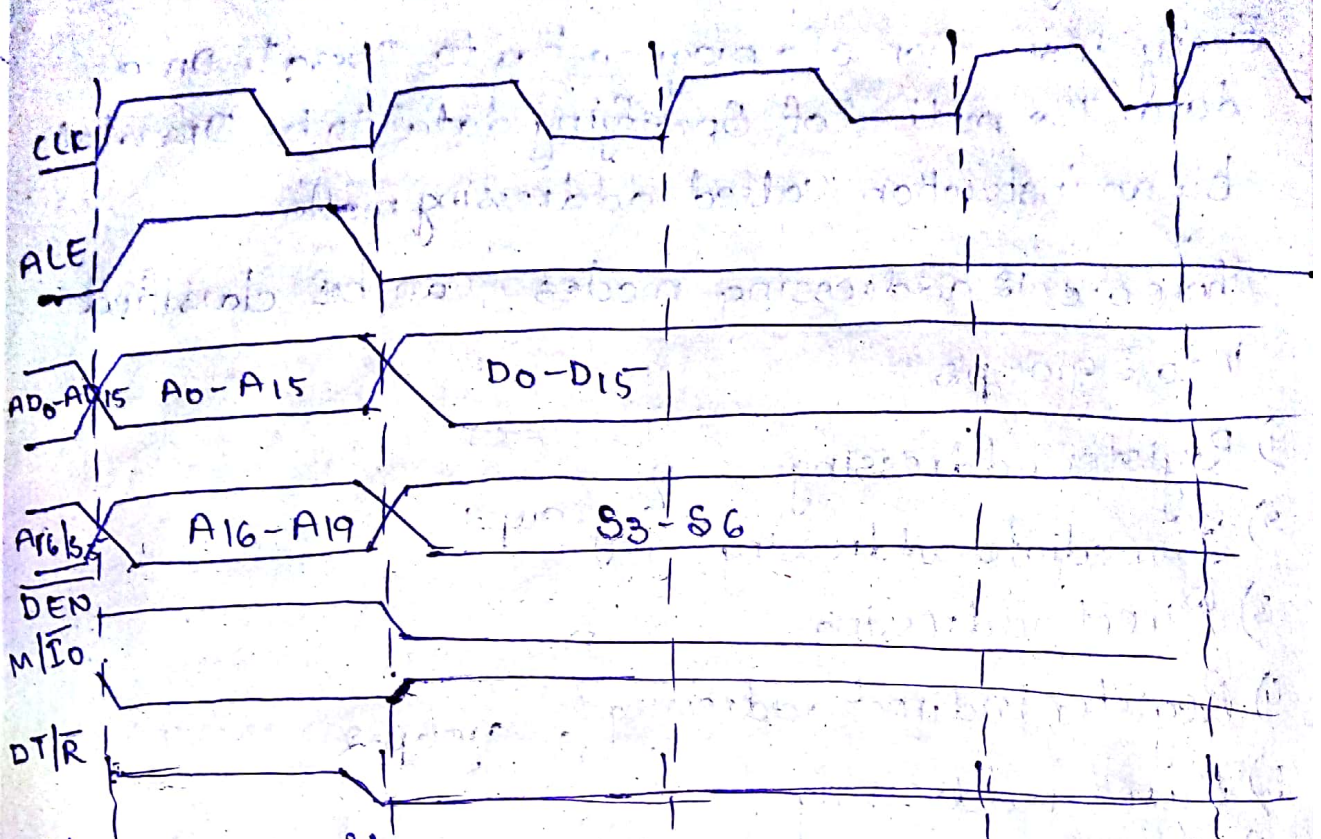




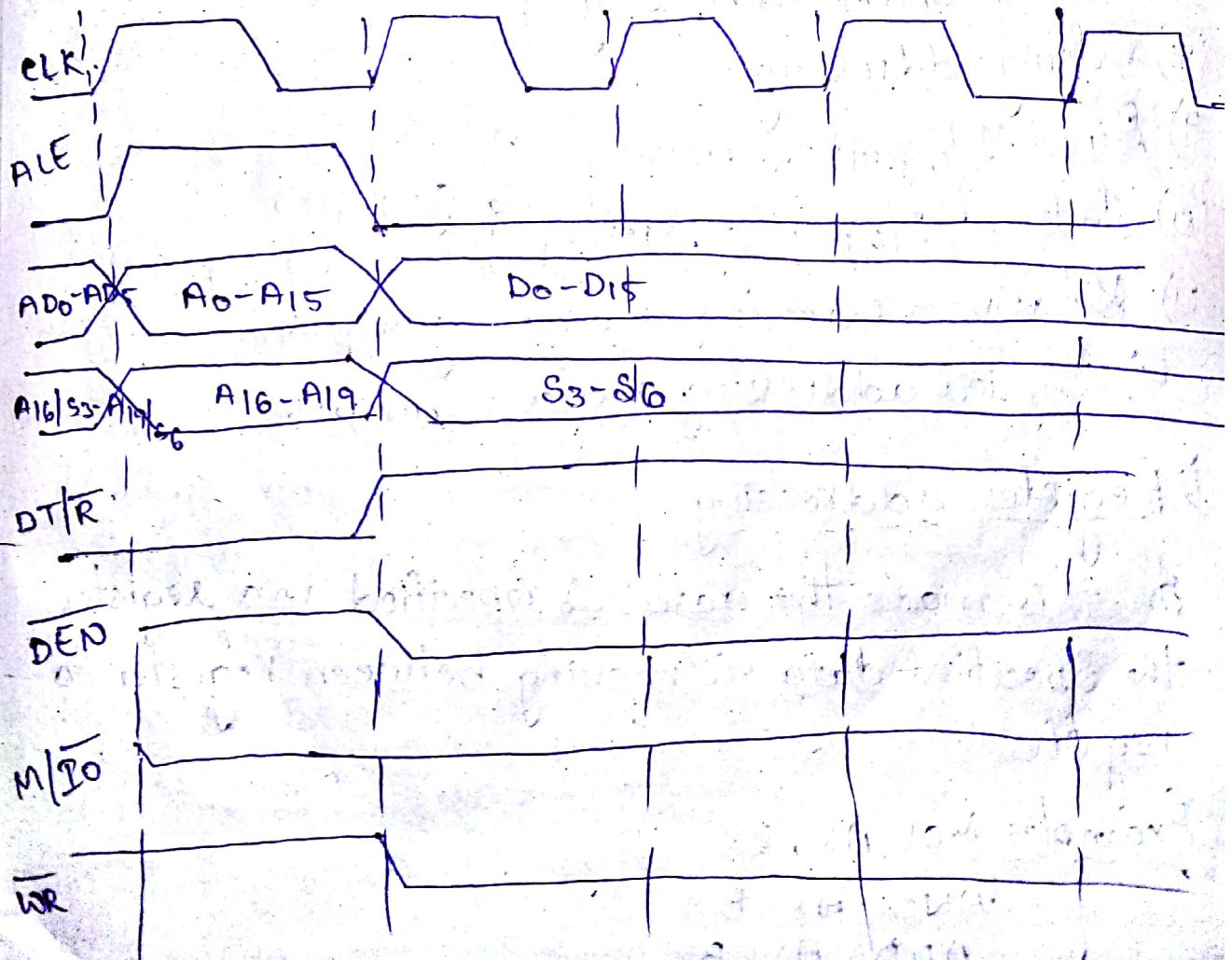
Memory read for in Maximum mode of 8086



I/O read in minimum mode



Memory write in minimum mode



Addressing modes of 8086.

Every instruction of a program has to operate on a data. The method of specifying data to be operated by an instruction called addressing mode.

There are 12 addressing modes. Can be classified into 5 groups.

- 1) Register addressing
 - 2) Immediate addressing
 - 3) Direct addressing
 - 4) Register indirect addressing
 - 5) Based addressing
 - 6) Indexed addressing
 - 7) Based Index addressing
 - 8) String addressing
 - 9) Direct I/O port addressing
 - 10) Indirect I/O port addressing
 - 11) Relative addressing modes
 - 12) Implied addressing modes
- group 1: (1, 2, 3)
group 2: (4, 5, 6, 7, 8)
group 3: (9, 10)
group 4: (11)
group 5: (12)

1) Register addressing

In this mode the data is specified in a register. The specified data is moving between register to register.

Exampler MOV AX, BX
AND AX, BX
ADD AX, BX

2) Immediate addressing mode

In this the 8 bit or 16 bit data is specified as part of the instruction

Ex: MOV AX, 0015

3) Direct addressing mode

In Direct addressing mode the 16 bit or 8 bit will be specified in the instruction with effective address.

