MICROPROCESSOR AND MICROCONTROLLER

III BTECH II SEM

ELECTRICAL AND ELECTRONICS ENGINEERING

R20

(JNTUK)

VSM COLLEGE OF ENGINEERING
RAMACHANDRAPURAM,
ANDHRA PRADESH
533255

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA–533003, Andhra Pradesh, India
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

III Year – II SEMESTER		L	T	P	C
		3	0	0	3
MICROPROCESSORS AND MICROCONTROLLERS					

Preamble:
Microprocessor and Microcontroller have become important building blocks in digital electronics design. It is important for student to understand the architecture of a microprocessor and its interfacing with various modules. 8086 microprocessor architecture, programming, and interfacing is dealt in detail in this course. Interfacing, PIC, architecture, programming in C.

Course objectives:
- To understand the organization and architecture of Microprocessor
- To understand addressing modes to access memory
- To understand 8051 micro controller architecture
- To understand the programming principles for 8086 and 8051
- To understand the interfacing of Microprocessor with I/O as well as other devices
- To understand how to develop cyber physical systems

UNIT - I
Introduction to Microprocessor Architecture
Introduction and evolution of Microprocessors – Architecture of 8086 – Memory Organization of 8086 – Register Organization of 8086– Introduction to 80286 - 80386 - 80486 and Pentium (brief description about architectural advancements only).

UNIT - II
Minimum and Maximum Mode Operations
Instruction sets of 8086 - Addressing modes – Assembler directives - General bus operation of 8086 – Minimum and Maximum mode operations of 8086 – 8086 Control signal interfacing – Read and write cycle timing diagrams.

UNIT - III
Microprocessors I/O interfacing
8255 PPI– Architecture of 8255–Modes of operation– Interfacing I/O devices to 8086 using 8255– Interfacing A to D converters– Interfacing D to A converters– Stepper motor interfacing– Static memory interfacing with 8086.
Architecture and interfacing of 8251 USART – Architecture and interfacing of DMA controller (8257).

UNIT - IV
8051 Microcontroller
Overview of 8051 Microcontroller – Architecture– Memory Organization – Register set – I/O ports and Interrupts – Timers and Counters – Serial Communication – Interfacing of peripherals- Instruction set.

UNIT - V
PIC Architecture
Block diagram of basic PIC 18 micro controller – registers I/O ports – Programming in C for PIC: Data types - I/O programming - logical operations - data conversion.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA–533003, Andhra Pradesh, India
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

Course Outcomes:

After the completion of the course the student should be able to:

- Know the concepts of the Microprocessor capability in general and explore the evaluation of microprocessors.
- Analyse the instruction sets - addressing modes - minimum and maximum modes operations of 8086 Microprocessors
- Analyse the Microcontroller and interfacing capability
- Describe the architecture and interfacing of 8051 controller
- Know the concepts of PIC micro controller and its programming.

Text Books:

1. Ray and Burchandi - "Advanced Microprocessors and Interfacing" - Tata McGraw–Hill - 3rd edition - 2006.
2. Kenneth J Ayala - "The 8051 Microcontroller Architecture - Programming and Applications" - Thomson Publishers - 2nd Edition.
3. PIC Microcontroller and Embedded Systems using Assembly and C for PIC 18 - -Muhammad Ali Mazidi - RolindD.Mckinay - Danny causey -Pearson Publisher 21st Impression.

Reference Books:

1. Microprocessors and Interfacing - Douglas V Hall - Mc–Graw Hill - 2nd Edition.
2. R.S. Kaler - "A Text book of Microprocessors and Micro Controllers" - I.K. International Publishing House Pvt. Ltd.
3. Ajay V. Deshmukh - "Microcontrollers – Theory and Applications" - Tata McGraw–Hill Companies –2005.
4. Ajit Pal - "Microcontrollers – Principles and Applications" - PHI Learning Pvt Ltd - 2011.

UNIT 1. INTRODUCTION TO MICROPROCESSOR ARCHITECTURE

UNIT 2. MINIMUM AND MAXIMUM MODE OPERATION

Unit-I

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.

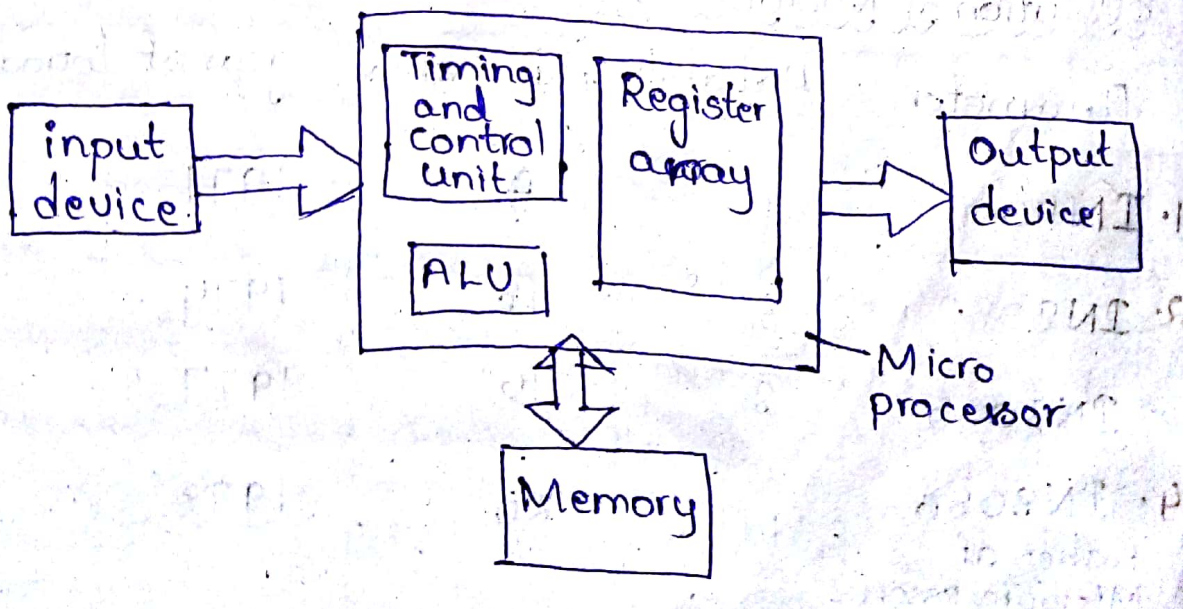
Example of Digital Ic is Microprocessor

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.

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

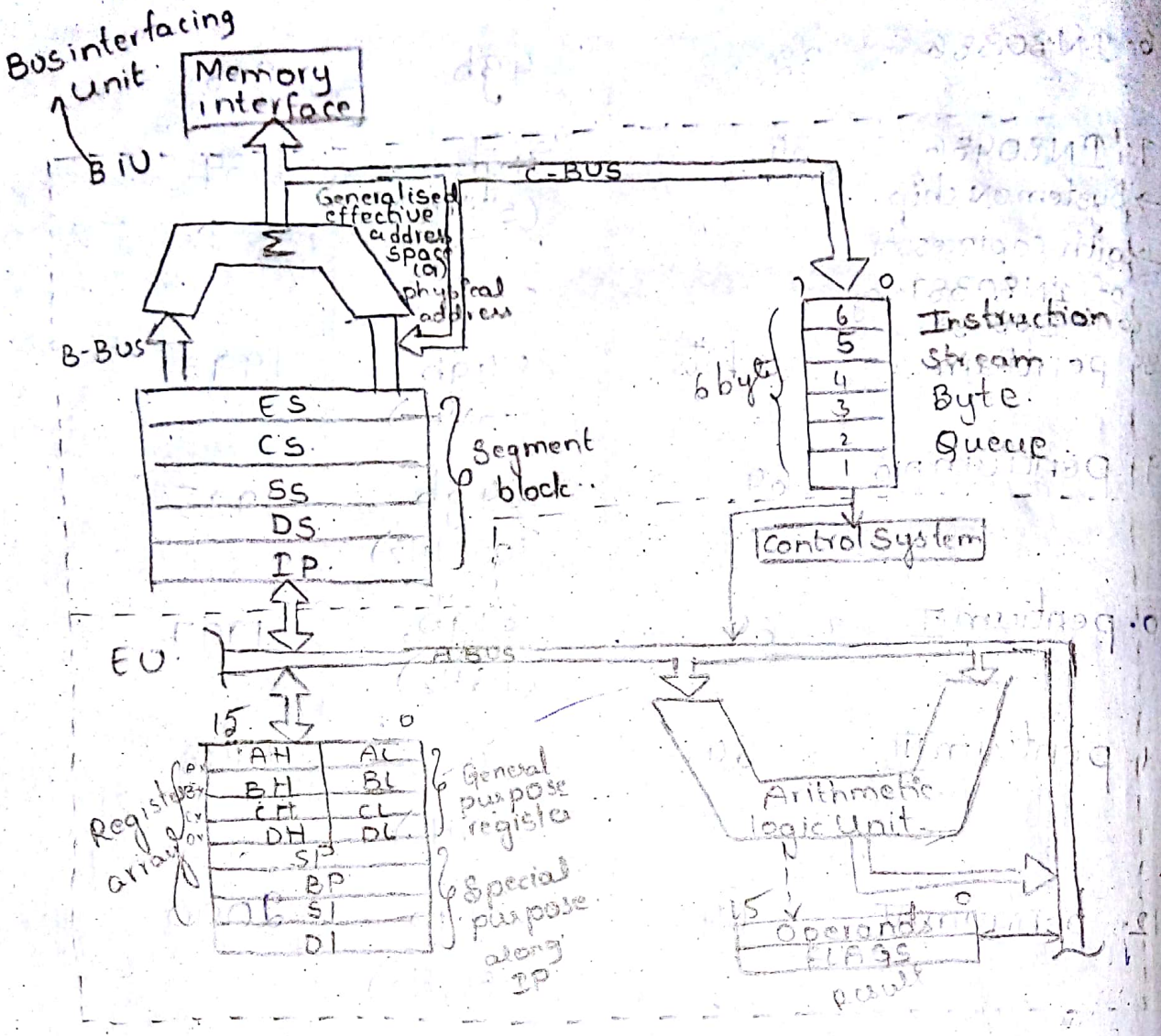
Digital data 4 types 1. Binary data 2. ASCII data 3. Signed number 4. UnSigned number

Evolution of Microprocessor:

Parameter	Databits	Addressbits	year of Introducing
1. IN4004	4	8	1971
2. IN8080	8	16	1974
3. IN8085	8	16	1977
4. IN8086 father of Microprocessorf family	16	20	1978

5. IN 80286	32	24	1983
6. IN80386	32	4gb	1986.
7. IN80486. → System on chip → with coprocessor of IN80387 along with IN80386	64	4gb (5MHz)	1989.
8. pentium	64 bits	4gb (100MHz)	1993
9. pentium pro	64	64gb (100MHz)	1995
10. pentium II	64	64gb (450M ~~1~~ GHz)	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.

Generalis

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

AL → Accumlator → 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 Versions based on the

Operating frequencies:

8086 → 5 MHz

8086(2) → 8 MHz

'c': 8086(1) → 10 MHz. (commercial)

→ This are used in military purpose, Industrial purpose, and commercial purpose.

→ 8086 is having additional clock generator

→ 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:

→ Bus interfacing unit is an interface between User and the execution. It responsibility is to transfer data.

→ 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

There are used to hold program code and

Each Segment register Specially designed with 64kbot memory (4x64 = 256kb) for temporarly 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:

Arithmatic and logical Unit performs addition, Subtract multiplication, division. and also perform logical Operations and shift and rotate

The program Code is executed by Using registers.

1.e Special purpose registers.

1) pointer registers 2) Index registers

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

Finally the execution result is holded into a flag register

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

Flags are the 16 bit registers having 9 invidual 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 → Carry flag
PF → parity flag.
AF → Auxilary carry flag } Status flags
ZF → zero flag.
SF → Sign flag.
TF → Trap or Trace flag } Control flags.
IF → Interrupt flag.
DF → Directional flag.
OF → Overflow flag. } 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 → enters to program.
=0 Otherwise →

TF → Single Step flag → obseves each and every Instruct
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 Auxilary Carry flag.
ZF = Zero flag

Signed flag - Signed - MSB is 1
Unsigned - MSB is 0

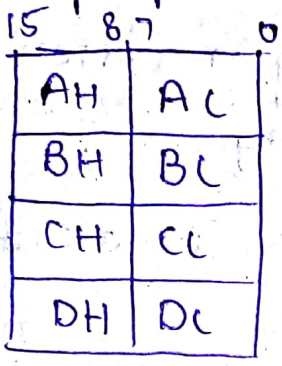
Register Organisation of 8086.

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

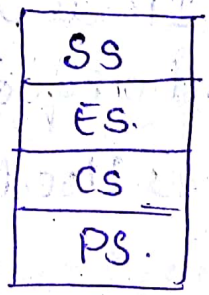
8086 registers are classified into 4 types

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

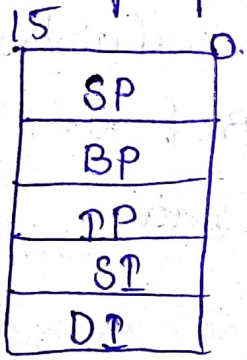
1) General purpose registers



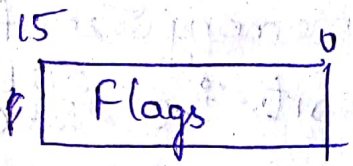
2) Segment registers



3) Special purpose registers



4)



AX - Accumlator (16 bit)
again divided into AL and AH
AL - lower byte AH - Higher byte

BX - Base register
To perform any ALU Operation it requires a base regist

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 Speed up the Operation.

SS - Stack. } String Operation
ES - Extra }

→ CS - holds only program code
DS - Data.

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

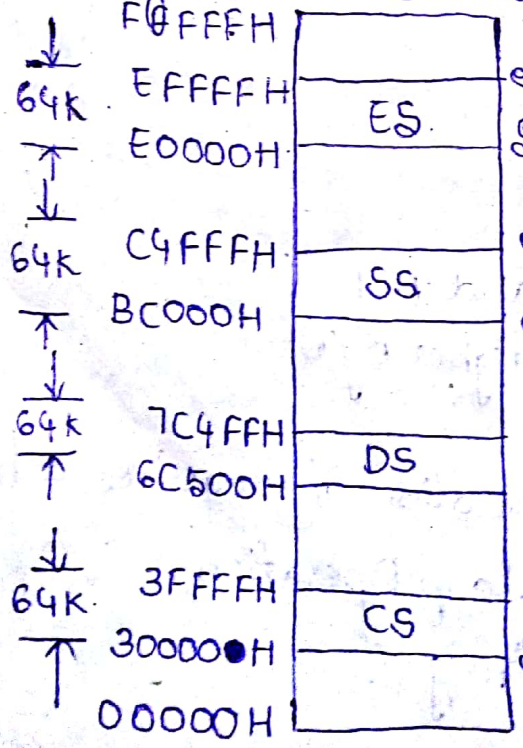
SI, DI, IP is used to pointing or addressing instruct in Instruction queue and memory

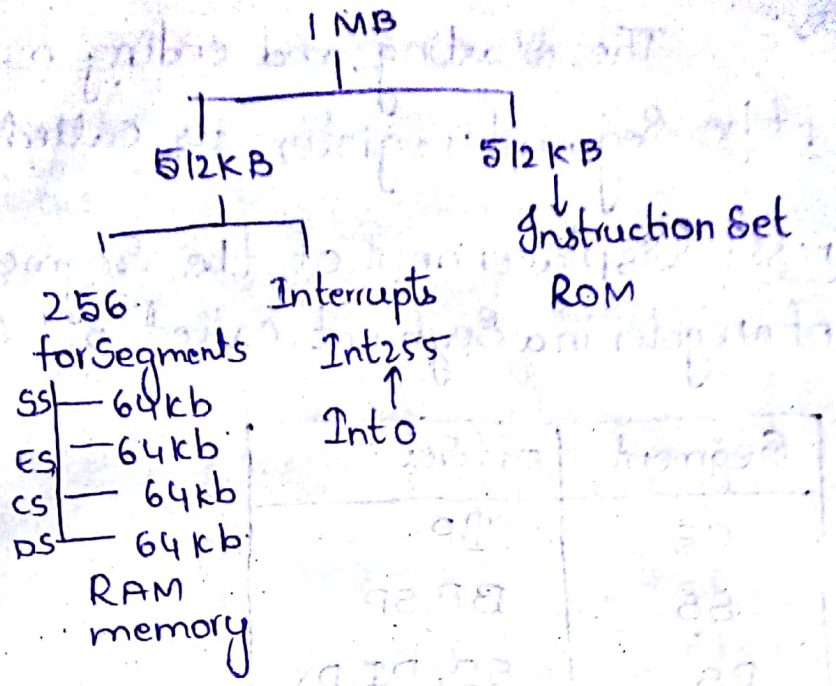
→ SI, DI is used for performing input output Operatio

Memory Organisation of 8086

The memory Size of 8086 is 1MB. It is divided in to two parts i.e 512 kB 512 kB odd memory 512 kB of even memory.

Each memory location Stores 8 bits data.





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

Each Segment Occupies 64kb 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 B0000H 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 1MB memory.

The generation of 20 bit physical address Consist 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 = seg X 16 + offset address

PA = CS + 0 + IP

= 2500 + 0 + IP

= 25000 + 3140

PA = 28140

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

PA = Stack Segment X 16 + offset address

= SS X 16 + SP, BP

= 3FFF + 0 + 1200

= 31110 + 1200

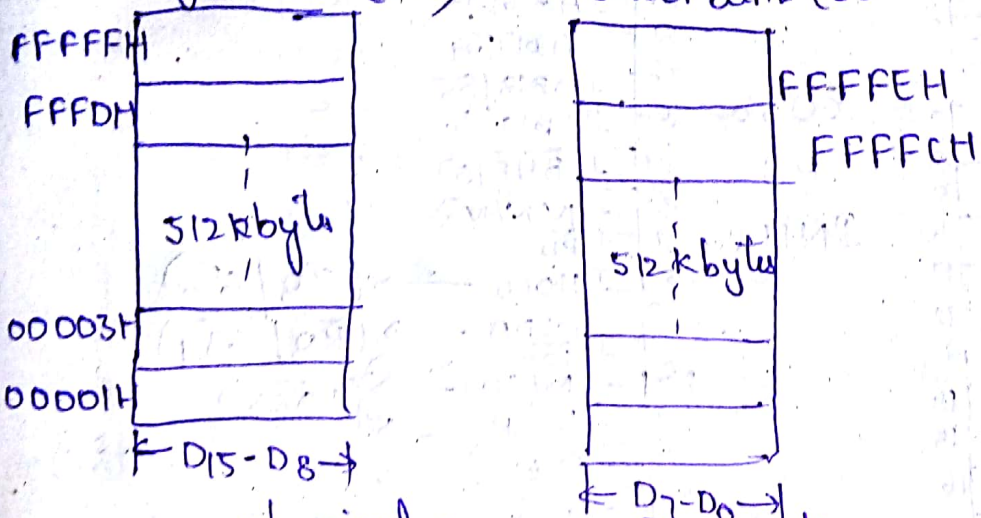
= 32310

31110
1200
32310

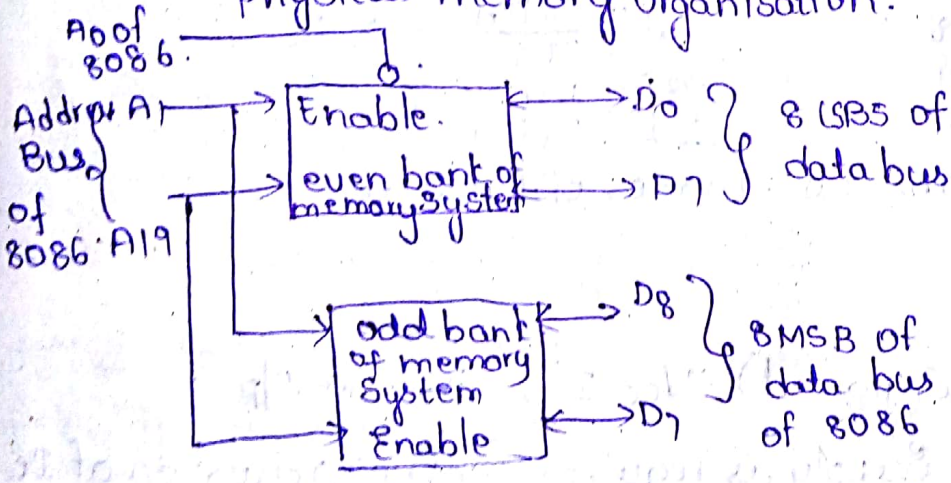
physical memory Organisation of 8086.

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

1) 512KB - odd memory
   512KB - even memory as shown in below fig



physical memory Organisation.



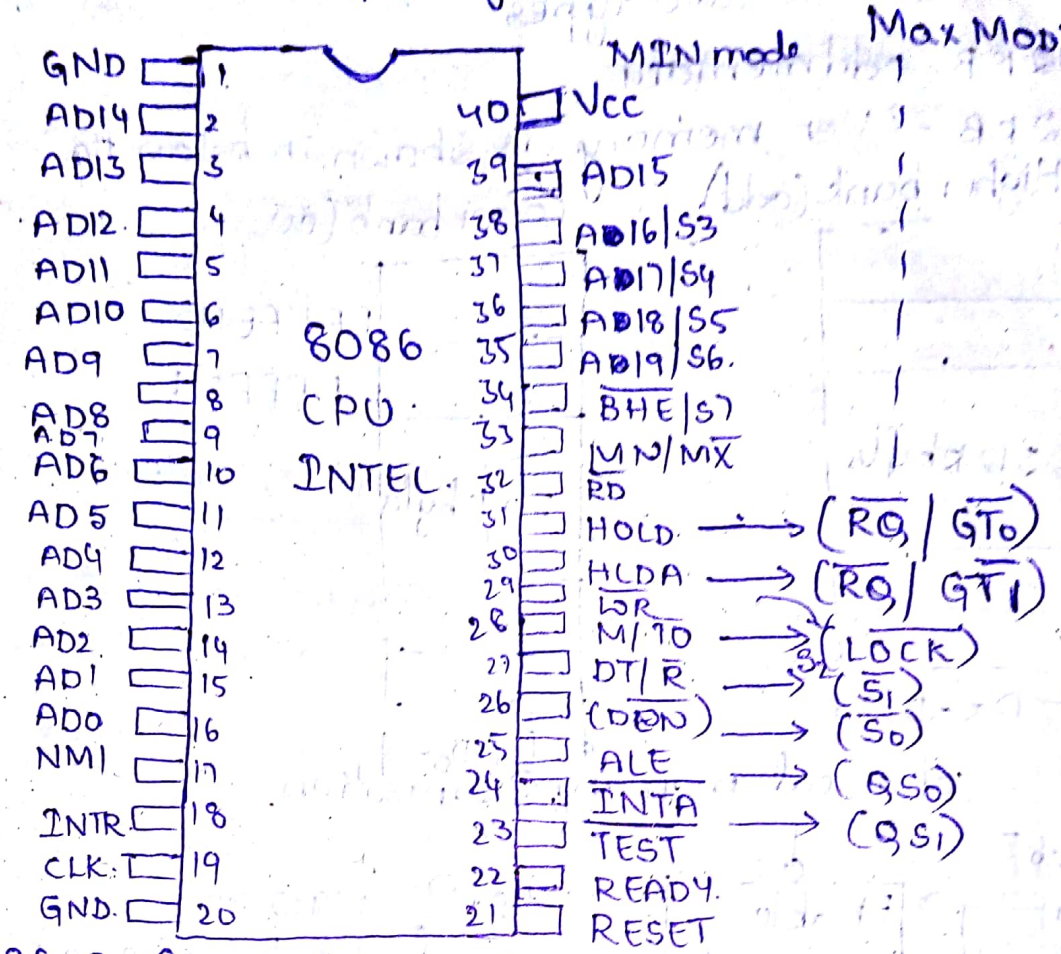
$\overline{BHE}$ of 8086 Interfacing Even and odd memory

$\overline{BHE}$ Bus high enable

$\overline{BHE}$	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) chip



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.

→ $AD_0 \rightarrow AD_{15}$ (from pin:- 2 to 16, 39) these are multiplex data and adress 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. Th during first clock cycle. ALE contains addresses lines seperates from the Status line.

→ 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 clk frequency - 8086 5MHz, 8284 15MHz

It is an Output and time multiplexed Signals. During first clock cycle BHE is Seperated during 2nd clock Cycle S7 is Set as high. BHE Seperates address bits and data bits

→ MN/MX (pin 33)

when it is in MN mode it performs internal Operati
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

$\overline{INTA}$ (Interrupt Acknowledgement)
(active low Signal)

ALE (Addresses latch enable) (pin26)

pin26:- (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 transmissio

pin27 DT/$\overline{R}$ (

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

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

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

pin28 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 IO Operation.

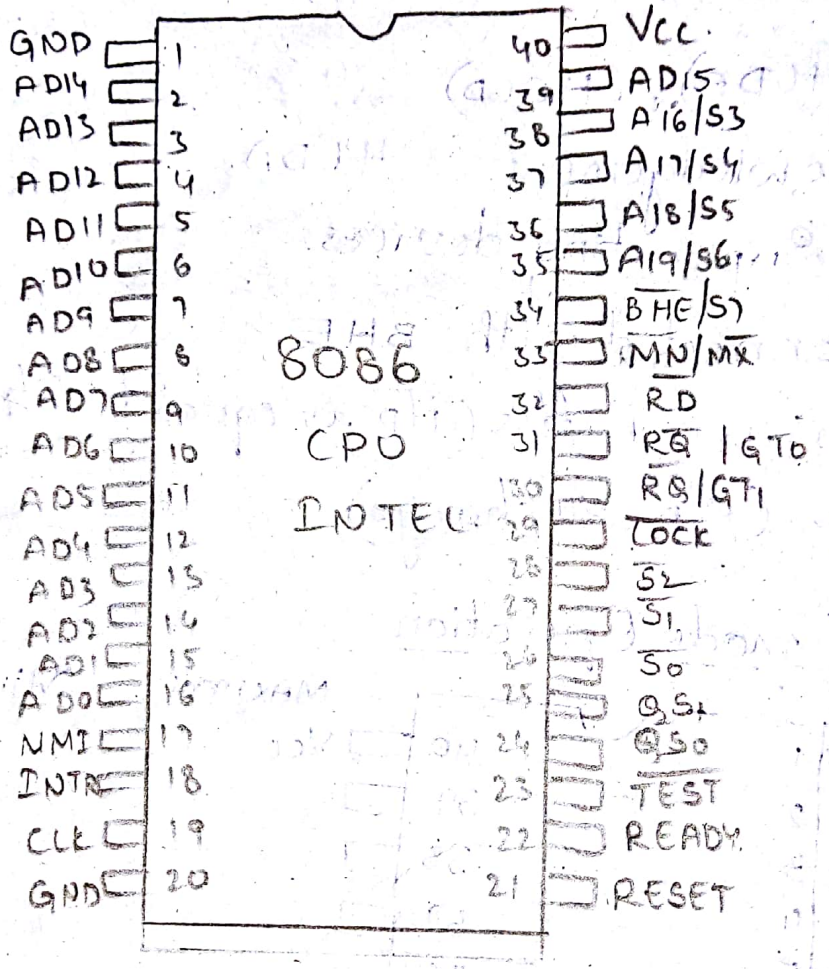
WR (Write)
This Signal is Used to indicate the Value of data to Write Or read. after completion of program

pin-13 (HLDA), (HOLD)
Hold Acknowledgement - HLDA
Those 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)

QS_0 and QS_1

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

Maximum mode Operation



QS_0 C

QS_1	QS_0	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

$\overline{S_2}, \overline{S_1}, \overline{S_0}$

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	Write I/o port
0	1	1	Halt
1	0	0	Code 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.

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

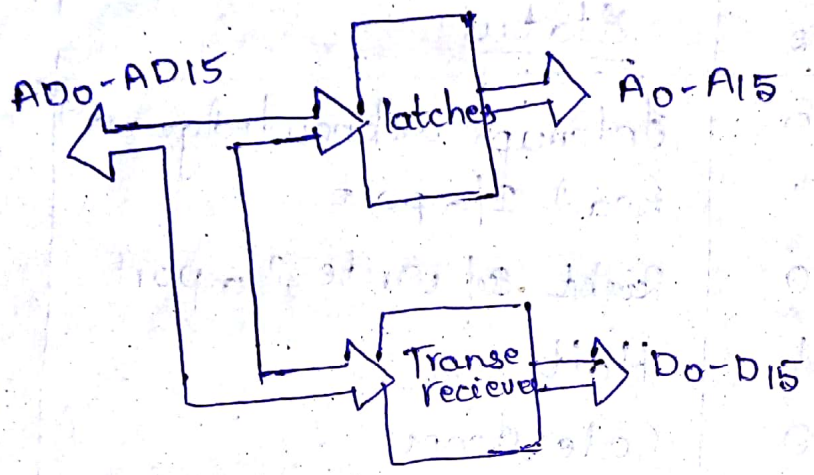
Bus request / Bus grant

General bus Operation of 8086

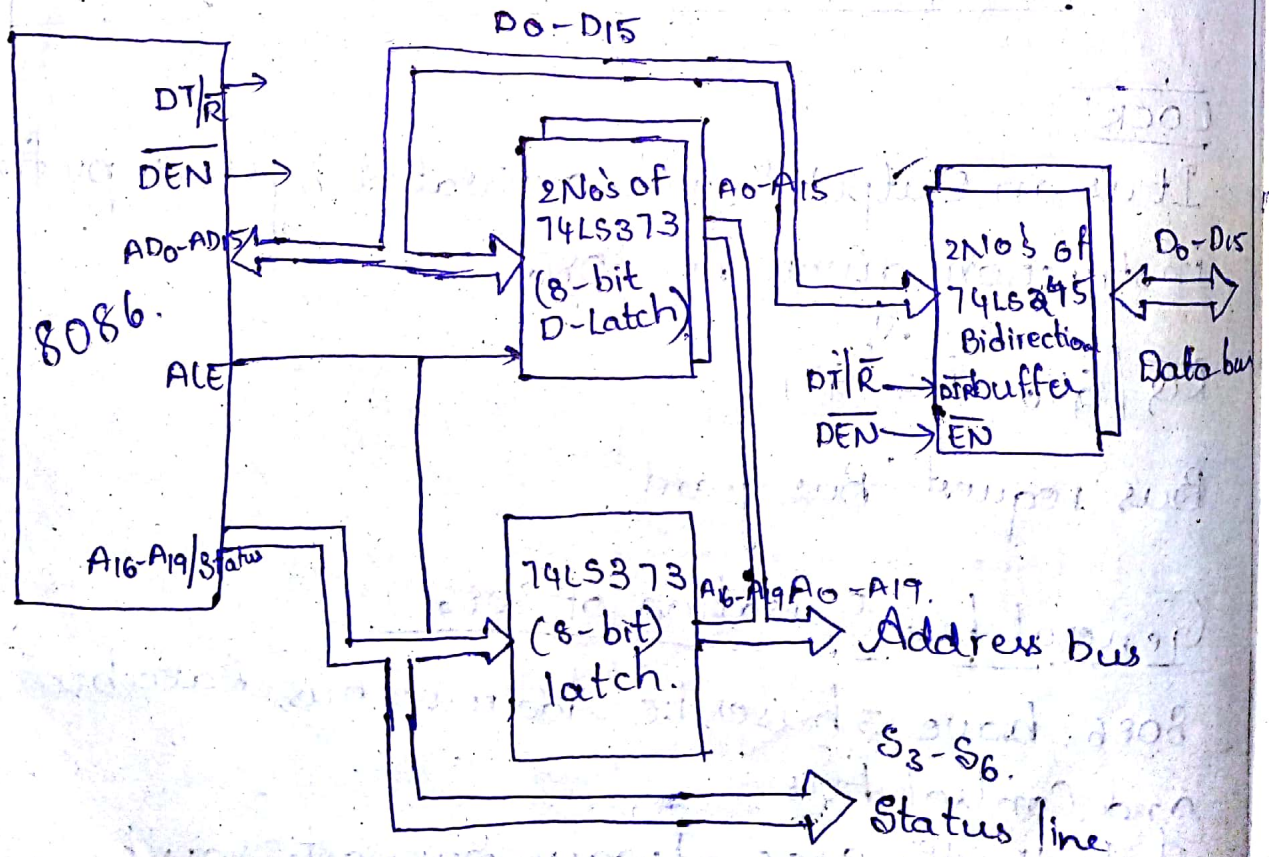
8086 have 3 buses i.e Address bus, Data bus and Control bus.

Initially the 8086 always Supports multiplexed addresses and data Signal for read and Write Operation. But pheripheral devices not necessary

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



Interfacing of 8086 with latches and transerecievers

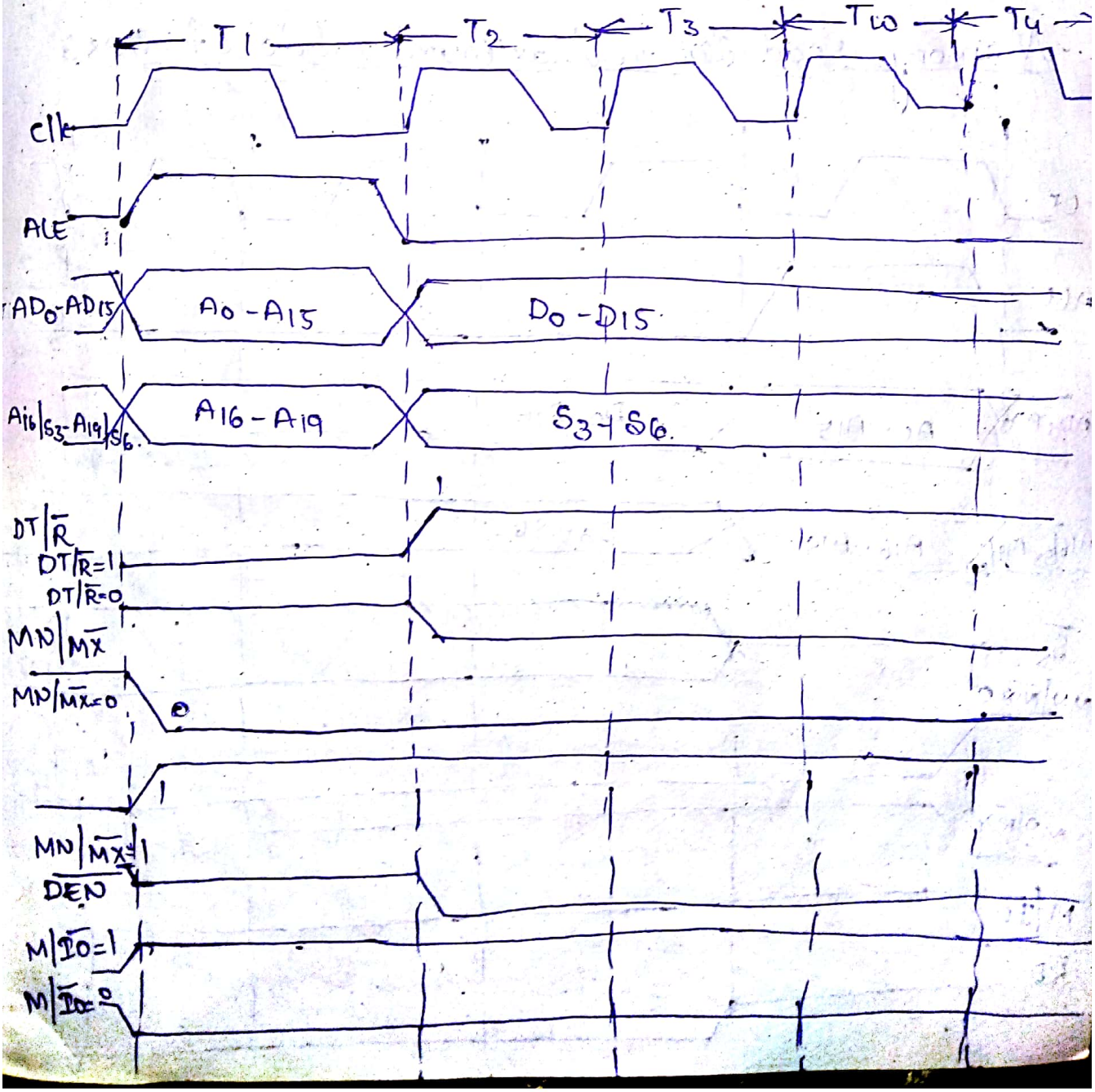


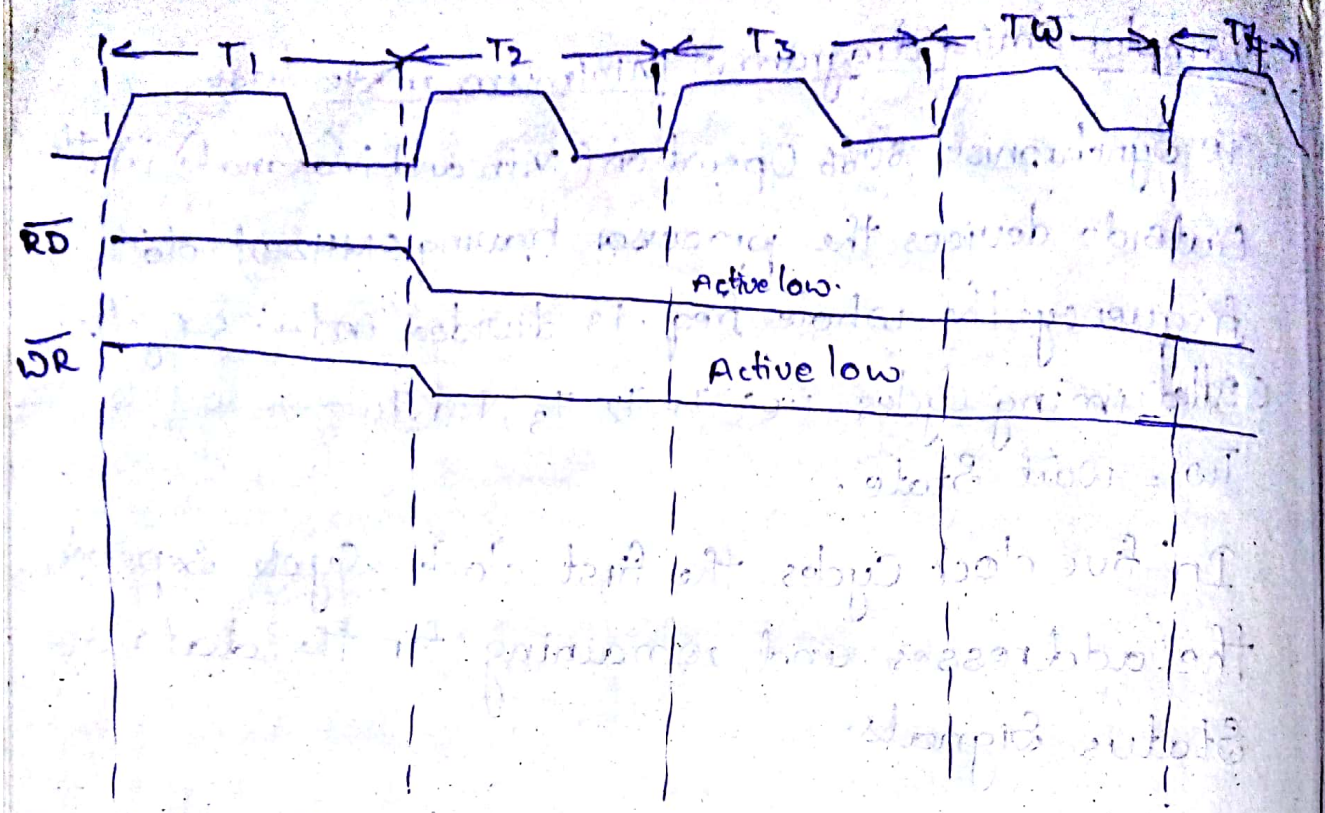
Example: Timing diagram of Minimum mode 8086

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_W, T_4.

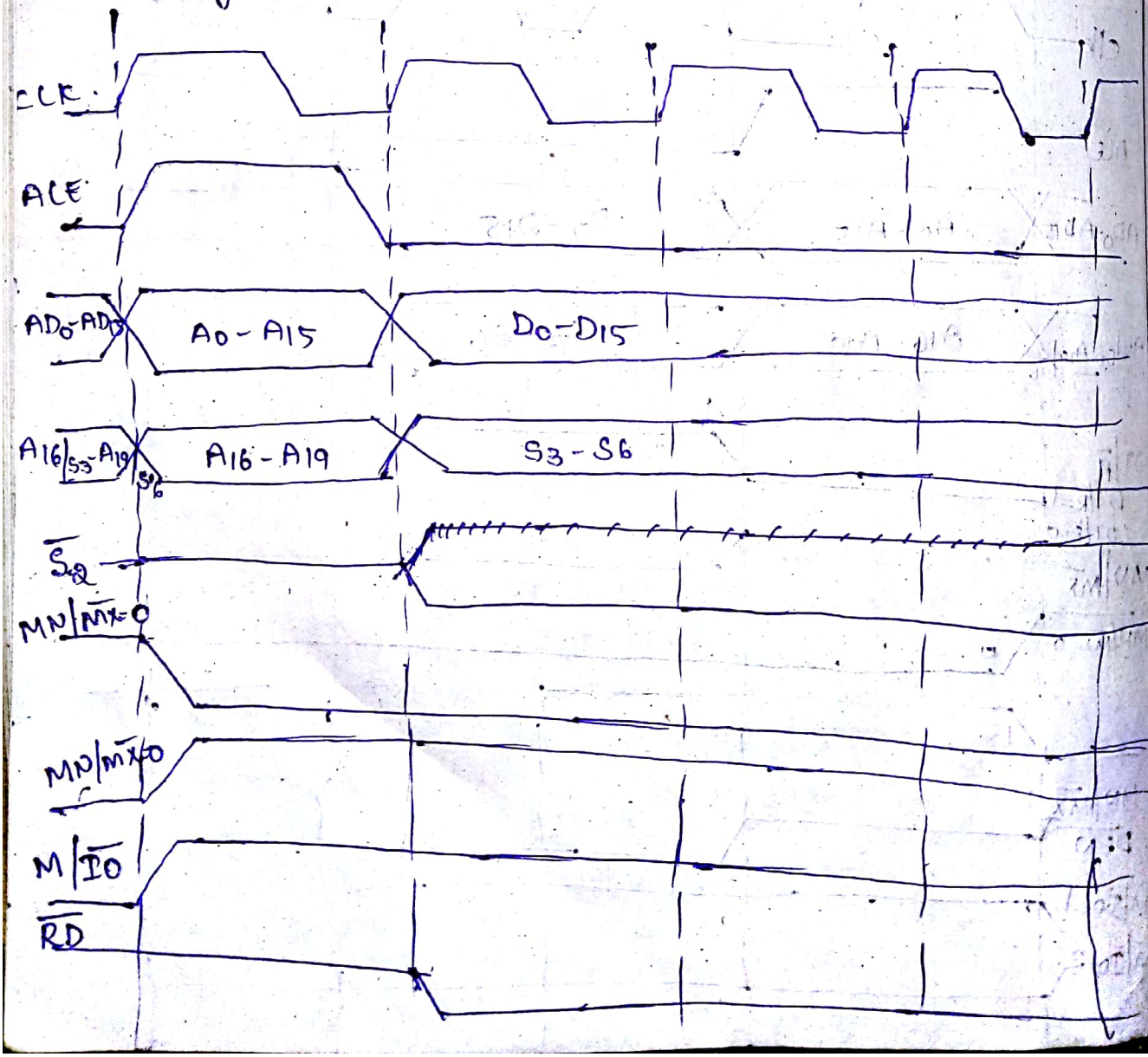
T_W = wait State.

In five clock cycles the first clock cycle seperates the addresses and remaining for the data and Status Signals.