Ex: MOV ^{SI} ~~AX~~, [0010]H
MOV AL, [08]H

4) Register indirect addressing mode

In register indirect addressing mode the name of the register holds effective addresses will be specified in the instruction

Ex: MOV AX, (SI)H

MOV AL, (DH)H

5) Base Addressing mode

Here the instruction specifies data in base registers (BX, BP) that is moved to another register

MOV SI, BP

LEA DX, BX

MOV BX, AL

Index Addressing mode

The data is operated between index register to another register.

Exampler.

MOV SI, BP

LEA DX, SI

MOV [DI], AL

Base Index Addressing mode:-

In this addressing mode the instruction specifies operand in base and index register to another register

MOV AX, [BX+SI]

String addressing mode

In this addressing mode the operand is specified in OPCode directly

Ext MOVSB

MOVSW

REP

CMP

Direct Input/Output port addressing mode or fixed I/O

1. Input addressing mode
2. Output port addressing mode

In this addressing mode instruction specifies the address and the port number

For Input IN AL, 40H

Out PORT, AX

Indirect I/O port addressing mode (Variable I/O)

In this addressing mode port is available in register DX before executing Input/output Operations

IN AL, [DX]

OUT DX, AX

Relative addressing mode

In this addressing mode the instruction specifies Operand in Source register with constant displacement to another register.

Example

LEA DX, [SI] 08H

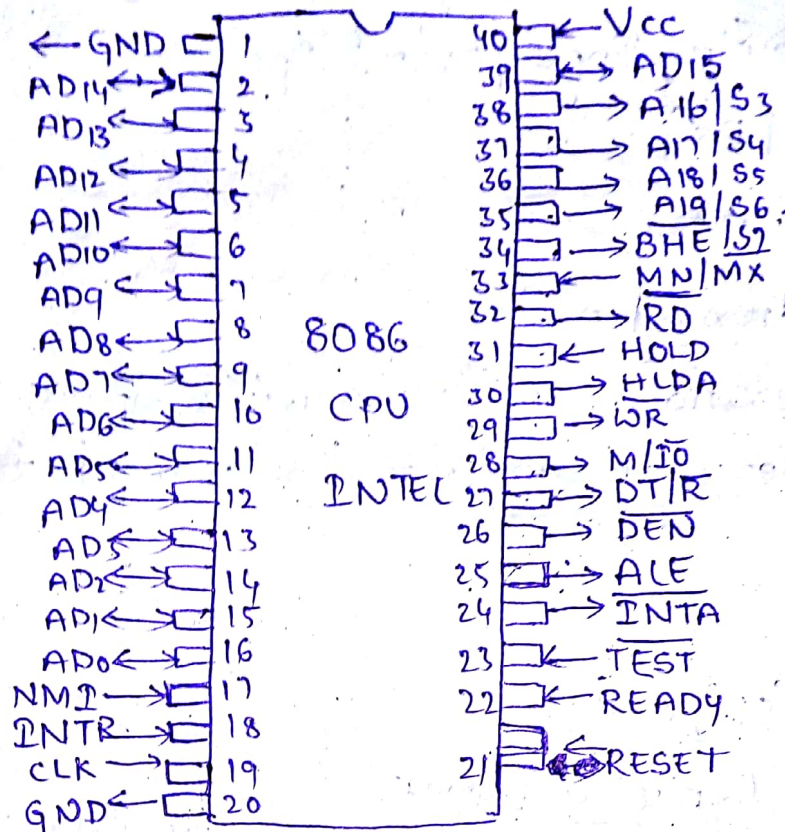
Implied addressing mode

In this addressing mode instruction is predefined.