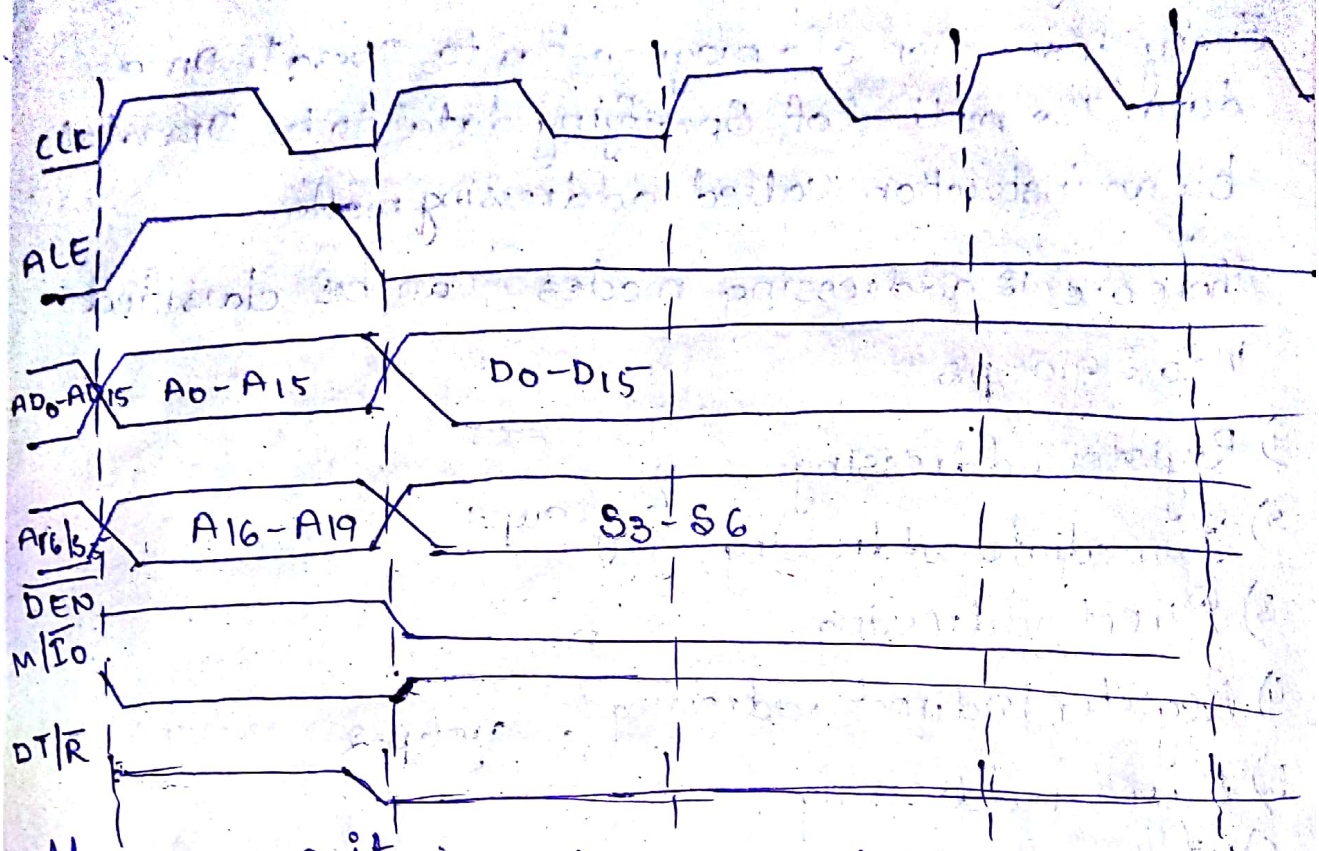




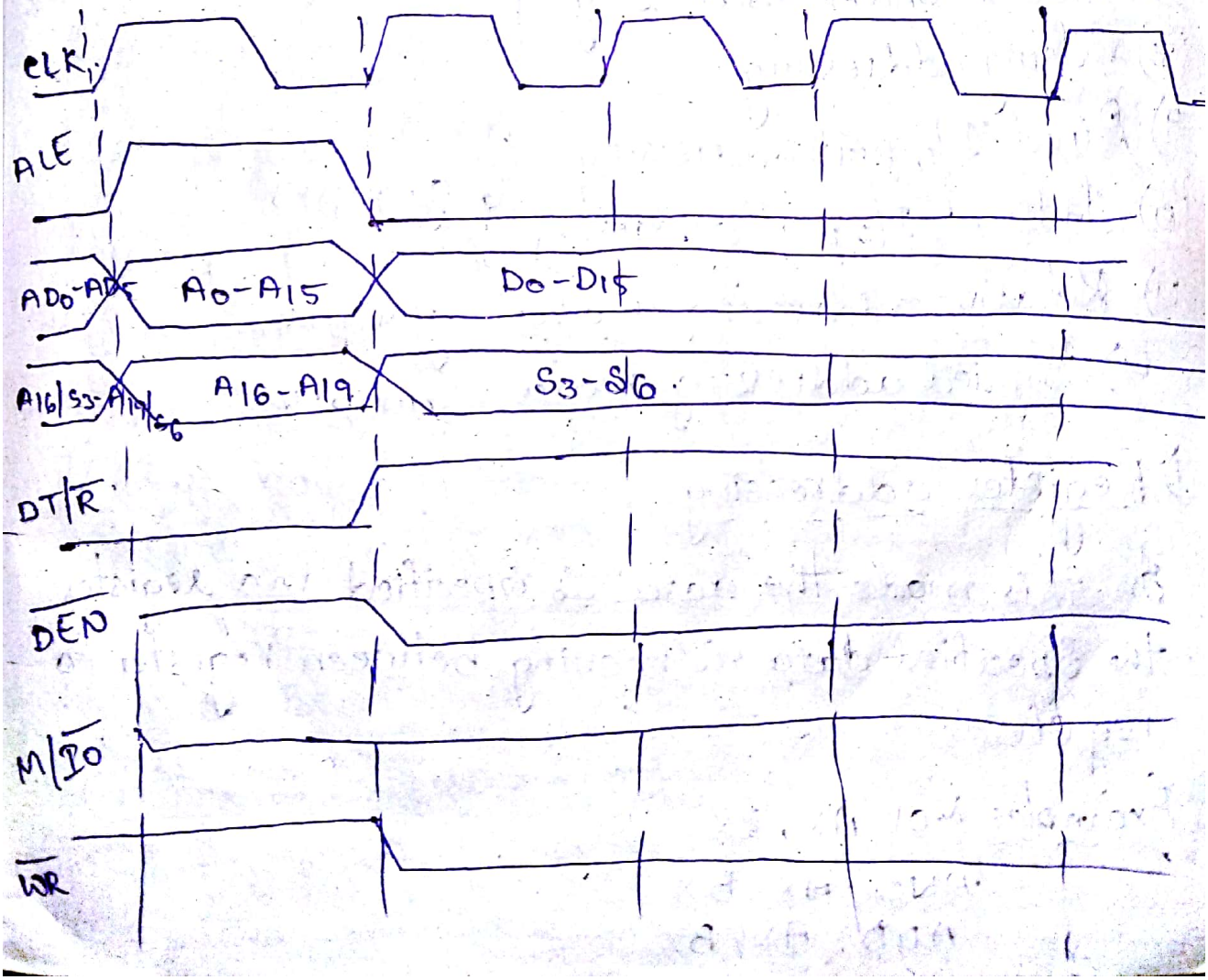
Memory read for in Maximum mode of 8086



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

1) Register addressing

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

Example: 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, $[0010]_H$

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 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.

Example:

```
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

Ex: 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 F0F0H, 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
```

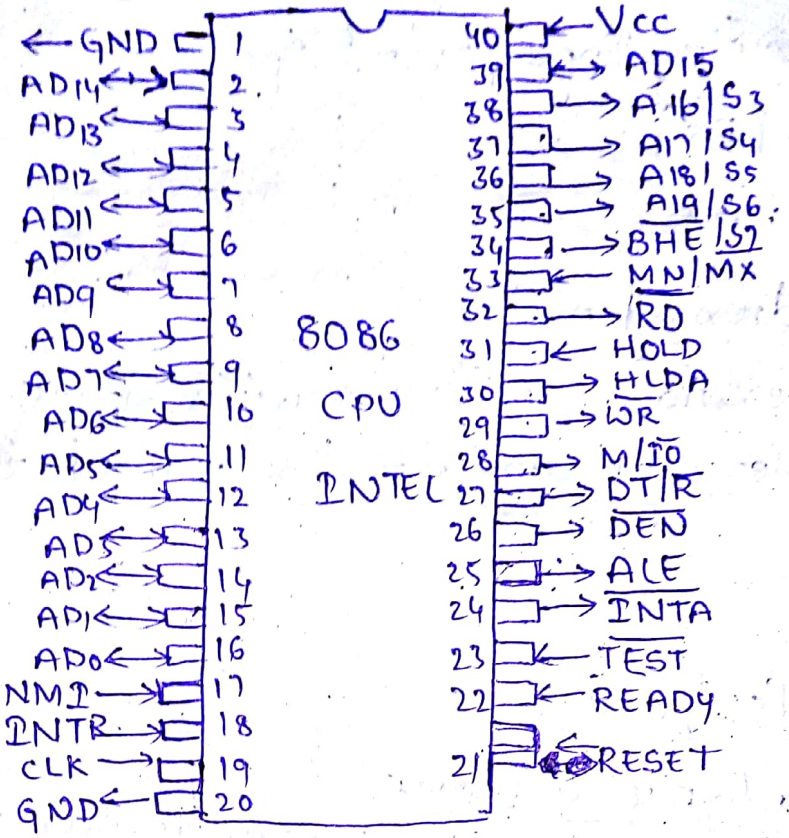
Implied addressing mode:

In this addressing mode instruction is predefined.

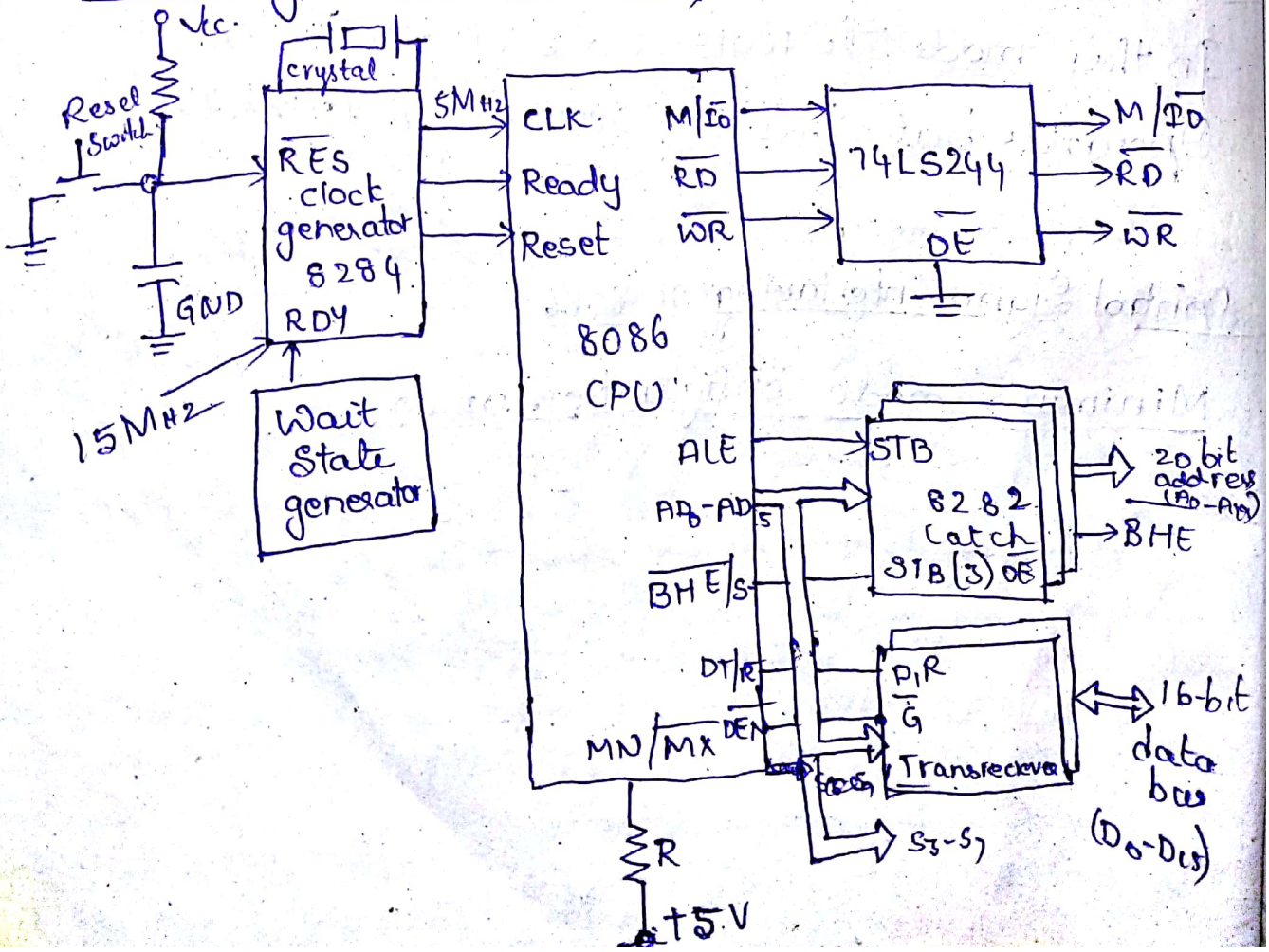
Ex: MUL BX, AAA, DAA, AAM

In this mode the register are Used for Specifying Operands but this registers are predefined.

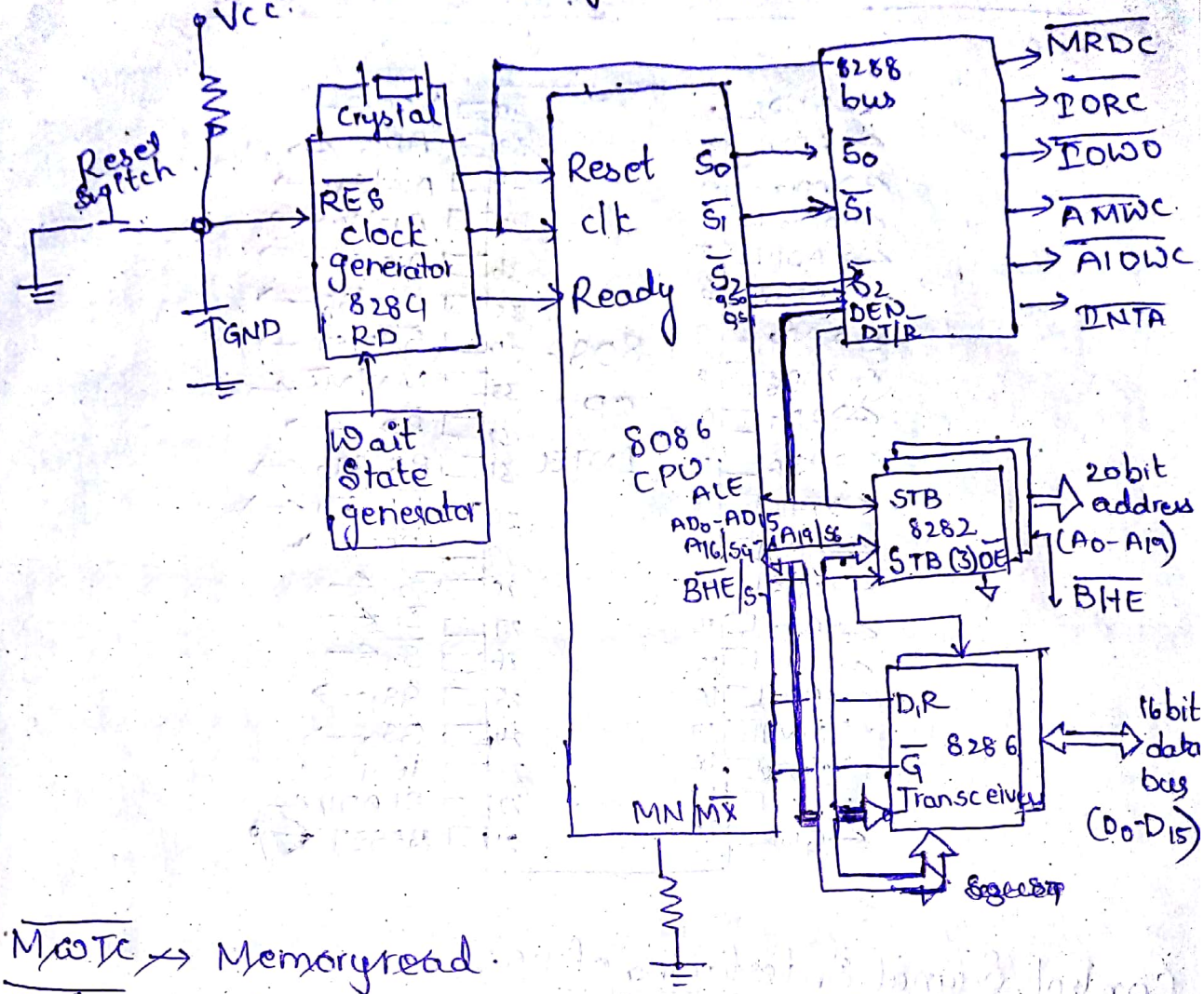
Minimum mode configuration of 8086



Interfacing of 8086 (Min mode)



Draw the Maximum mode Configuration of 8086.



$\overline{MEMR}$ → Memory read.

$\overline{MEMW}$ → Memory write

$\overline{IOR}$ → I/O read

$\overline{IOW}$ → I/O write

$\overline{MRDC}$ → Memory read

$\overline{MWTC}$ → Memory Write

$\overline{IORC}$ → Io read

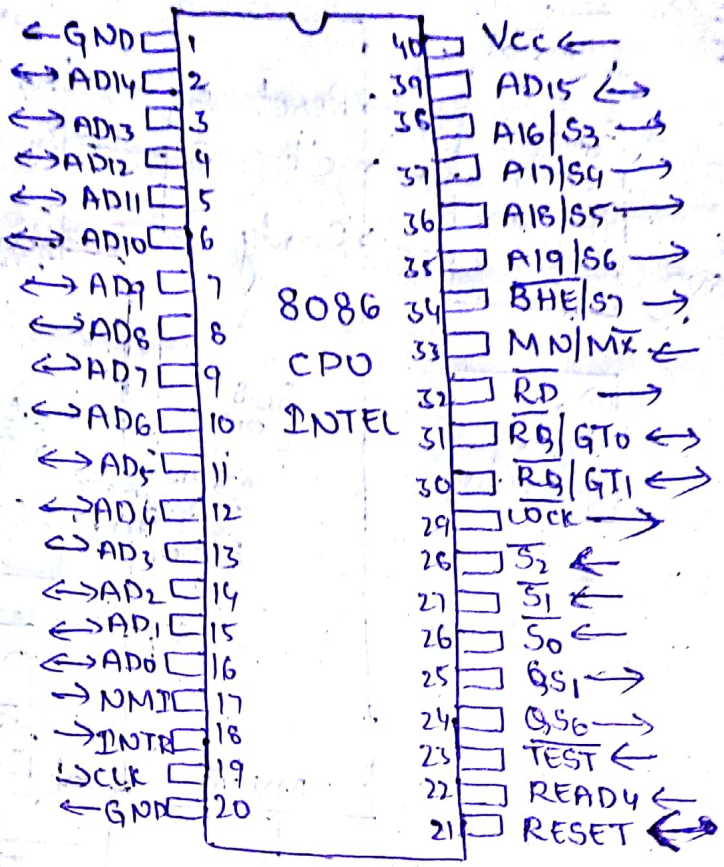
$\overline{IOWD}$ → Io write.

$\overline{AMWC}$ → Advanced memory write

$\overline{AIOWC}$ → Advanced Io write.

$\overline{INTA}$ → 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/$\overline{MX}$: 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

$\overline{BHE}/S_7$:- It is an Output Signal. If it is Set 0' it shows availability of buses. if it is 1' inactive.

$\overline{DEN}$: It is a Output Signal. This Signal is Used for Data transmitting and recieving

INTR: It is a input Signal. It is activehigh 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

$\overline{INTA}$:- 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.

$\overline{LOCK}$: It is an Output Signal. It performs predefined Operatio

DT/$\overline{R}$:- It Controls the data duection.

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

$\overline{RQ}/GT_1$:- Request and grant Signals

Advanced microprocessor 80286:

80286 is advanced Version of 8086 microprocesor and is designed for multibuses and multitasking.

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

It has two modes of memory Operation

i.e) Real mode

2) protected mode

It is designed by 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 8k bytes of Catch memory and 80387 Arithmetic coprocessor, it executes many instructions in One clocking period.

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

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

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

New Catche registers

1) Cathe 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.

UNIT 3- MICROPROCESSOR I/O INTERFACING

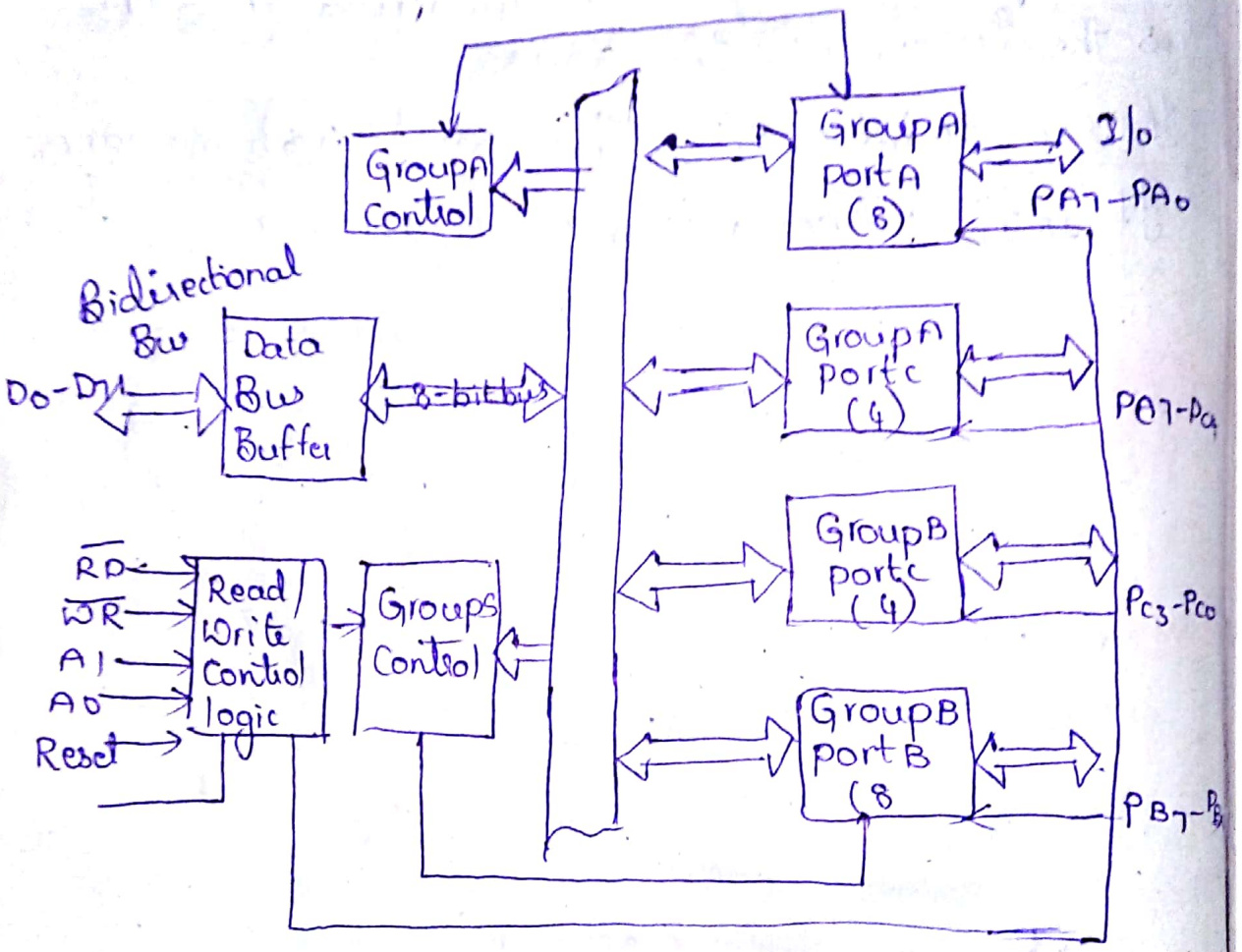
Unit-III

Interfacing of 8086

Parallel peripheral Interface 8255

8255 is input Output interfacing device or PPI

It acts as a mediator between cpu to external world



Features of 8255

1). It is a 40pin device.

2) It Contains four ports A, B, C upper port C lower

port A - $PA_7 - PA_0$

port C upper - $PC_7 - PC_4$

port C lower = $PC_3 - PC_0$

port B - $PB_7 - P_{B0}$

3. In this ports one port acts as a control port i.e port c PC_3-PC_0 and PC_7-PC_4

4. The 24 I/o lines divided into two groups i.e GroupA and GroupB

GroupA Controls the 12 I/o Signals i.e 8 lines of group A and 4 lines of GroupA port c upper.

Similarly GroupB also Controls the 12 I/o Signals

8255 Operating On two modes i.e BSR (Bit Set/Reset mode)

I/o mode

I/o mode have again 3 modes mode0 mode1 mode2

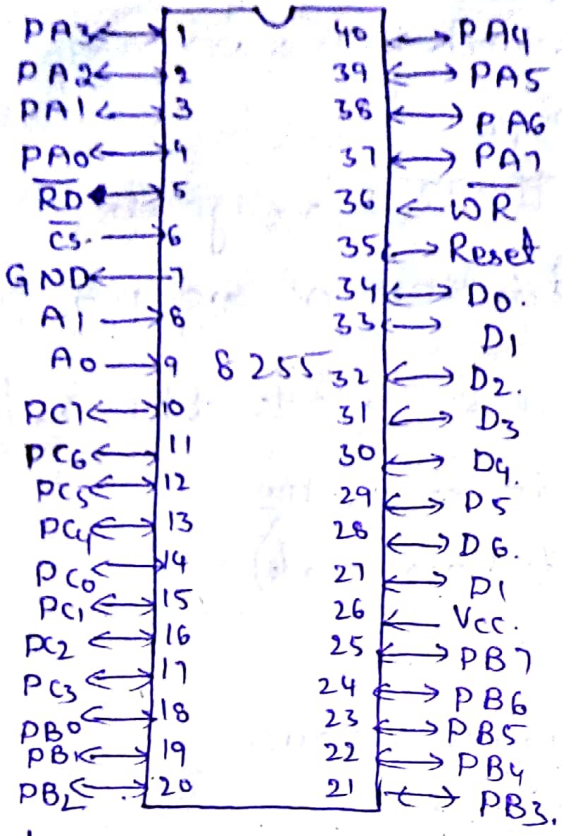
mode0 called as Simple I/o mode.

mode1 called as Hand Shaking mode

mode2 called as Bidirectional mode.

A_0	A_1	parameter
0	0	port A
0	1	port B
1	0	port c
1	1	Control logic

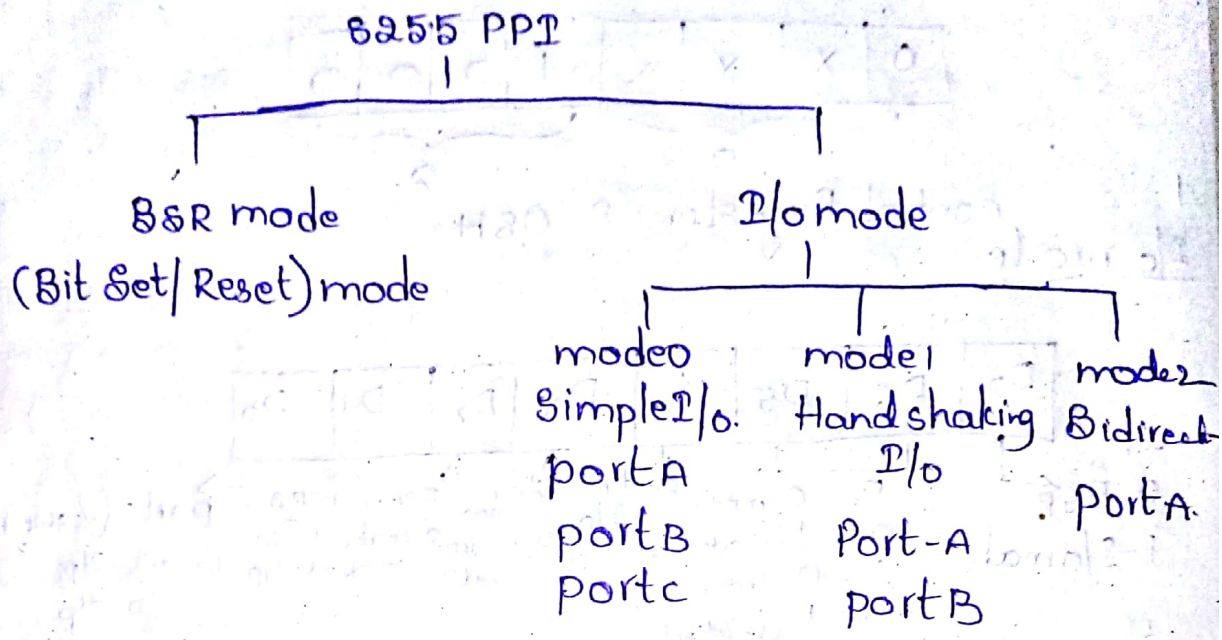
Pin diagram of 8255



Pin numbers	Description
D_0 - D_7 (27-34)	Bidirectional data lines
V_{cc} (26)	V_{cc} +5V Supply (Input)
GND (7)	It is a Output pin. It is used to measure the Signal.
A_0, A_1 (8, 9)	port Selection.
$\overline{RD}$ (5)	Read, It is an input Signal to 8255
$\overline{WR}$ (36)	Write
$\overline{CS}$	chip Selection. The Signal is coming from the decorder
RESET	Reinitialising the data.
PA7 - PA4 (37-40)	Bidirectional 8 I/O lines
PB0 - PB7 (18-25)	8 Bidirectional I/O line
PC0 - PC7 (10-17)	8 portc I/O lines

Modes of 8255

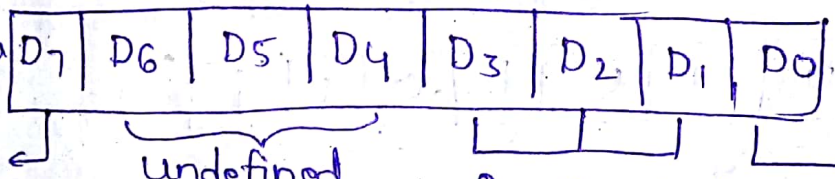
classification of 8255 modes



BSR mode:

In BSR mode Only portc is Used for Data transfering

CR Control Register



D7: 0-BSR, 1-I/O

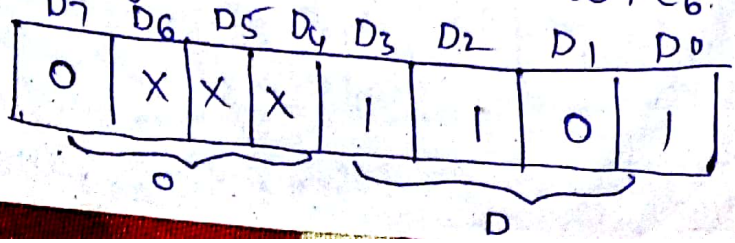
D6 D5 D4: undefined

D3 D2 D1:

0	0	0-PC0
0	0	1-PC1
0	1	0-PC2
0	1	1-PC3
1	0	0-PC4
1	0	1-PC5
1	1	0-PC6
1	1	1-PC7

D0: 1-I/P, 0-O/P

Calculate the port Value of BSR mode in 8255 where the input Signal is Connected to PC6.



CR = 0D.

The micro processor of 8086 generates data from SRAM to a monitor through portc of 4th line.

0	x	x	x	1	0	0	0

0 8

control Register CR = 08H

I/O mode

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

D_7: 0-BSR, 1-I/O mode

D_6 D_5: 0 0-mode0, 1-mode1, 1 0-mode2, 1 1-undefined

D_4: portA 1-I/P, 0-O/P

D_3: portc upper 0-mode0 1-mode1

D_2: portB 0-mode0 1-mode1

D_1: portB 1-I/P, 0-O/P

D_0: portc(lower) 1-I/P, 0-O/P

Calculate the control mode register of 8255 where portA Handshaking acts as an input in mode1 and port B Output in mode0 and portc is input

1	0	1	0	1	0	1	1

A B

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	1	0	0	1	1	1

E 7

D_0 - Input portc lower

D_1 - Input port B

D_2 - mode1

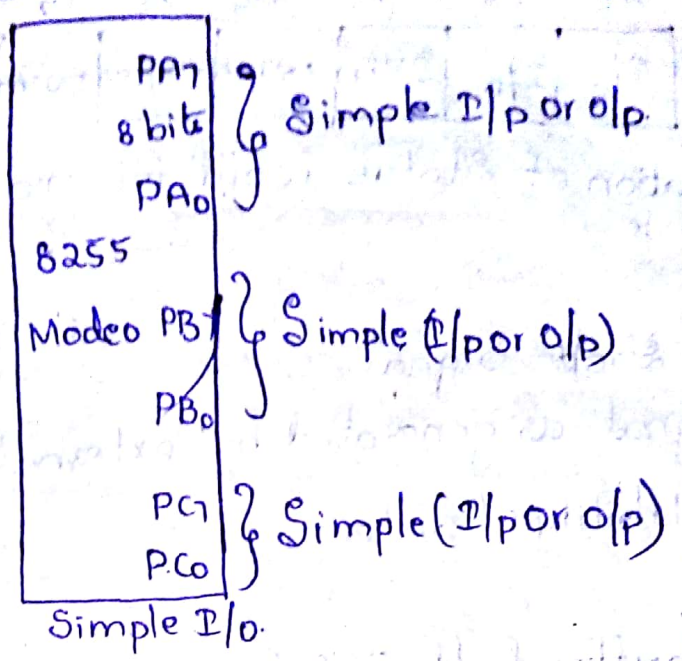
D_3 - portc Upper (Output)

D_4 - port A Output

D_6, D_5 = Undefined.

D_7 - I/O

Mode 0 Configuration



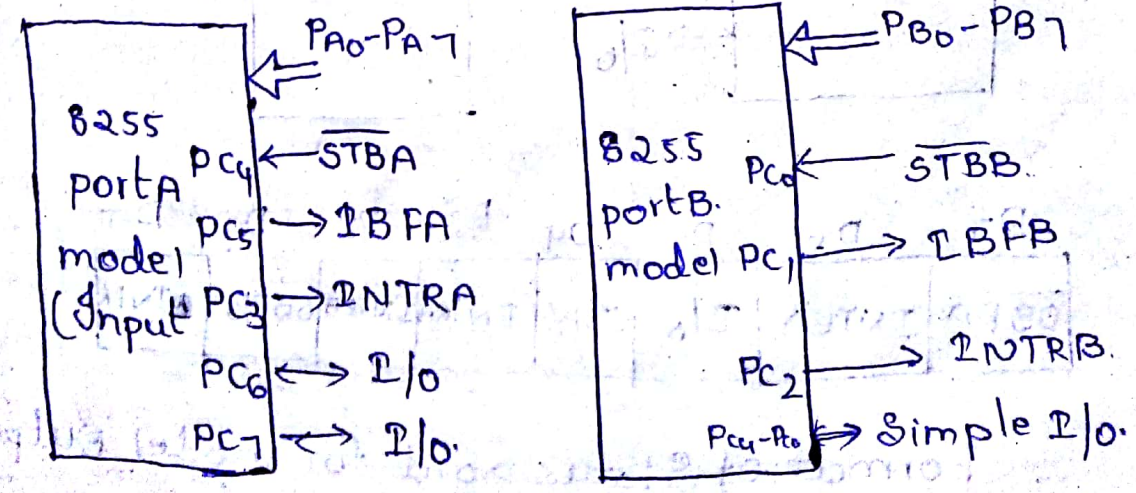
In mode 0 8255 can access any one of the port (port A or B or C) for simple input output operation

Mode1 Configuration of 8255 used for data transfering between I/o devices through 8086.

In this Configuration the data is accessed by using port A or port B with handshaking signal

Mode1 Input Configuration

In mode1 Input configuration 8255 allows port A or port B for data transfer and port c is used for controlling the data transfer.



8255 mode1 configuration

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CWR (01)	I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB

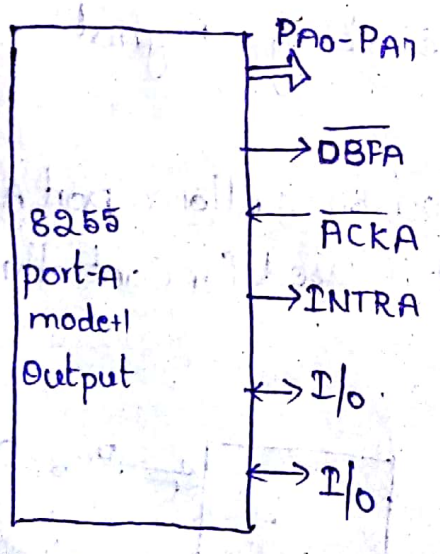
Formation of Status word for mode1 input Operation.
Status registe.

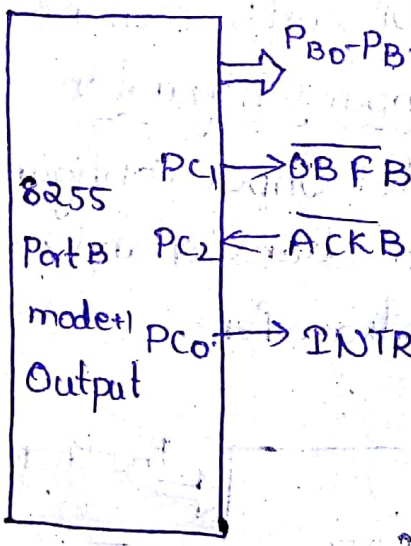
$\overline{STBA}$ - Strobe Signal
This Signal is generated by external I/o devices, indicate the availability of data

IBFA
Input buffer full. This Signal shows that the buffe is full

INTRA
Interrupt request Signal.

Mode1 Output Configuration.





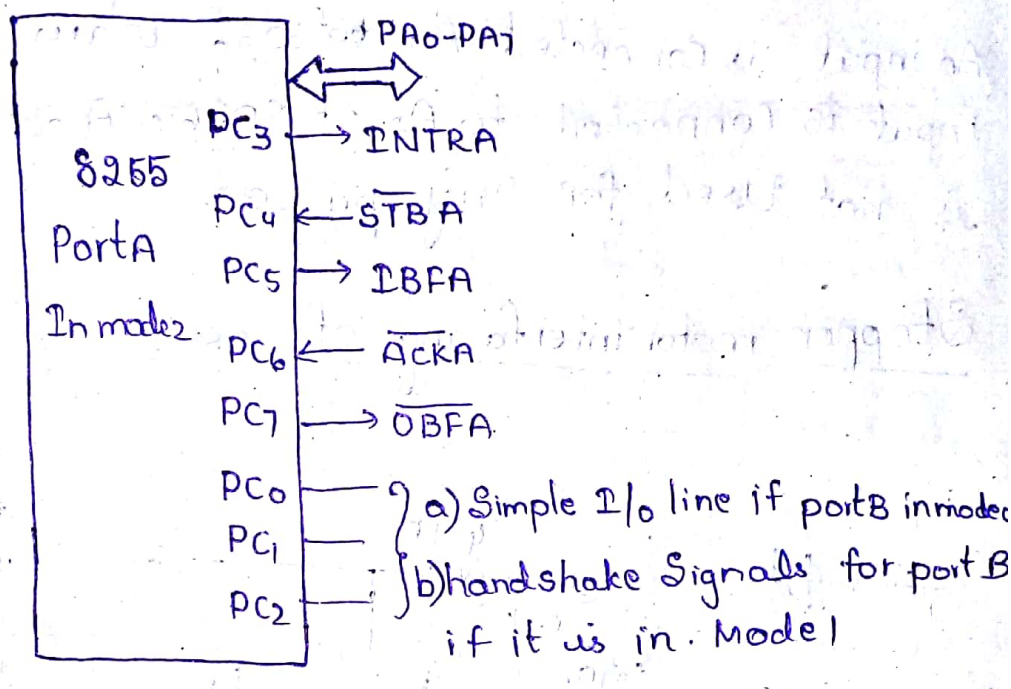
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
OBFA	INTEA	I/O	I/O	INTRA	INTEB	OBFB	INTRB

Format of Status word for mode1 Output Operation.

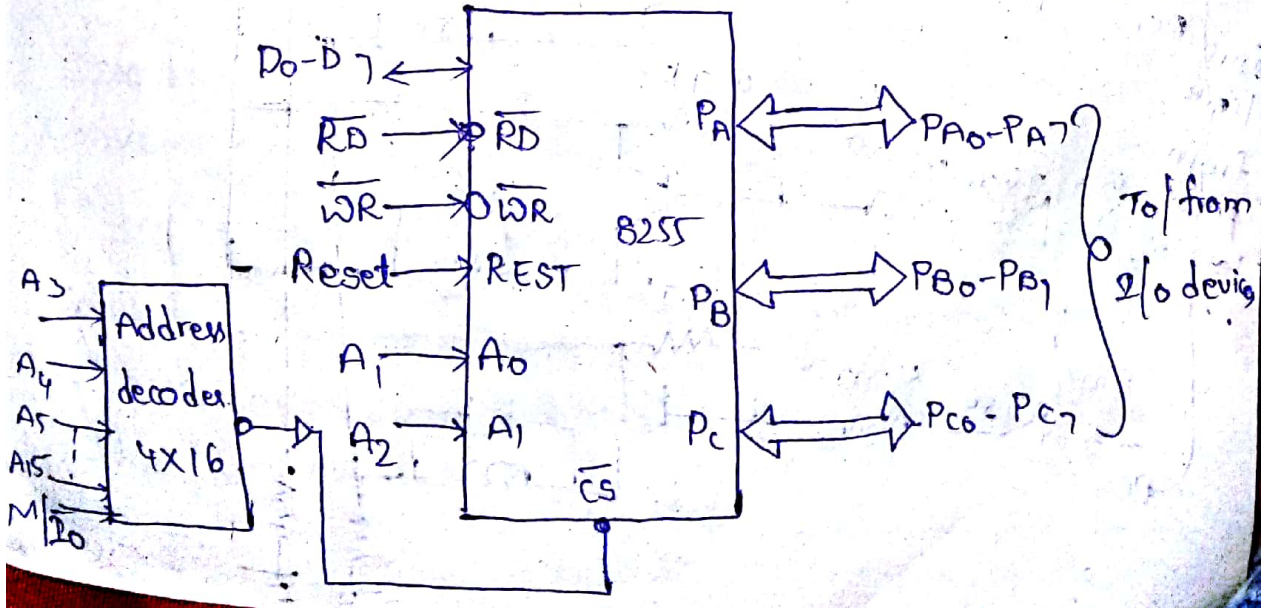
Mode 2 Configuration

In this mode bidirectional data transfer by using port A that means transmitted and recieved data in same lines.

Here port A can be used for data transfering and 5 bits of port c used for hand shaking.



Input Output interfacing of 8255 with 8086



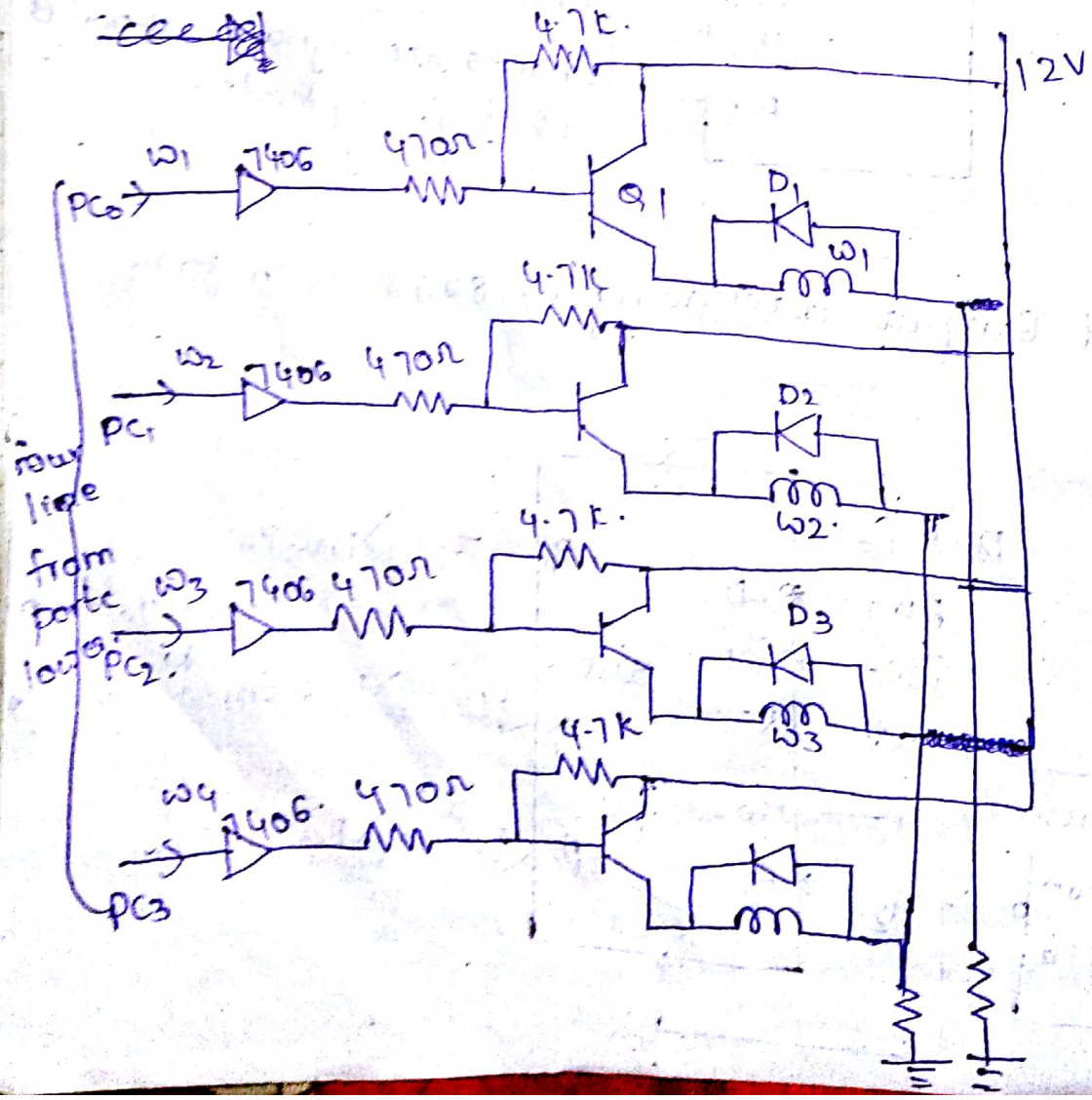
$\overline{CS}$ = chip Selection.

A_1	A_0	Operation
0	0	port A
0	1	port B
1	0	port C
1	1	No operation.