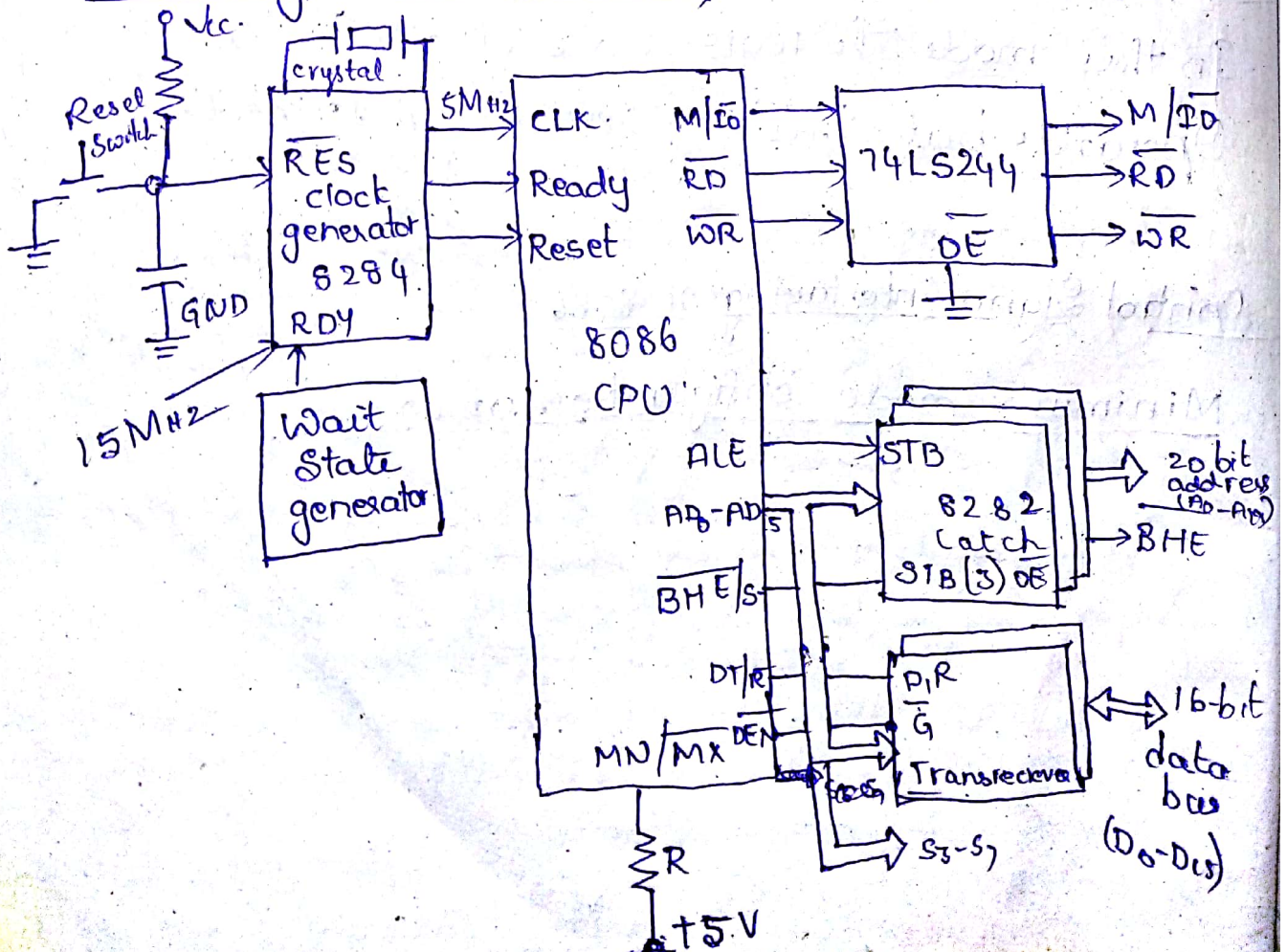
Ex MUL BX, AAA, DAA, AAM

In this mode the registers are used for specifying Operands but these registers are predefined.