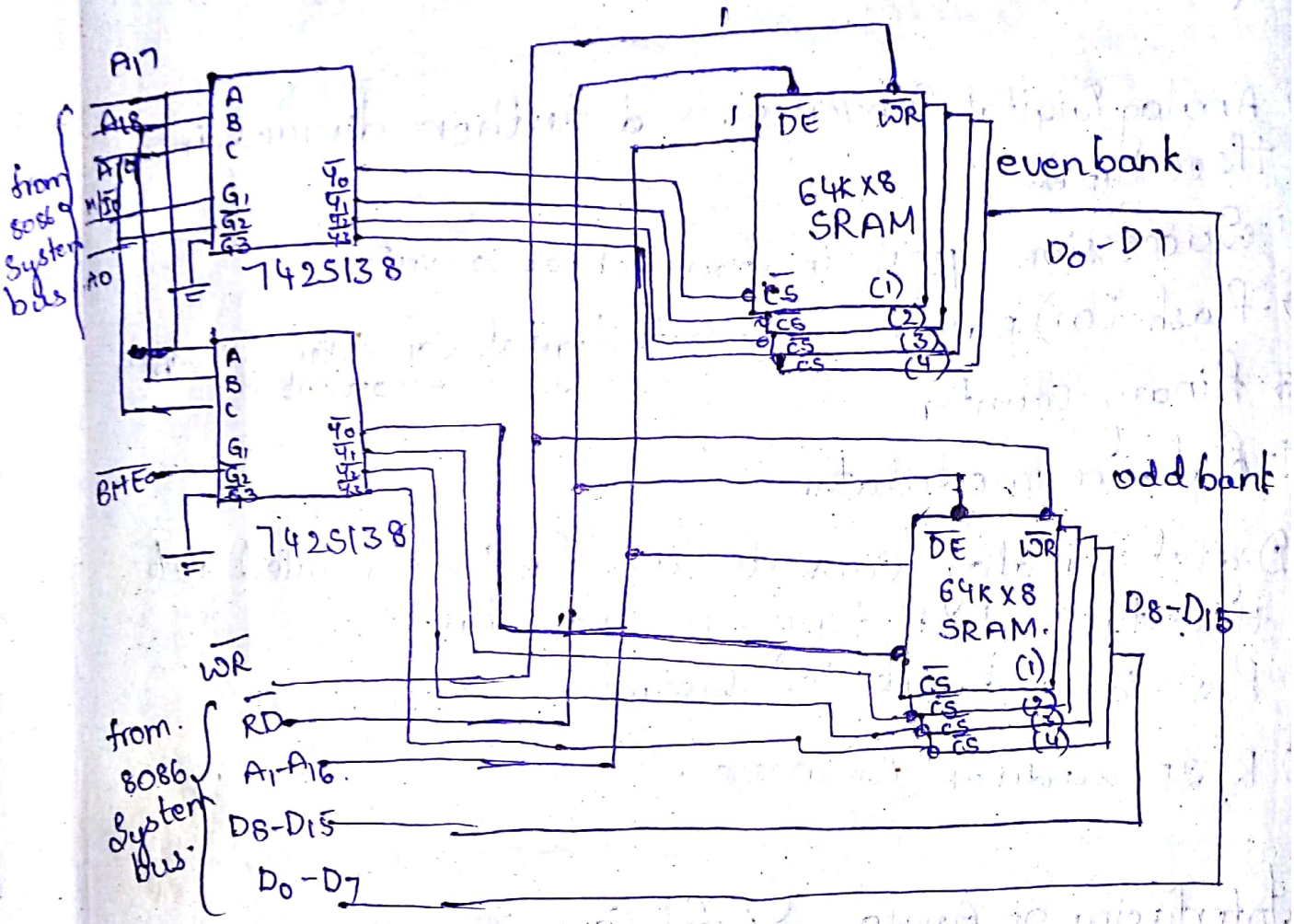
A_0 input is Connected to A_1 of 8086 where A_1 input is Connected to A_2 of 8086. A_0 of 8086 is not Used for any purpose.

from 8086 System bus

Stepper motor interfacing of 8086.



Static memory interfacing



Converter:

Converter converts one form of energy into another form

Example: transducers

Converters are two types

In microprocessor technology converters are 2 types

1. Analog-Digital converter
2. Digital-Analog Converter

Analog Signal:-

A Signal changes w.r.t time changes is called Analog Signal

Digital Signal:-

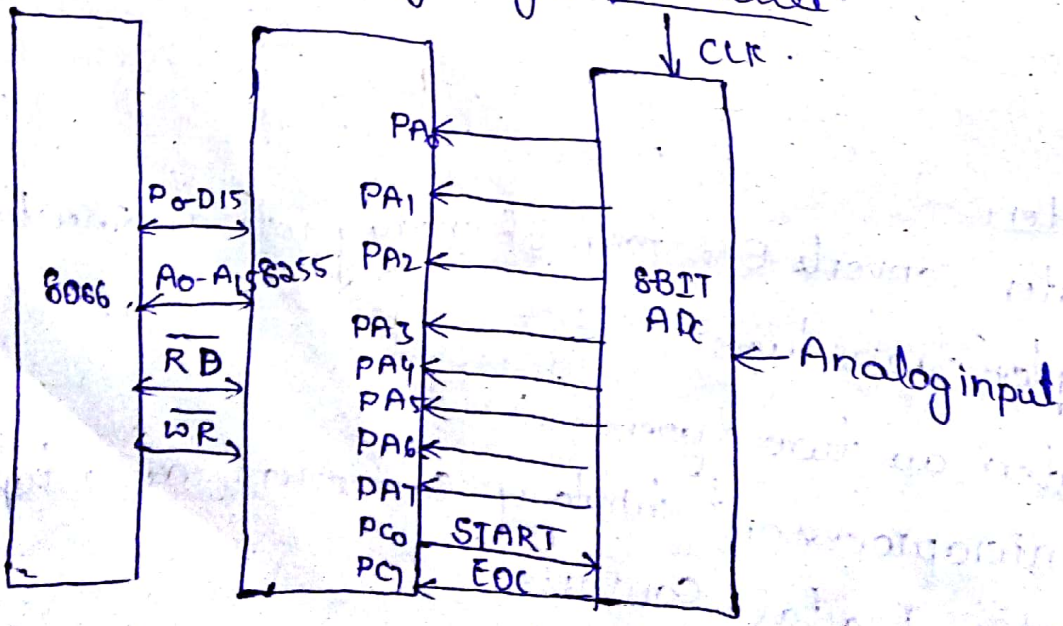
A Signal represented in binary form is called as Digital Signal.

1) Analog-Digital Converter is further divided into three types.
   1. Successive approximation. - cheap converter
   2. flash (or) parallel. - fastest digital converter
   3. Binary counter - costliest
   4. Balance modulator

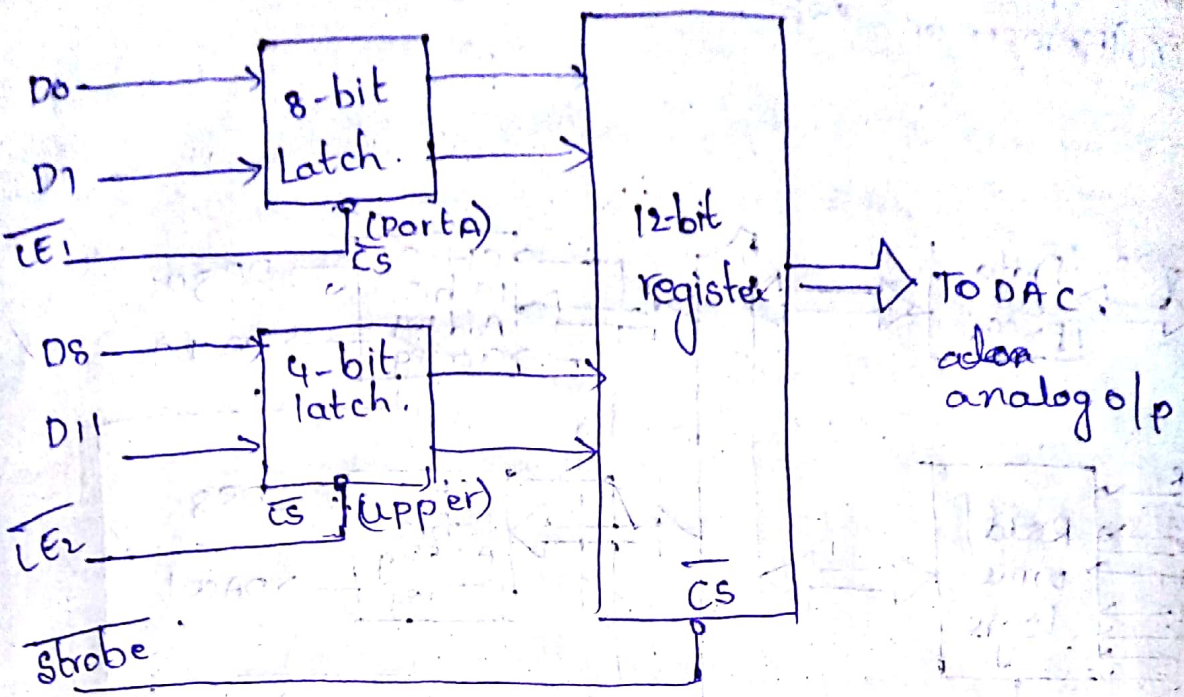
2) Digital-Analog Converter is further divided into three types. but two are important
   1) Resistive network (or) weighted resistor
   2) R/2R. Ladder Network.

10

Interfacing of Analog-Digital Converter



Latches for 12-bit D/A Converter



8255 is having two ports
Latches for the Output
Buffers for the input

6)

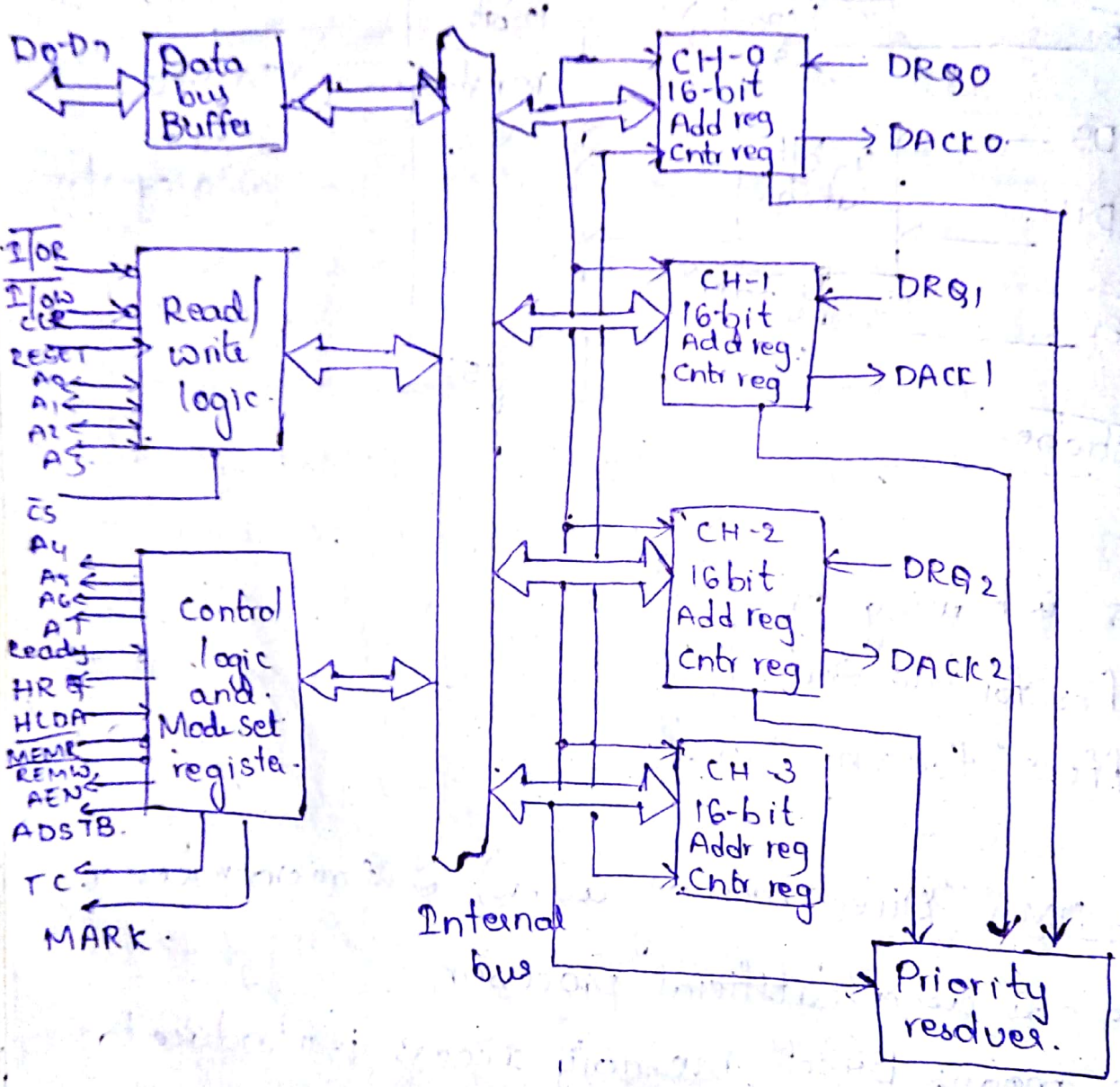
8257 DMA (Direct memory access) Self microprocessor

1) DMA acts as a artificial processor
2) DMA means Direct memory access is introduce to Speed up the data transfer b/w CPU and peripherals
3) Normally in a programmed I/o data transfer an interrupting driven happen by CPU (without CPU the data willnot be transfered b/w peripheral to peripheral)
4) This Situation is Solved by using DMA Controller. to enable data transfer b/w peripheral devices and memory without involvement of Microprocessor.

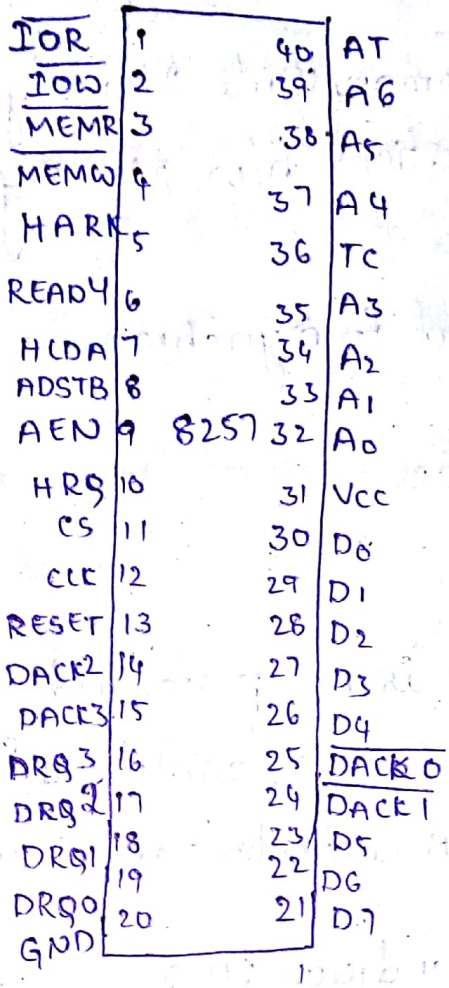
Block diagram of 8257

Single processor - 8257

Multiprocessor - 8237.



Pin diagram of 8257



8257 Supports 3 types of data transfer i.e

1. Single bit transfer
2. Double bit transfer
3. Demand bit transfer.

DMA Controller is used to provide an interfacing Memory to I/o. it also Supports I/o to I/o.

DMA is Self programmable Ic

The 8257 have different functional blocks name

1. Data bus buffer
2. Read and Write logic
3. Control logic Modeset register
4. Addresses register
5. Priority resolver.

D_0-D_7 bidirectional data Used for Read and Write operation b/w Memory to I/o and I/o memory.

I/o Read, I/o Write Memory Write Memory Read performs Read and Write operations b/w Memory to I/o and I/o memory

CLK Signal is Used to Synchronises the DMA Operation.

DMA Operates in 3 modes

1. Ideal state
2. DMA State

1. Ideal mode → DMA is not performed
2. Slave mode → It is taking Conditions from microprocessor
3. Master mode → DMA Controller takes the decision

RESET reinitialises the Condition to 0

A_0, A_1, A_2, A_3 address Signal.

These address Signals are Used to Select channels for data transfer

A_3	A_2	A_1	A_0	channels
0	0	0	0	CH-0 address register
0	0	0	1	CH-0 Control register
0	0	1	0	CH-1 address register
0	0	1	1	CH-1 Control register
0	1	0	0	CH-2 address register
0	1	0	1	CH-2 Control address register
0	1	1	0	CH-3 address register
0	1	1	1	CH-3 Control register
1	0	0	0	Control-Status register

Memory is Connected to A_4, A_5, A_6, A_7

Ready Signal will provide information about whether the DMA Controller is ready to Send the Signal.

Address Strobe ADSTB

This Output from 8257 Used to latch 8 bit data from I/O to memory device.

Memory read Memory Write

Memory read Used to read the data from I/O to Memory.

HRQ (Hold Request)

TC - Terminal Count

MARK - It is a pointer decides the Starting and ending of data

DRQ0

Registers of 8257

There are 2 registers Supported by 8257 Controller is

1) Control-Word register

2) Status word register

Control Word Register

B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
EAL	ETCS	EEW	ERP	ECH_3	ECH_2	ECH_1	ECH_0

FORMAT OF CWR

$B_0 - B_3$ Bit 0 to Bit 3 Used for enabling or disabling the channels.

Bit-4 (ERP)

Enable Rotating priority

If the bit is 0 fixed priority if it is 1 automatic prior

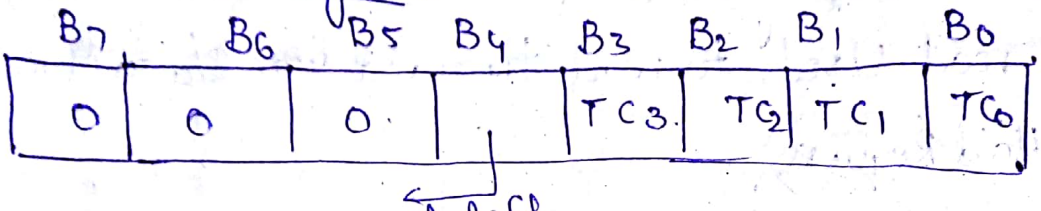
EEW - Enable extended write

ETCS - Enable Terminal Count Stop

EAL — Enable Auto load.

If the bit is 1 it performs repeated Operation.

Status word Register



FORMAT OF SWR

B0 - B3 - These 4 bits are Used to indicate Status of terminal Count of Corresponding channel.

B4 - Update flag

0 - Reset

1 - Auto reload.

Programmable Interrupt Controller 8259

8259 is an interrupt Controller Used to Call Or divert or Switch Microprocessor to User applicati PIC generates hardware Software interrupt Contr Signals for Subroutine programs

Without Interrupt Controller the buses willn't Shares bus Signals for multi tasking.

Interrupt is nothing but a task driver to executes multiple tasks in sequence.

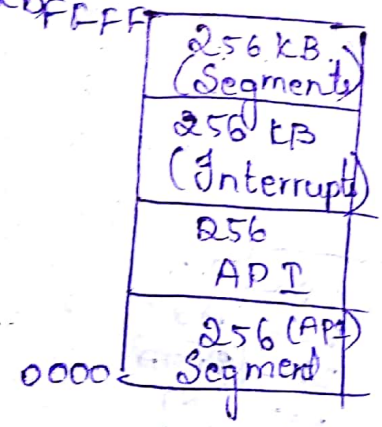
There are 2 types of interrupts namely hardware interrupts and Software interrupt

hardware interrupts:

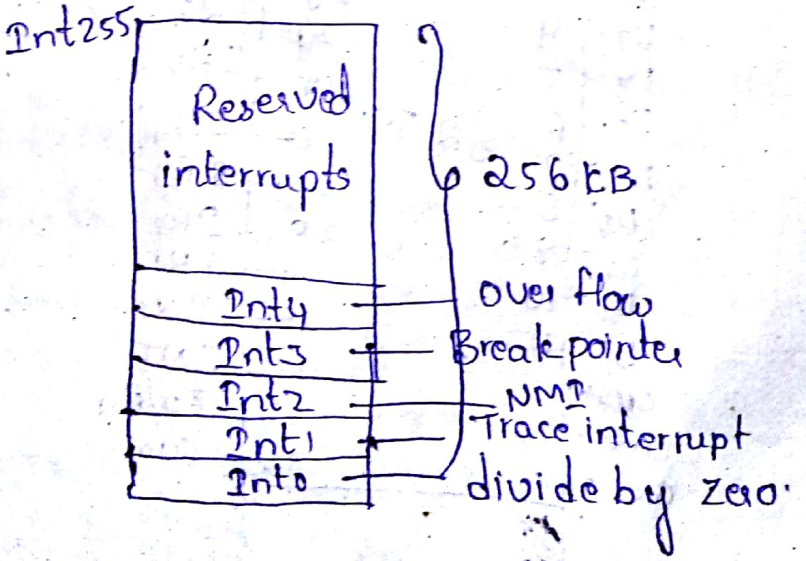
1) Maskable :- It diverts Uprocessor for Subprogram
2) Non maskable execution permanently.
2) Non maskable:- For a time it executes the Sub program and Continue the main program.

Software interrupts

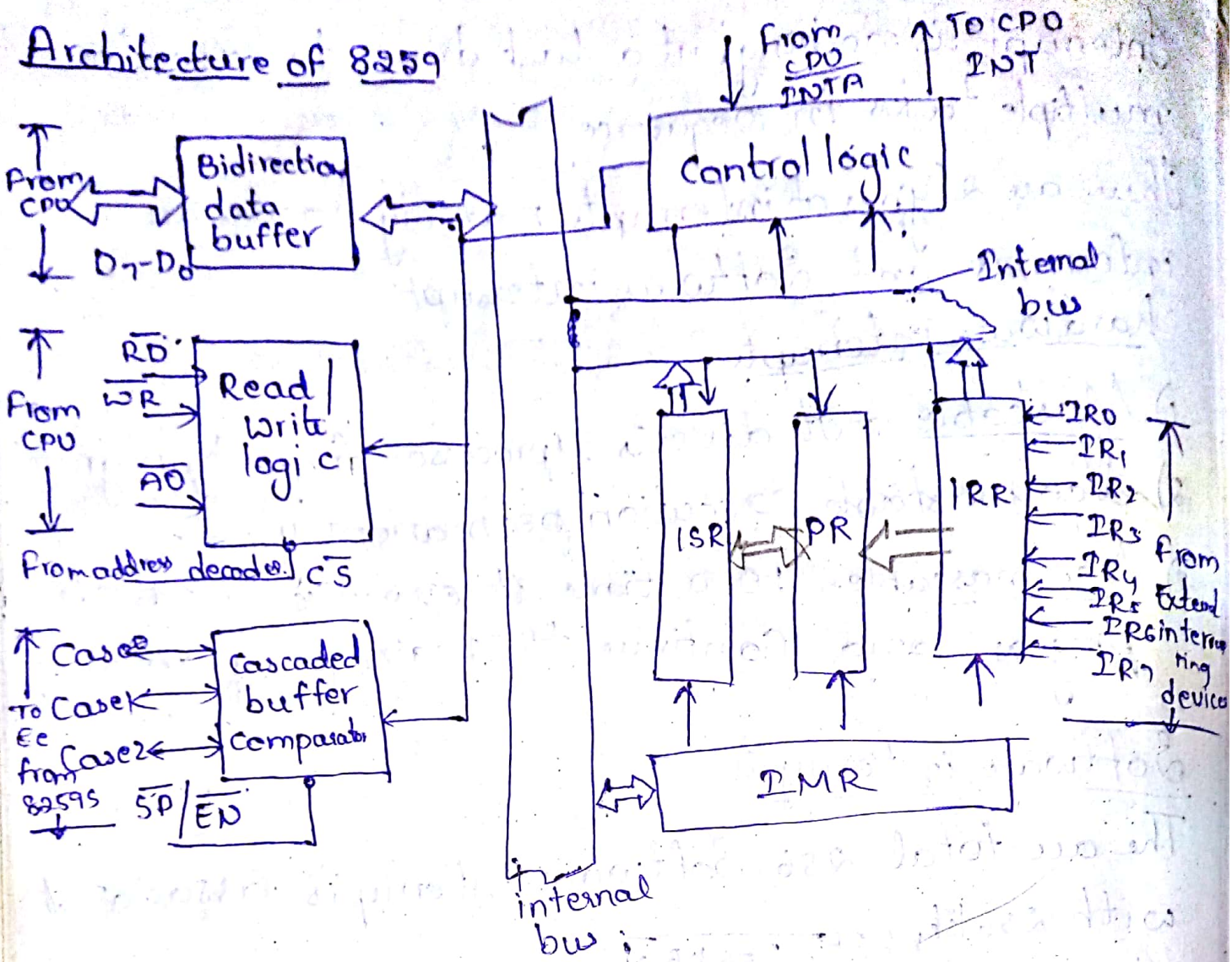
There are total 256 Software interrupts associated with 256 kb



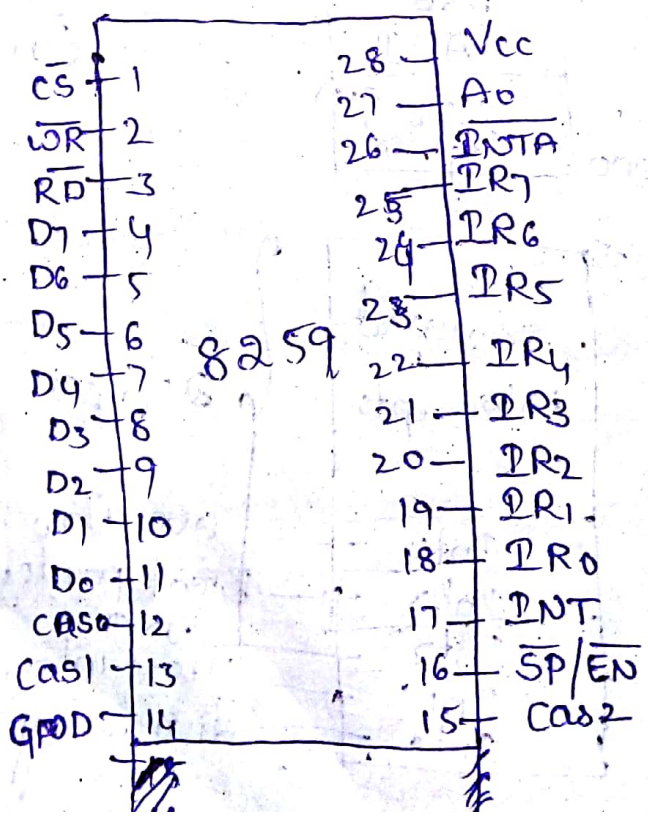
API - Application Specified Interrupt



Architecture of 8259



Pin diagram of 8259



8259 can be divided into 8 functional blocks.

Bidirectional data buffer:

Read and Write logic:

It controls the data transfer.

$\overline{A0}$:

It is an input signal to 8259 to enable the address signal

$\overline{CS}$:

chip selection.

cascaded buffer of comparator:

This block provides interfacing b/w one 8259 to another 8259 (or) master 8259 to slave 8259 to expand the interrupts

Control logic block:

Controls the timing and activities of other blocks

IRR Interrupt request register.

This block accepts 8 interrupts from external devices.

PR: Priority resolver

It determines the priorities of active interrupt and decides which interrupt should be service and when.

There are different priority schemes which can be selected by software

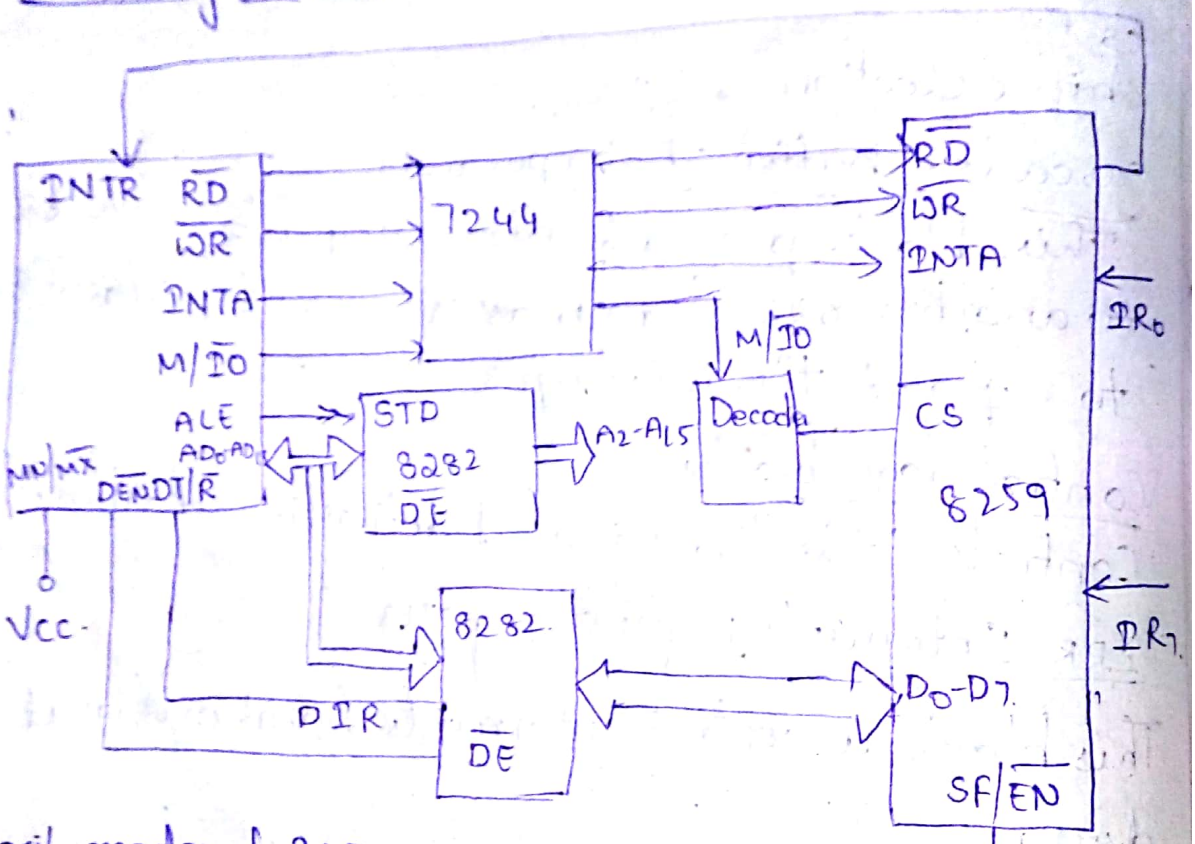
1. fixed priority $IR_0 - IR_7$
2. ISR - Service register.

This register keeps track of which interrupt is being Serviced Currently

IMR Interrupt mask register

This register is Used to mask or disable the interrupt request.

Interfacing of 8259



Priority modes of 8259

8259 get many modes to determine the priority of interrupts. Those are Controlled by Software Control Can be changed dynamically

fully Nested mode (or) Default mode

In this mode the highest priority Said by IR0 and IR7 has lowest priority

2) Specific rotation mode:

In this mode the priority is assigned to lowest priority and then its interrupts takes in Cyclic manner.

UNIT 4-8051 MICROCONTROLLER

Unit-V

Microcontrollers 8051

Introduction of 8051 micro controller

Microcontroller: Micro Controller is a Single purpose programmable Semiconductor Ic. It includes microprocessor IO ports, timers, counters, interrupts, memory fabricated into a Single chip.

It is happened because of VLSI technology (Very large Scale Integration technology)

Micro controllers are available from 4 bit to 32 bit

Ex: 8051 - 8 bit microcontroller
PIC - 16 bit micro controller etc

Important features of micro Controller (8051)

1) Clock Speed is 11.059 MHz to 20 MHz
2. It is having two general purpose registers name A & B
A - Accumulator register
B - Base register

and also have Special registers They are
$R_0, R_1, R_2, R_3, R_4, R_5, R_6, R_7$

The father of microcontroller is 8051 Or 8031

It holds 8 bit CPU (8085)

It has 8 bit Stack pointer, 16 bit data pointer

It Supports two 16 bit timers (T_0 and T_1)

It has 4 IO ports They are
port 0
port 1
port 2
port 3

It also includes power Controller register of 8...

It has two extanal interrupts. They are

i) NMI

ii) INTR

10. It has three internal / two external interrupts. They are

i) LCall - long Call (It permanently Stop the program)

2) SCall - short Call

3) retun

11) In micro Controller inbuilted memory

128 bytes of internal RAM

4 KB of internal ROM

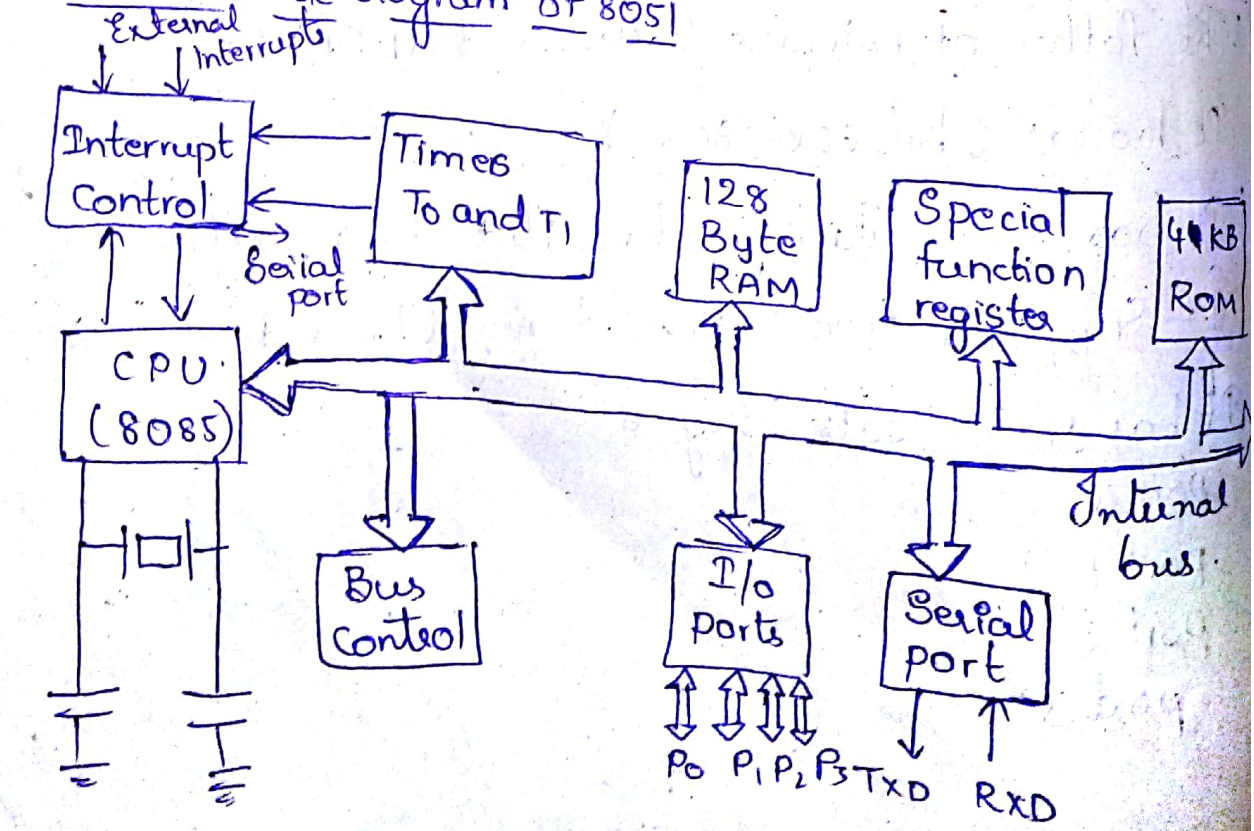
12) It has 8-bit program Status bar.

It is Operated in full duplex mode

i.e All ports are bidirectional mode

It has 8 bit data lines and 6 Address lines

Functional block diagram of 8051



→ 8051 is the hardward architecture means seperated memory for program code and data.

→ In this CPU is nothing but microprocesor. It holds the instructions.

Interrupt controlr

→ It is used for the execution of the program. INTR, NMI.

→ A part of 8259 is inbuilt in 8051 microprocontroller

→ Timers and Counters are used to speed up the Operation Or to Synchronise the Operation

→ Here the timers are 16 bit timers.

→ Again the timers are divided into lower and upper.

Special function registers

Here registers are used to hold the data temporarily and directs the data from various location to the buses.

Buses are 3 types

1. Addressbus
2. Data bus
3. Control bus.

Bus is an interfacing channel to transfer the data

Here the communication is Serial and hence it is full duplex.

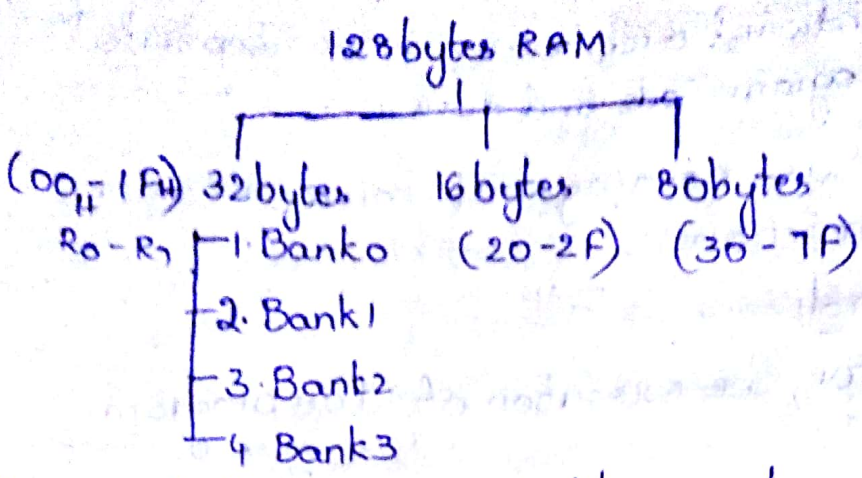
The 8051 supports 10 registers

A, B, R0-R7

A - Accumulator

B - Base.

R0-R7



→ R0 - R7 is the programmable registers

→ Accumulator and base registers are Special purpose registers.

→ 32 bytes of memory banks

→ 16 bytes of memory for addressing purpose.

→ 80 bytes of memory for Special programming purpose.

→ 8051 has 4KB of ROM.

Serial port:

It performs Operates ~~Serial port~~ two types of Communication.

1. Serial Communication - (Serial port)
2. Parallel Communication - (I/O port)

Serial port:

It is Used to increase the Speed of Operation. with Various. baudrate.

TXD - Transmitter.

RXD - Reciever.

I/O ports:

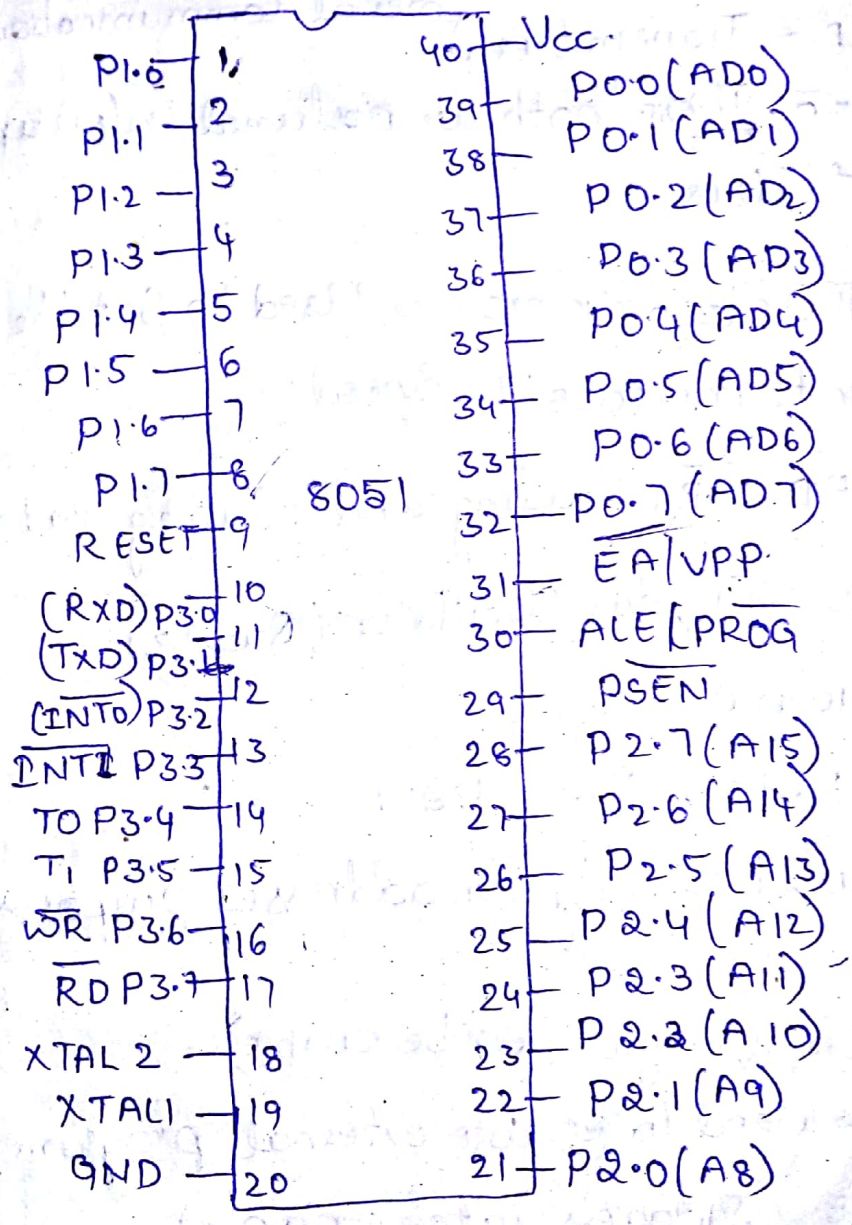
It is ~~bi~~ bidirectional. Input Output data ports

Here I/o ports are 4 ports

1) Port0 - P0.0 - P0.7.
2) Port1 - P1.0 - P1.7
3) Port2 - P2.0 - P2.7
4) Port3 - P3.0 - P3.7.

Each port is 8 bit bidirectional port.

Pin diagram of 8051



8051 is available in 40 pin dual in line package.
Each port is 8 bit port
Pin1 to Pin8. P1.0 to P1.7 (bidirectional port I/O pr

These 8 lines are used by port 1 of I/o purposes

Pin9: Reset - Reinitialisation of micro Controller - (Input)
Input to 8051

Pin10 - 11, 12, 13, 14, 15, 16, 17 - P3.0 - P3.7.
These are multiplexed I/o with Special functions
Used by port 3 of 8051

Pin10: RXD - Recieved. } Serial Communication
Pin11: TXD - Transmitter. }

Pin12: $\overline{INT0}$, $\overline{INT1}$ These both are external Interupters given by the User.

T0, T1 These two timers are used to set the time and also to increase the speed.

Pin16,17 $\overline{RD}$, $\overline{WR}$ (Reading and Writing data)

XTAL1 Pin18. Crystal Oscillator. (Quartz)

Pin20: Ground.

Pin21-28 Port2 P.2.0 - P.2.7.
This port is used only for addresses purposes ~~the data~~

Pin29 $\overline{PSEN}$ (Program Strobe enable)
This pin is used to enable external program

Pin30 ALE (Address latch enable)
To Seperate address Signal with the data.

Pin31 ($\overline{EA}$/VPP) $\overline{EA}$=0.
Enable-EA This Signal is used to Select the ROM Addresses EA=1 (~~ROM~~ EPROM)

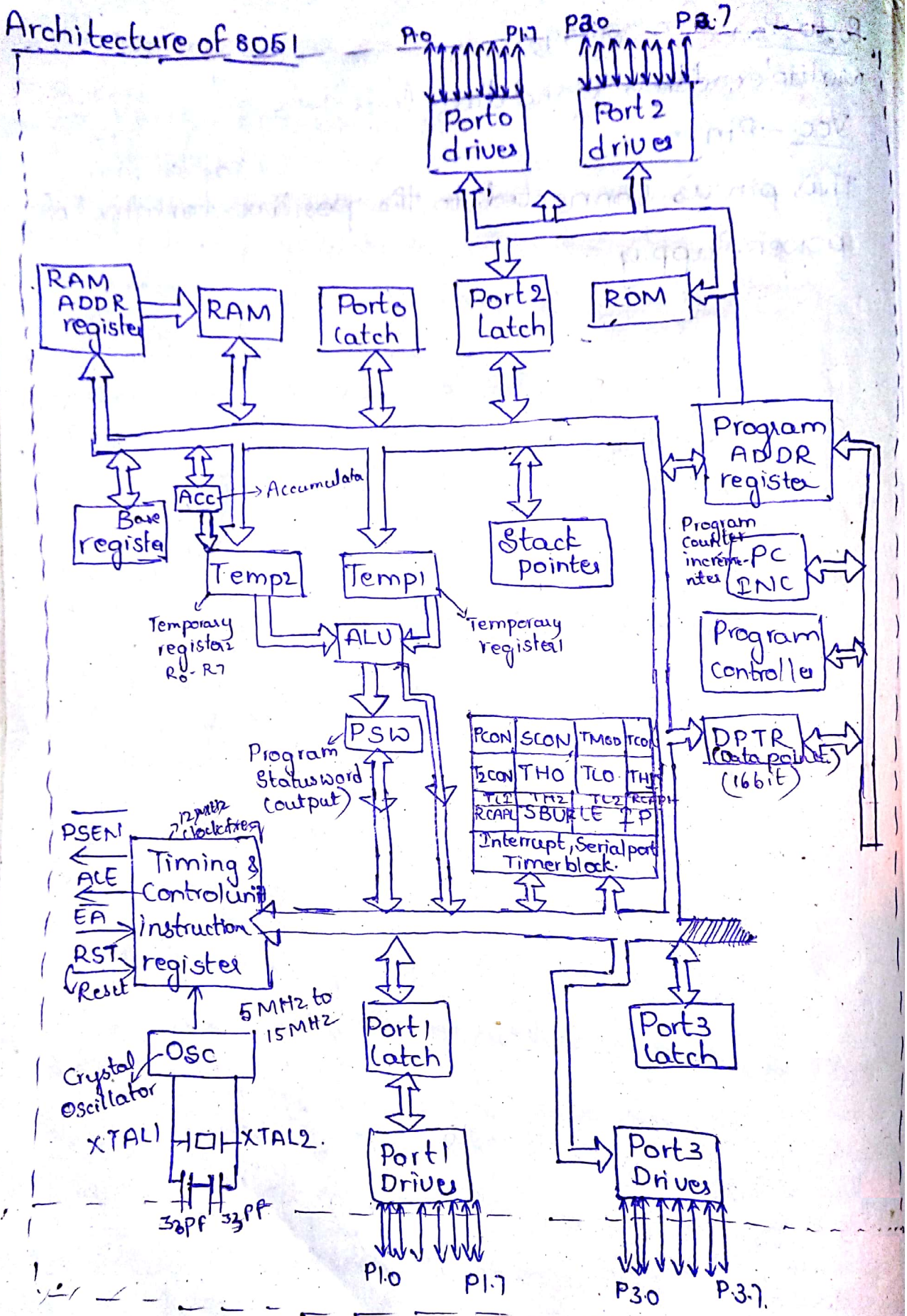
P_0 to P_{07} (Pin 32-39)

Multiplex addresses and data lines.