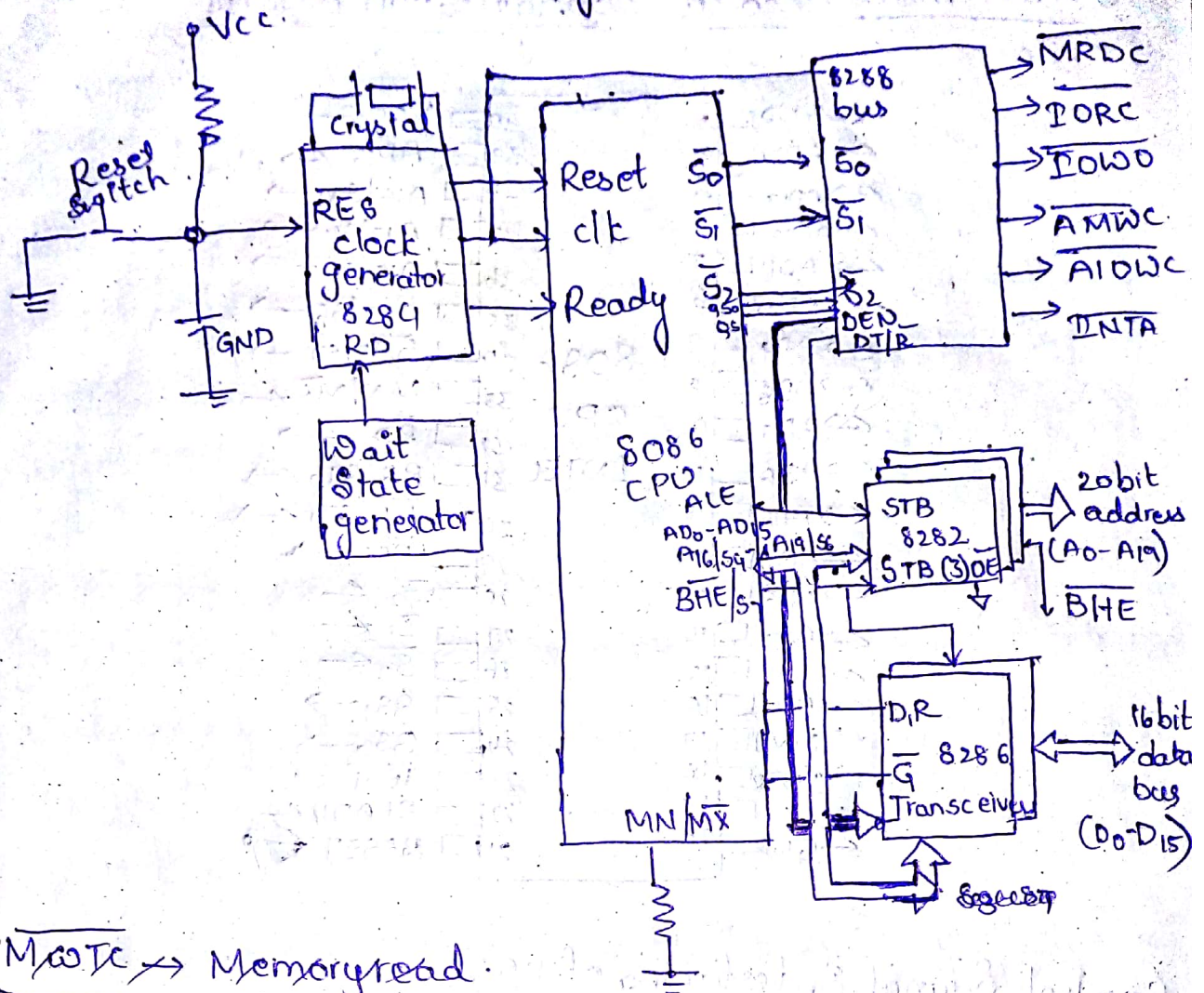
Minimum mode configuration of 8086



Interfacing of 8086 (Min mode)



Draw the Maximum mode Configuration. of 8086.