Vcc - Pin 40

This pin is Connected to the positive terminal of power Supply.

Architecture of 8051



Drivers: It is a programmable driver interfaces are used to provide interface b/w user to the system

Modelling Structures

1) Hardwad - Microcontroller.

2) Wannewan.

The capacitor bypasses the extra current to the ground. It acts as a protector.

The Oscillator o/p is always Sine wave

PCON - Power Controller - To Stabilize different Voltage rating.

SCON - Serial Controller used to control Serial data

TMOD, TCON, T2CON are timer Controllers.

for timing and Counting purposes.

CE - Latch Enable.

differences between microprocessor and microcontroller

Micro processor	Micro controller
1) Microprocessor is a programmable Semi Conductor device Controls a dataflow from input devices to Output through CPU [Microprocessor] by fetching, decoding and executing the data from memory	1) Micro Controller is a programmable Semi conductor Controller It executes fetched data in a memory and Send Output to external devices
2) Executed results is shown by flags in 8086	2) The executed result is shown by PSW in 8051

3) The microprocessor is having 1MB of memory	3) 8051 is having 128 Kb of RAM and 4 Kb of ROM
4) The microprocessor is having 5MHz of clock frequency	4) The microcontroller is having ≈12MHz of clock frequency
5) Seperated register for program. AX, BX, CX, DX, SI, DI, SP, BP, IP, ES, DS, SS, CS, flags	5) In 8051 registers are A, B, R0-R7.
6) Microprocessor is a Single chip it is having number of instructions	6) Microcontroller is a Single chip having no of inbuilted peripheral device
7) It a general purpose device	7) It is a Special purpose device.

Register Set of 8051

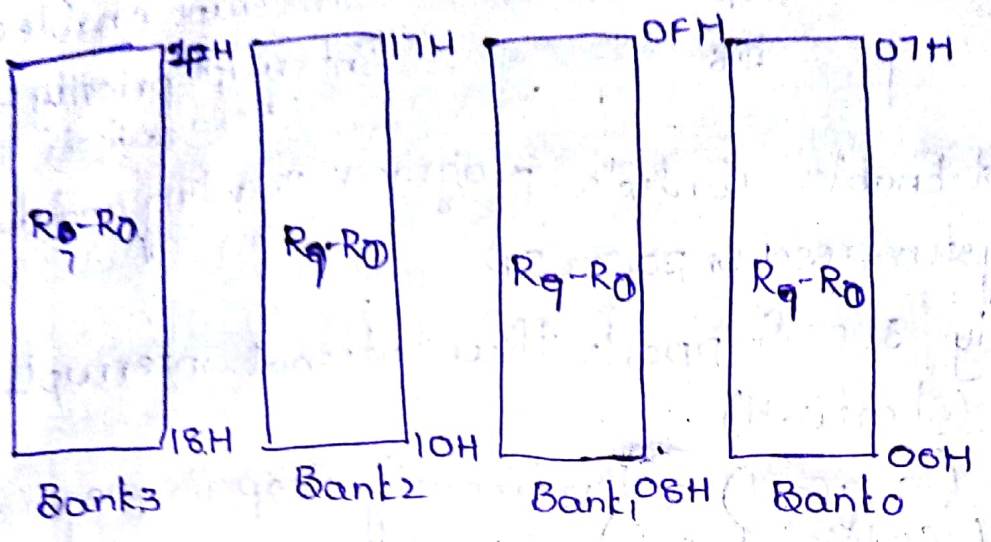
Register is a Space Used to Store the data temporarily
There are two types of register Set in 8051

1) General purpose registers
2) Special purpose register.

1) General purpose register

It is also called register banks. The Size of general purpose register is 32 memory locations of internal RAM

divided into 4 banks i.e bank0 bank1 bank2 banks shown in below



(Memory location 00 - 1FH)

In the fig the 8 registers in each bank refferedas as R_7-R_0 can be active at a given point time.

These registers Can be Used for data Storage and data movements temporarily.

Instructions Using these registers will be faster than instructions Using another memory locations.

Special function registers

There are 21 Special function registers like in the range 80H - FFH of internal RAM.

Name	Address in Hexa	Function
A	E0	Accumulator
B	F0	Arithmetic Or base register or Math register

DPL	82	Data pointer lower.
DPH	83	Data pointer higher
IE	A8	Interrupt enable contro
IP	B8	Interrupt priority cont (FIFO fashion)

Interrupt Enable controls program enabling and disabl after System reset or power ON.

Generally 8051 Supports three internal interrupts i.e

SCALL (shortcall)

LCALL (longcall) - Permanently Stop the program.

ACALL (Absolute call)

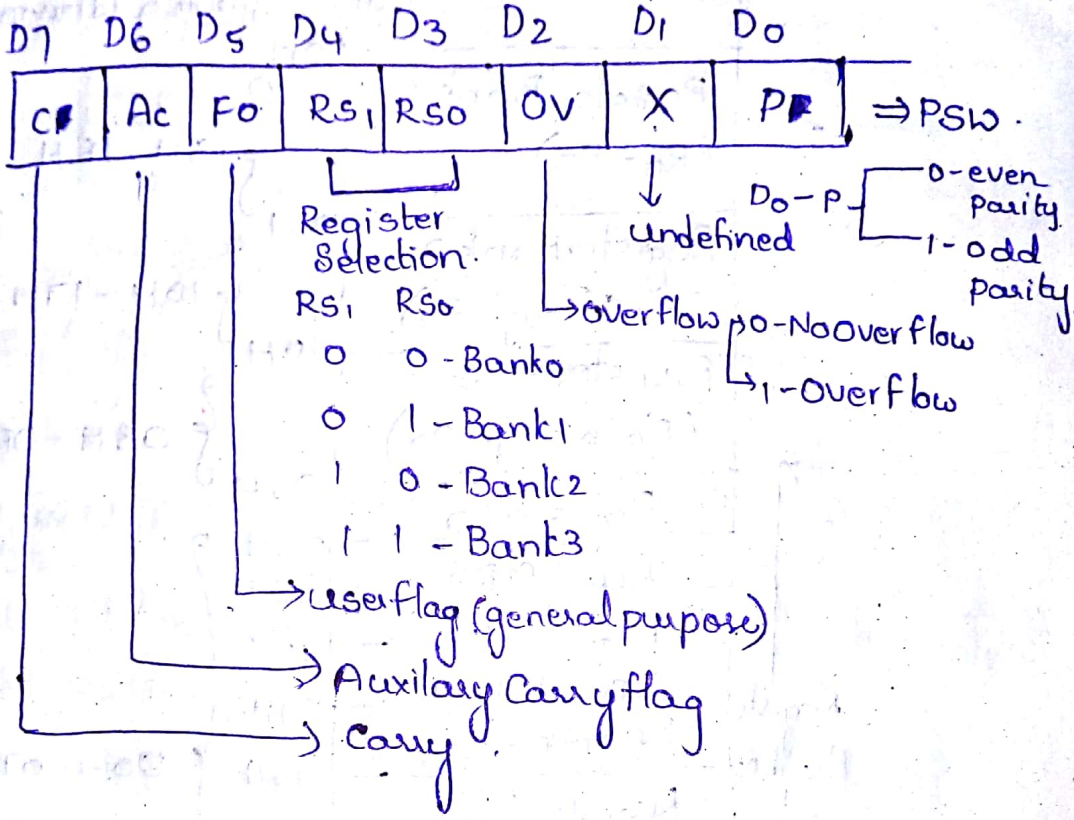
RETI (Return Interrupt)

P_0 (Port 0)	80	Port0
P_1 (Port 1)	90	Port1
P_2 (Port 2)	A0	Port2
P3 (Port 3)	B0	Port3
Pc (Program Counter)	-	Program Counter
PCON	87	Power Controller
PSW	D0	Program Status Word (Flag)
SBUF	99	Serial port data buffer
SCON	98	Serial Controller
SP	81	Stack pointer
TCON	88	Timer or Counter Contro
TMOD	89	Timer mode or Counte mode Control
TH0	8C	Timer Zero high byte
TH1	8D	Timer One high byte

TLO	8A	Timer Zero lower byte.
TL1	8B.	Timer One lower byte

PSW (Program Status word)

It shows the status of 8051 after the operation of ALU.



Memory Organisation of 8051

→ 8051 has inbuilted memory.

→ Memory is used for Storing program code and Variable data.

It Contains two inbuilted memory

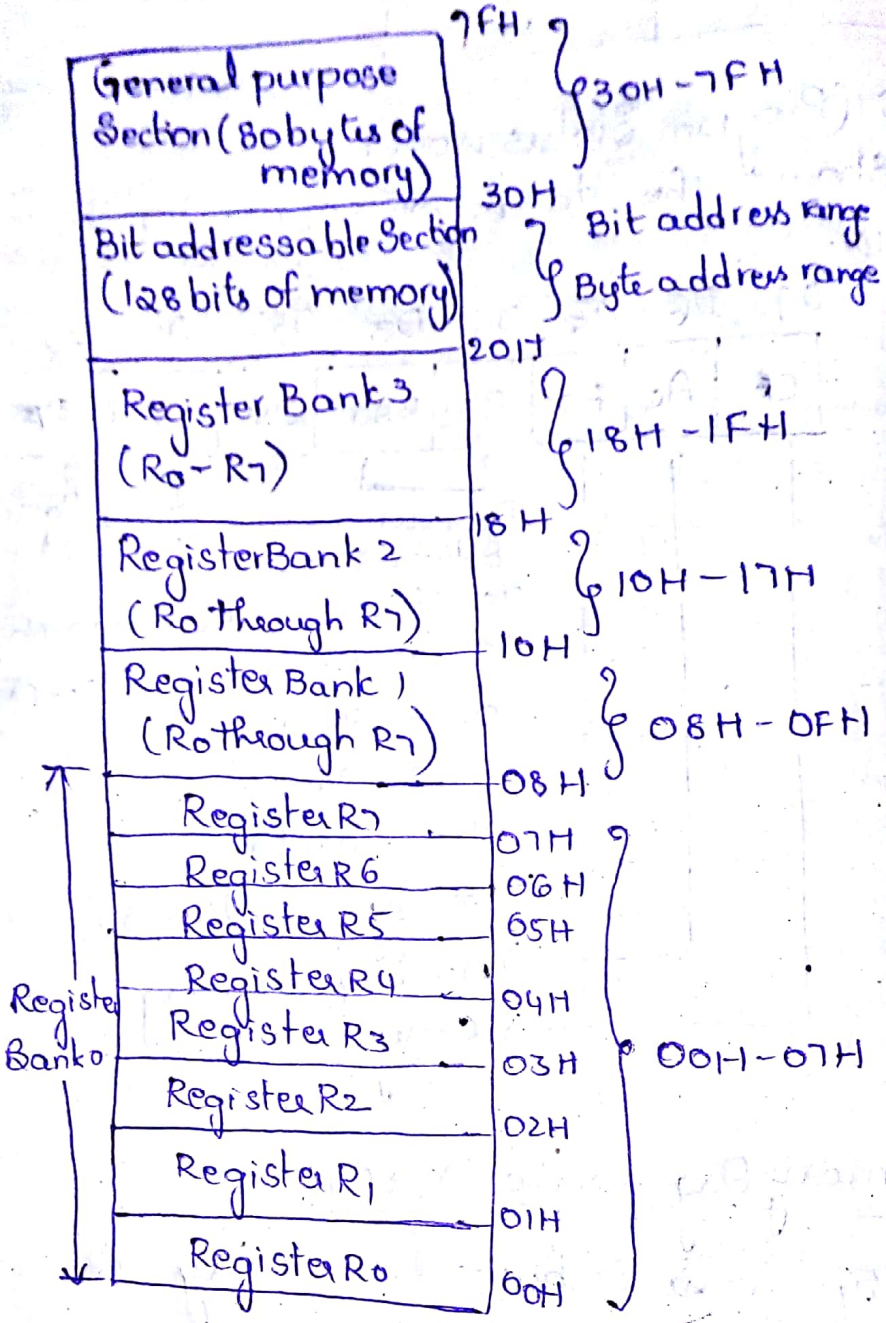
→ 128 bytes of internal RAM

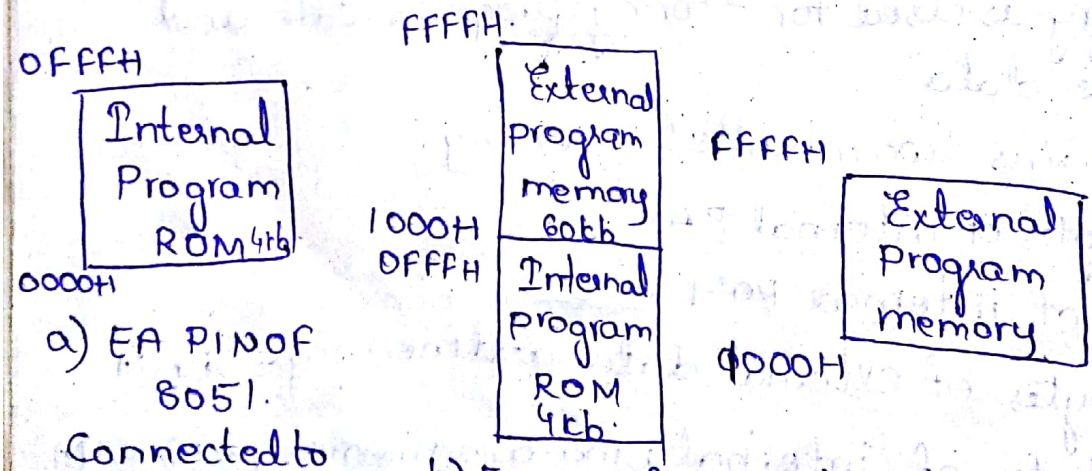
→ 4kb of internal ROM also

→ 64kbytes of external data memory - SRAM

→ 60kbytes of internal program memory - EPROM

Internal RAM Organisation





Program Memory Organisation

Bit addressable Section Used for read and Write through 20H-7FH for read and Write without disturbing remaining bits

This bits are available for user purpose

30H-7FH memory holds Special function register for read and Write Operation.

Program Memory

Pin+31 $\overline{EA}$/VPP (External Access)

If $\overline{EA}$=1 (It Selects internal ROM)

$\overline{EA}$=0 (It

External RAM

64 kbytes of external data memory.

Here the 64 kilobytes of memory Used for Variable external data purpose.

The range is 0000_H-FFFF$_H$

This memory can be accessed through instruction MOVX and DPTR.

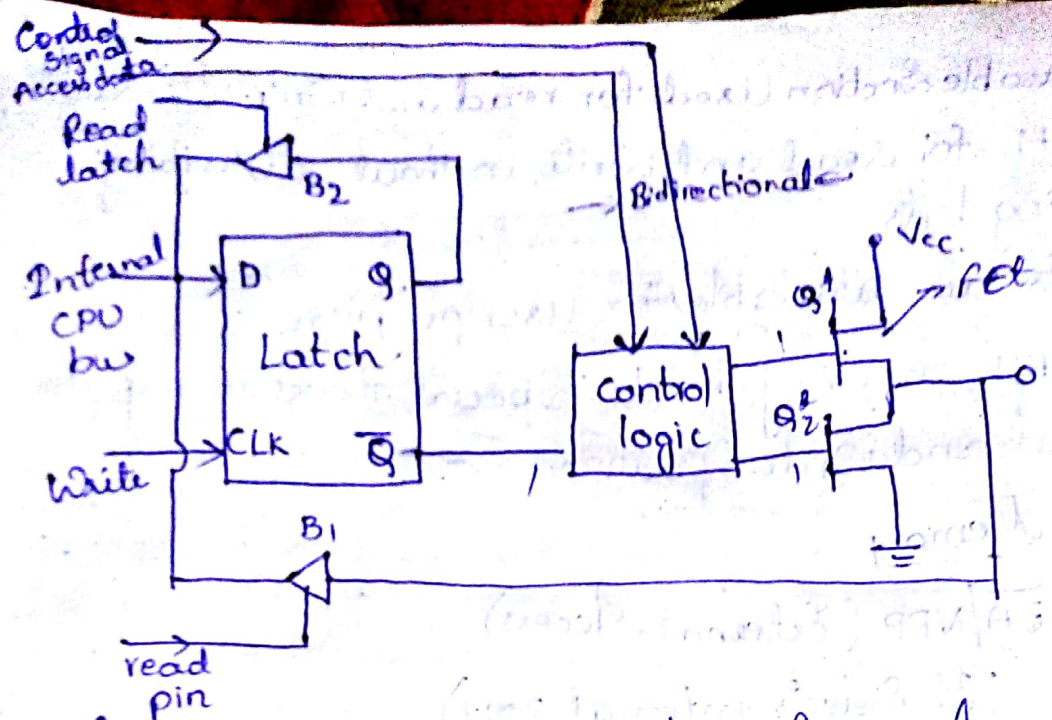
I/o ports of 8051

There are 4 I/o ports in 8051

port0, port1, port2, port3.

port0 is a bidirectional I/o port Used for input Output Operations and also holds addresses and datalines.

There are 8 addresses and datalines ($P_{0.0}$-$P_{0.7}$)



→ Latch is a register which performs logical Operation
→ Latch is independent of the clock Signal.
→ It is the reverse of flipflop.

D	Q	$\overline{Q}$
0	0	1
1	1	0

Case(iii) In bidirectional mode: if Control logic=1 It transfers address if Control logic=0 the data Signal will come out

→ Control logic is nothing but a gate to Control FET's
→ Vcc is to Control the load.

Port 0 performs 3 Operations

It is Configured as Input, Output and bidirectional.

1) Input (Port 0 as I/P)

Case1 D=1, Q=1, $\overline{Q}$=0

Then Q_1=OFF Q_2=OFF. Pin = high impedance State At this time we Can write the data Or read the data.

It means Output is in high impedance State

Case 2 D = 0 Q = 0 $\overline{Q}$ = 1

Q_2 is in ON State Q_1 is OFF State

Logic low returns to the D-latch

This will effectively ground the pin.

Hence a logical low Output is achieved by writing low at input.

To float the O/p external pull ups are required

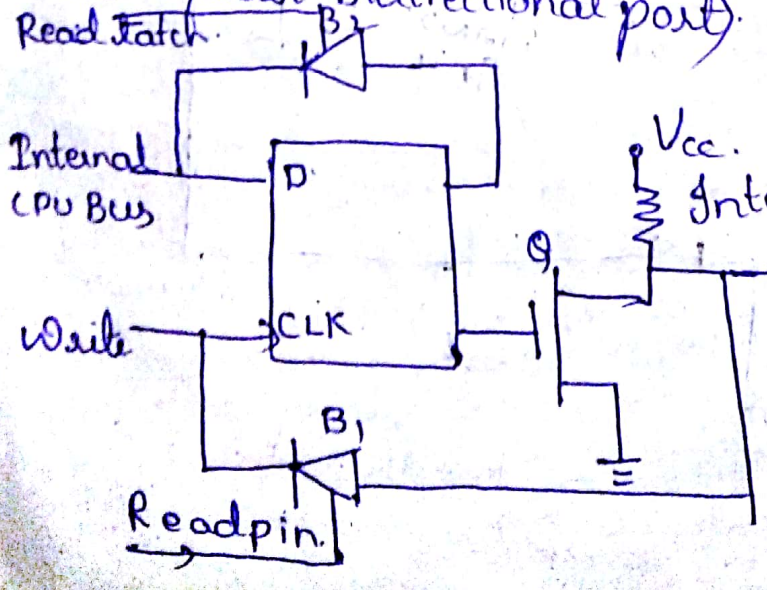
For Configuration port as bidirectional a Control logic Circuit Comes to the Operation and controls the Q_1 transistor.

A Control logic allows a logic high at the gate of the Q_1 transistor, Q_1 Comes to the ON. and Q_2 is OFF.

This provides a logic high Output.

To get a logic low a transistor are changed. from Q_1-OFF, Q_2-ON. (External pull up registers are used to drive the data in Port 0)

Port 1 (Quasi bidirectional port).



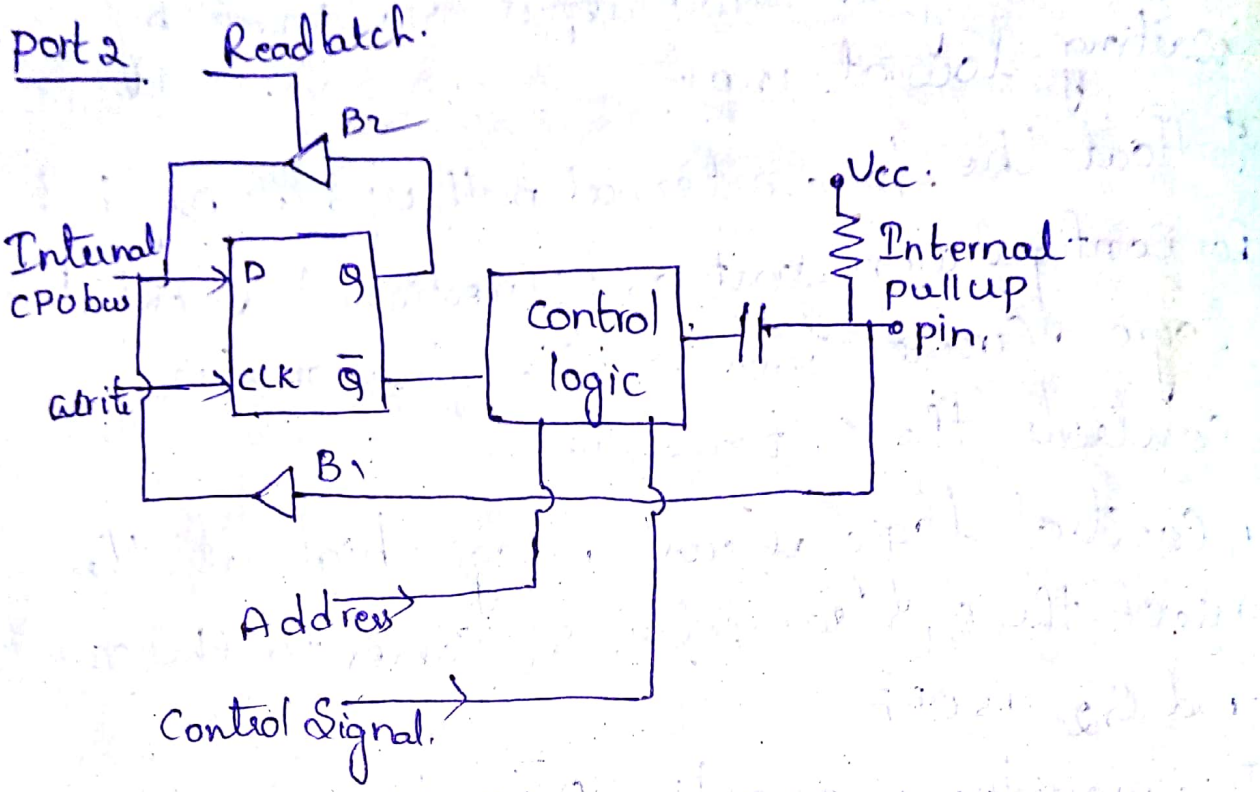
It is used to maintain Stability

It acts as an isolator.