$\overline{MRDC} \rightarrow$ Memory read

$\overline{IORC} \rightarrow$ Memory write

$\overline{IOWC} \rightarrow$ I/O read

$\overline{MRDC} \rightarrow$ Memory read

$\overline{MWTC} \rightarrow$ Memory write

$\overline{IORC} \rightarrow$ I/O read

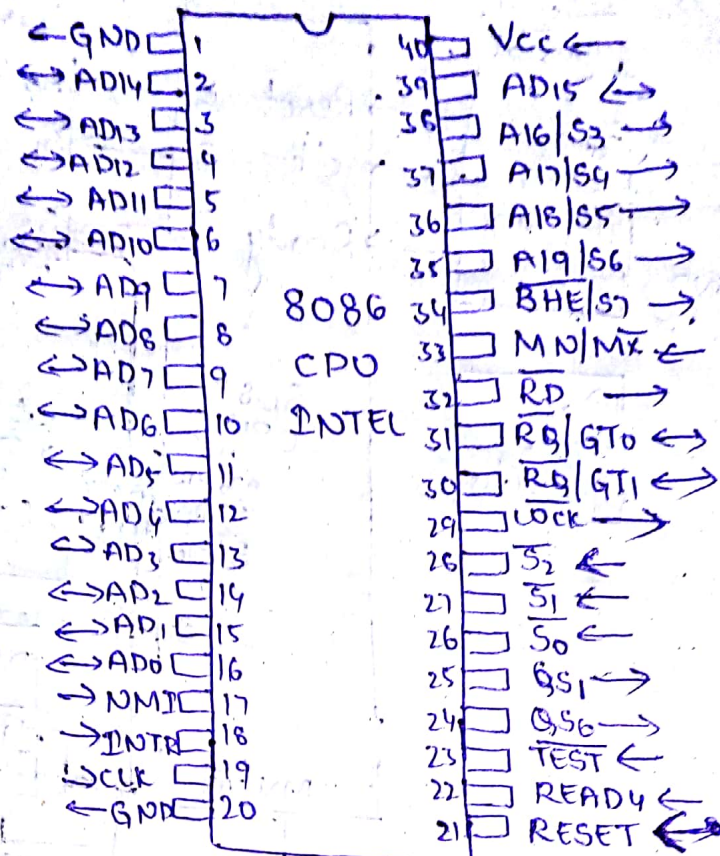
$\overline{IOWC} \rightarrow$ I/O write

$\overline{AMWC} \rightarrow$ Advanced memory write

$\overline{AIOWC} \rightarrow$ Advanced I/O write

$\overline{INTA} \rightarrow$ Interrupt Acknowledgement

Maximum mode Configuration of 8086



Control Signal Interfacing of 8086

The 8086 is a 40pin microprocessor having 40 available pins to interface external world easily

The 40pin Signals divided into Various groups based On their function One of the Signal group is called Control Signals.

MN/ $\overline{\text{MX}}$: It is a input Signal to 8086 It performs Single mode of Operation in minimum mode If the Signal is Set by 1 it is in min mode It performs multiple mode of Operation in max mode If the Signal is Set by 0 it is in max mode

BHE/S₁ : It is an Output Signal. If it is Set 0, it shows availability of buses, if it is 1, inactive.

DEP : It is a Output Signal. This Signal is Used for Data transmitting and receiving

INTR : It is a input Signal. It is active high when it is logic 1. It interrupts the Signal.

Xclock : It is an input Signal. It is Used for the Synchronisation of data

NMI : It is an input Signal. when it is logic 1, it interrupts the Signal

INTA : It is an Output Signal. It is Used for interrupt Acknowledgement. Active low Signal.

HOLD : It is an input Signal. It will hold the information

HLDA : Holding information is given.

LOCK : It is an Output Signal. It performs predefined Operation

DT/R : It Controls the data direction.

ALE : It is Used to first Separate the Address Signal and data Signals.

RQ/GT₁ : Request and grant Signals

Advanced microprocessor 80286

80286 is advanced Version of 8086 microprocessor and is designed for multitasking and multitasking.

It has memory management Unit. [16 mega bytes physical memory and 1 GB Virtual memory]

It has two modes of memory Operation

i.e) Real mode

2) protected mode

It is ~~see~~ designed by ~~see~~ SoC technology (System on chip)

80386

It is an enhanced Version of 80286 and also include MMU is enhanced to provide memory page

It also includes 32 bit enhanced registers and 32 bit addresses and data buses.

The memory Size is 4 gigabytes.

The register Setup 80386 Contains extended (32 bit) Versions of registers like EAX, EBX, ECX, EDX, EBP, ESP, EDI, ESI, EIP, EFlags

80386 increases the accessing time and reducing the Speed of memory.

80486

It is an improved Version of 80386 that Contains 8kbytes of Cache memory and 80387 Arithmetic coprocessor, it executes many instructions in one clocking period.

80486 executes a few new instructions that Controls the internal Cache memory.

A new feature found in 80486 is BIST (Built in Self Test)

In 80486 additional test registers are added to allow the Cache memory to be tested.

New Cache registers

- 1) Cache data
- 2) Cache States
- 3) Cache Control

Pentium

Pentium is a family of 32 and 64 bit microprocessor chip from intel.

Present pentium chips are widely used CPU's in the world for general purpose computing.

The first pentium chip was introduced in 1993 as the successor to the 80486.

It uses 64 bit internal bus with 4Gb memory.

It is the 4th generation microprocessor.