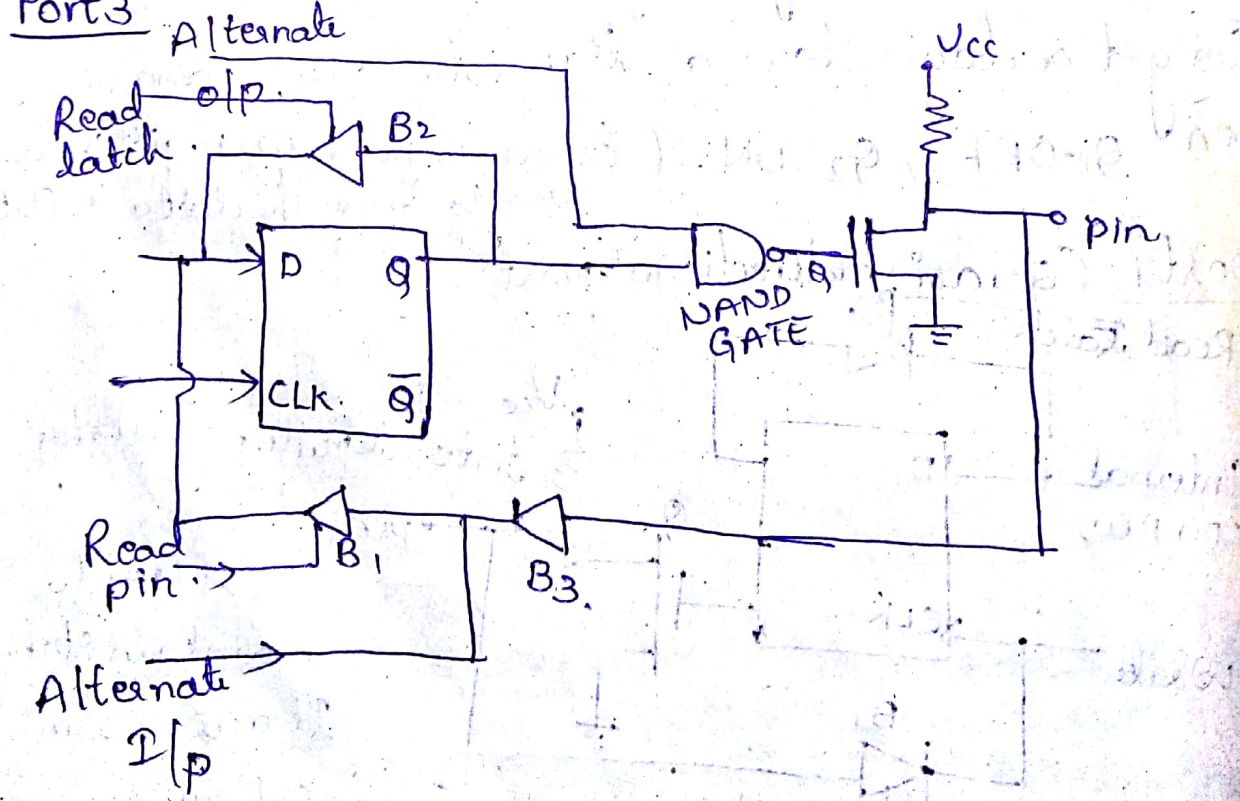
I/o ports of 8051

Port 0 ⟷ true bidirectional.
Port 1 ⟷ quasi bidirectional / Simple bidirectional
Port 2 ⟷ } Simple bidirectional Simple I/o.
Port 3 ⟷ } Special function port

Port 2



Port 3



Timers and Counters

8051 has is used for multipurpose programming So the 8051 is require timers and counters

Timer:

Timer is used to Set internal clock period to an Operation

Counter:

Counter is used to Set number of external counts given by the external clock

8051 has a two 16 bit timers namely

1. Timer 0 — TL0 - 8 bit, TH0 - 8 bit
2. Timer 1 — TL_1 (8 bit), TH_1 (8 bit)

TCON, TMOD is a Special register to control the Timer Operation.

Timer Controller

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
TF_1	TR_1	TF_0	TR_0	IE_1	IT_1	IE_0	IT_0

→ TCON is reserved for Controlling timer bits and timer flags.

→ In the format Lower Nibble Controls timer interrupts (level trigger and edge trigger interrupts of 0 and 1 timers.

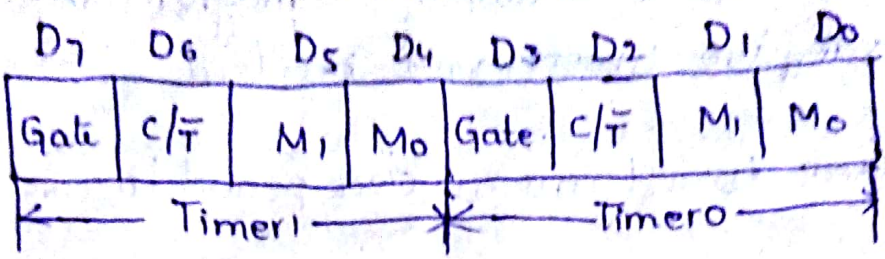
→ D_0 - IT_0 - external interrupt 0 edge timer. If D_0=1 edge trigger; If D_0=0 level trigger

→ D_1 - IE_0 external interrupt edge flag — D_1=1 flag detected; D_1=0 flag disabled.

→ D_2 - IT_1 — external interrupt 1 edge timer — If D_0=1 edge; If D_0=0 level.

→ D_3 - IE_1 - external interrupt edge flag — flag detected D_3=1; D_3=0 flag disabled.

TMOD



M_1	M_0	Mode	Description
0	0	0	Timer/counter (13-bit)
0	1	1	16-bit timer/counter
1	0	2	8-bit autoreload timer/counter
1	1	3	Two seperate 8-bit counter for timer-0. Timer1 Stops

TCON

D_4 – TR_0 – Timer Run in Timer 0 Run.

If. $D_4 = 1$ Timer Run enable

$D_4 = 0$ Timer Run disable.

D_5 – TF_0 – Timer 0 Over flow

If $D_5 = 1$ Timer Over flow enable

$D_5 = 0$ Time Over flow disable.

D_6 – TR_1 – Timer 1 Run

$D_6 = 1$ enable.

$D_6 = 0$ disable.

D_7 – TF_1 – Timer 1 Over flow

$D_7 = 1$ enable.

$D_7 = 0$ disable.

TMOD

M_0 and M_1 are mode Selection bits

$C/\overline{T}$ = Counter/$\overline{\text{Timer}}$

$1/\overline{1} = 1/0$ = Counter (external Operation)

$0/\overline{0} = 0/1$ = Timer (for internal Operation)

Gate: It will acts as an interface gate. The gate will be Open when the Operation is done.

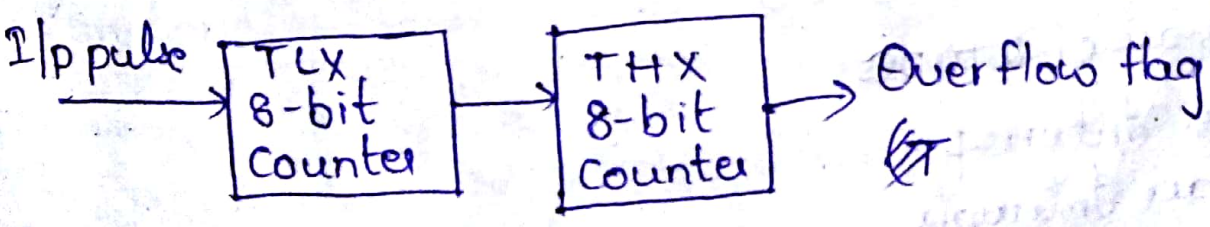
Modes of Timers

There are 4 modes in Timers and Counters

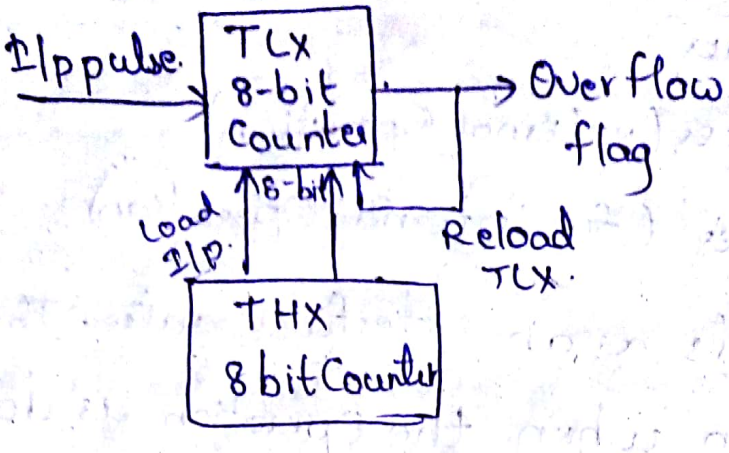
i.e 1) mode 0 (13-bit)
2) mode 1 (16-bit)
3) mode 2 (8-bit autoreload)
4) mode 3 (Seperate 8-bit)

Mode 0:-

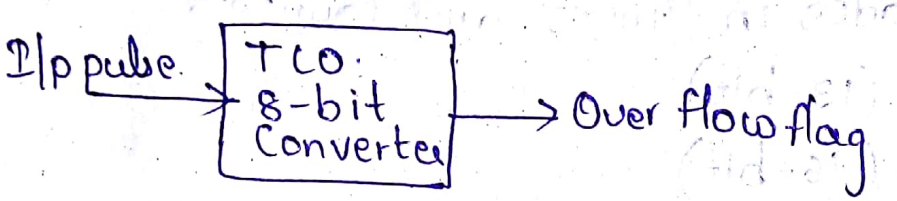
Mode 1:-

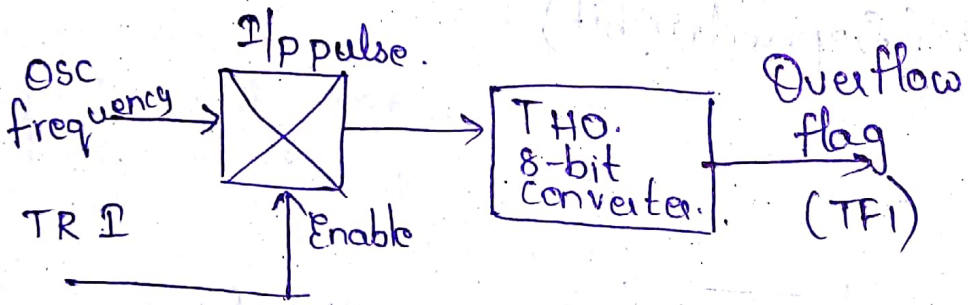


Mode 2 drawback = Overflow



Mode 3





Interrupts of 8051

Interrupt is an event driven element used to execute a CPU tasks.

Interrupts are of 2 types.

1. Software Interrupts
2. Hardware Interrupts
   - 1) INT0
   - 2) INT1

There are actually 6 interrupts in 8051 (included reset)

1. RESET: It reintialises the entire program. Setup.
The It interlinked to the all the Segments of Controller

2. Timer Interrupts:

There are 2 Timer interrupts

2. Timer-0
3. Timer-1

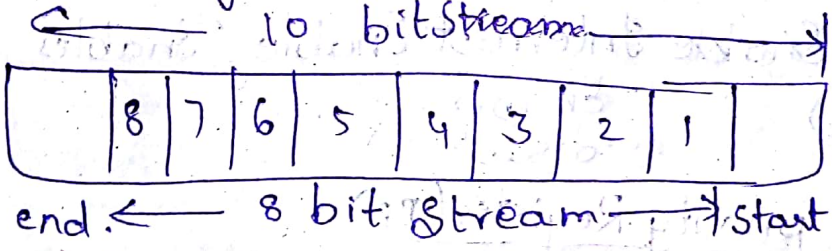
- There are 2 External interrupts

4. INT-0
5. INT-1

Serial port interrupt

8051 Supports Serial and parallel Communication.
but in 8086 - Serial Communication
with the help of 8255 - parallel Communication

UART - Data transmission (At present)
Universal Asynchronous. Reciever and Tranismitter



To enable Or disable interrupts in 8051 a Special functions registers of IE and Ip is Used

IE - Interrupt enable
Ip - Interrupt priority.

Interrupt Enable Register (IE)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
EA	-	ET2	ES	ET1	EX1	ET0	EX0

D_0 - EX0 - Enable external Interrupt 0
1 - enable
0 - disable

D_1 - ET0 - External Interrupt 0 Over flow. Timer.
1 - Over flow enable
0 - Over flow disable.

D_2 - EX1 - External Interrupt 1
1 - enable
0 - disable.

D_3 - ET_1 - External Interrupt 1 Over flow Timer.
1 - Overflow enable
0 - disable.

D_4 - ES - External Serial port

D_5 - ET_2 - External Timer-2 Over flow
This is not used in 8051. This is for futur purpose.

D_6 - Don't Care.

D_7 - EA - Enable Interrupt Enable. (Enables all the interrupts) 1 - Enable.
0 - disable.

Interrupt priority Register (IP)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
-	-	PT2	PS	PT1	PX1	PT0	PX0

D_0 - PX_0 - Priority External Interrupt 0.
1 - enable
0 - disable.

PT0 - Priority Timer 0 Over flow
1 - Over flow enable
0 - disable

PX1 = Priority External Interrupt 1
1-enable
0-disable.

PT_1 - Priority Timer1 Overflow.
1-enable
0-disable.

PS - Priority Serial port

PT_2 - Priority External Timer2 Overflow
Not Used in 8051.

Serial Communication of 8051

For Serial Communication of 8051 Supports 3 Special function registers

1) SBUF 2) SCON 3) PCON.

SBUF (Serial buffer)

It is a simple 8 bit register used to transmitte and recieve data. It acts as an Accumulator.

SCON :- (Serial controller)

D7	D6	D5	D4	D3	D2	D1	D0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

D0 - RI - Reciever.
D1 - TI - Transmitter.

D0 - 1 - Reciever mode enable.
0 - disable

D1 - 1 - Transmitter enable
0 - disable.

RB-8 - Reciever bit 8

TB8 - Transmitter bit 8

D_4 - REN - Return.

D_5 - SM_2 - Serial Mode 2 (Multicontrollers on a same board)

SM_1 - Serial Mode 1

SM_0 - Serial Mode 0.

SM_1	SM_0	Mode	Description
0	0	- Serial Mode 0	shift operation.
0	1	- Mode 1	8-bit UART
1	0	- Mode 2	9-bit UART
1	1	- Mode 3	9-bit UART with programmable baudrate (speed of data transmission)

PCON. (Power Controller)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SMOD	X	X	X	GF1	GF0	PD	IDL

D_0 - IDL - Ideal - 1

Running - 0

D_1 - PD - Power down mode -

This bit is Set 8051 enter to power down.

Otherwise it is Set to 0

D_2 - GF0 - General purpose flag 0

D_3 - GF1 - General purpose flag 1 } (User purpose).

D_5, D_4, D_6 (Undefined)

D_7 - Serial mode (SMOD) D_7 - 1

D_7 - 0 - parallel mode

UNIT 5- PIC ARCHITECTURE

14/3/20

PIC :

Unit-5

PIC Micro Controller

PIC - peripheral Interface Controller.

It is advanced microcontroller than 8051.

Microcontroller	8051	PIC	ARM (Advanced Risk Machines)
	RAM, ROM, EPROM.	EEPROM, FLASH ROM, UVROMS. (8 bit)	UVROM (128 bit)

1) PIC is an advanced 8 bit microcontroller given by Microchip technology Corporation in 1989.
2) PIC Combines a small amount of data RAM Onchip ROM few I/o ports and One timer on a single chip.
3) There are 5 major 8-bit microcontrollers
   1) MOTOROLA - 6811 (US)
   2) INTEL - 8051 (US)
   3) XILOG - Z8 - (china)
   4) ATMEL AVR (US)
   5) MICRO CHIPS - PIC. (US)

There are 6 different generations of 8 bit PIC introduced by MICRO CHIP Over the last few years.

8086 is not having inbuilt pheripherals

1) PIC-12xxx	8 pin	12 bit instruction format
2) PIC-14xxx	26 pin	14 bit instruction format
3) PIC-16C5X	18-28 pin	12 bit
4) PIC-16CXX	28-40 pin	14 bit
5) PIC-17XXX	8 - 40 pin	16 bit
6) PIC-18XXX	18-80 pin	16 bit

PIC 18 is advanced 8 bit micro Controller comprises 4 kbytes of RAM and 2MB ROM USART (Universal Synchronous Asynchronous Receiver Transmitter) watched dog timer I^2C (Buses inter integrated circuit) SPI (Serial peripheral Interface) CAN (Computer Aided network), pulse width modulation, LPO (Lower power Operation mode) 4-8bit timers, 1-Serial port, ADC (Analog-Digital Converter) Analog Comparator, flash memory, EPROM, EEPROM, SRAM

1) PIC is having inbuilted ADC but 8051 is not having inbuilted ADC.
2) SPI, I^2C, ADC are buses reading and writing data to from master to Slave. (buses for Communication).
3) Microprocessor is inbuilted in 8051 as it holds Only instructions.
4) Microprocessor is known as master as it holds the instructions.
5) Slaves are 8255, Serial port, RAM, ROM, Timers etc in 8051

6) SPI interfaces the data and transfers the data in Serial (bit by bit transmission)

7) I^2C is a true bidirectional interfacing device (Serial)

8) CAN is used to control the master devices of the System.

9) Watchdog timer is a hardware timer in that timer the timer module is designed by n Counter (ctrl+Alt+del) the Signal is going to connect watch dog timer.
This timer is used to Solve the hang up condition of the device.

9) USART - Synchronous and Asynchronous way of communication

10) flash memory is used to protect the program code. This protected memory is known as catch memory

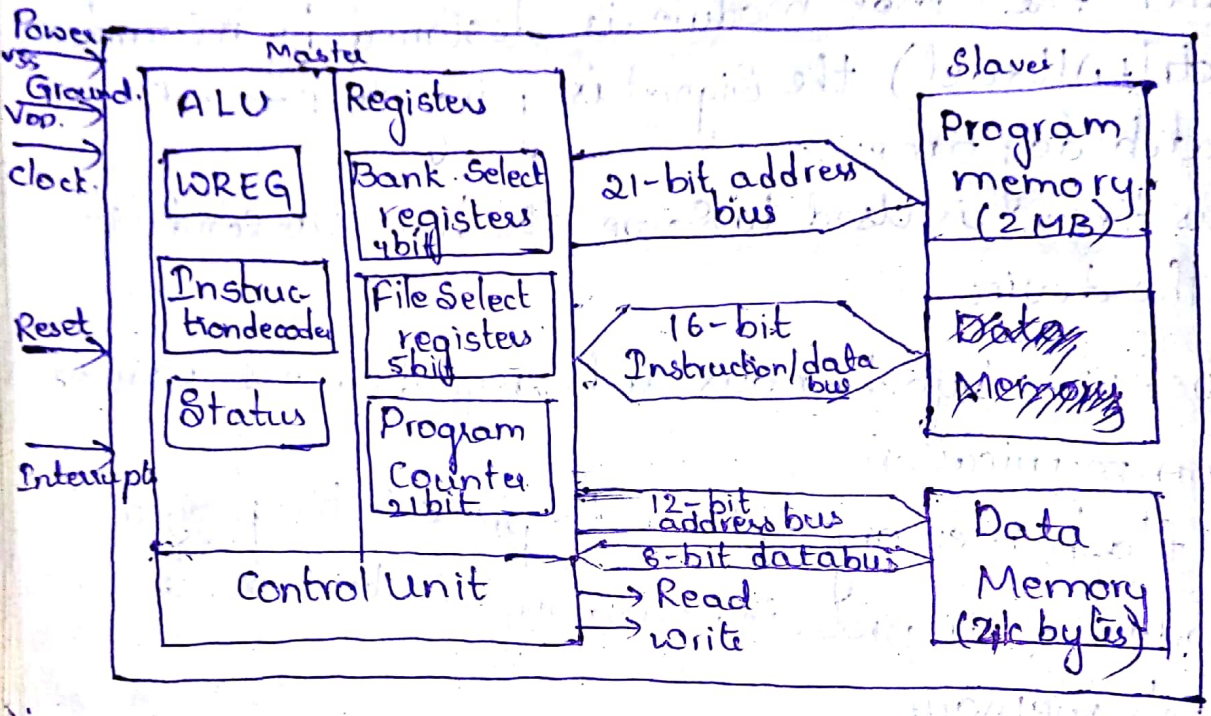
11) It is hardware architecture (PIC 18) Supported

12) by using RISC processor (Risk instruction Set Computer)

8051	PIC
1). 8051 is a 8bit processor	1) PIC is also a 8bit processor
2) 128 bytes of RAM	2) 4kbytes of RAM.
3) 64 kbytes of program memory	3) 2MB of program memory.

4) Two Timers in 8051	4) 4 Timers in PIC.
5) One Serial port	5) One Serial port
6) 32 I/o pins	6) 33 I/o pins
7) 8051 is not having inbuilted ADC.	7) PIC is having inbuilt ADC.

Architecture of PIC-18 MicroController



Data Memory is Variable memory and program memory is having inbuilt program dumper.

Data Memory and program Memory are two Sep Onchip memories.

The capacity of prog

3 parts are present in microprocessor Unit

1) ALU (Arithmetics logical Unit)
2) Registers. (To perform ALU registers are Used
3) Control Unit

Arithmetic and logical Unit is Used to perform all Arithmetic and logical Operations Using register. WREG is nothing but Accumulator. It is a destination register. It holds the data result temporarily. WREG (Working register). It performs read and Write Operations. (It is 8 bit)

Status: Status register is nothing but flags. It shows the result of ALU. It is an 8 bit register.

7	6	5	4	3	2	1	0
-	-	-	N	OV	Z	DC	C

Carry flag - If the result is generating Carry it is Set 1
1 - Carry
0 - No Carry

Digit Carry - Nothing but Auxilary Carry flag.
The 4th bit Send a Carry to the 5th bit.
It is generally present in Multiplication and division Operations.

Zero flag : 1 - Result 0
0 - Result not Zero

Overflow flag : It will appear in Mul and div. more than 8 bits. 8x8 = 64.

Negative flag : result Negative = 1
Positive = 0

Instruction decorder

It is Used to decode 16 bit instructions

Control Unit

It is Used to control Read and Write Operations

Registers

There are 3 types of Registers Used in PIC micro Controller

1. Bank Select registers (4 bit)
2. File Select register (5 bit)
3. program Counter (21 bit)

Program Counter:

1) It Counts the total no of lines of program.
2) It is interlinked with interrupter
3) It also Connects with memory
4) This program Counter is Used to hold program memory addresses.
5) but in 8051 we have DPTR as program Counter which is 16 bit (DPTL and DPTH)
6) In PIC it is divided into 3 register.

1) PCU (5 bit) 2) PCH (8) bit 3) PCL (8 bit)

By default PCL is available for starting address of any memory location.

Bank Select Registers

Bank Registers are Used to Select file register file registers are Used to hold the program code

It is a 4bit Special function register Used in direct addressing data memory.

File registers are Used to Select Indirect data memo

Bank registers are Used to Select direct data memo

whole memory is divided to banks.

Banks are divided into 16 banks

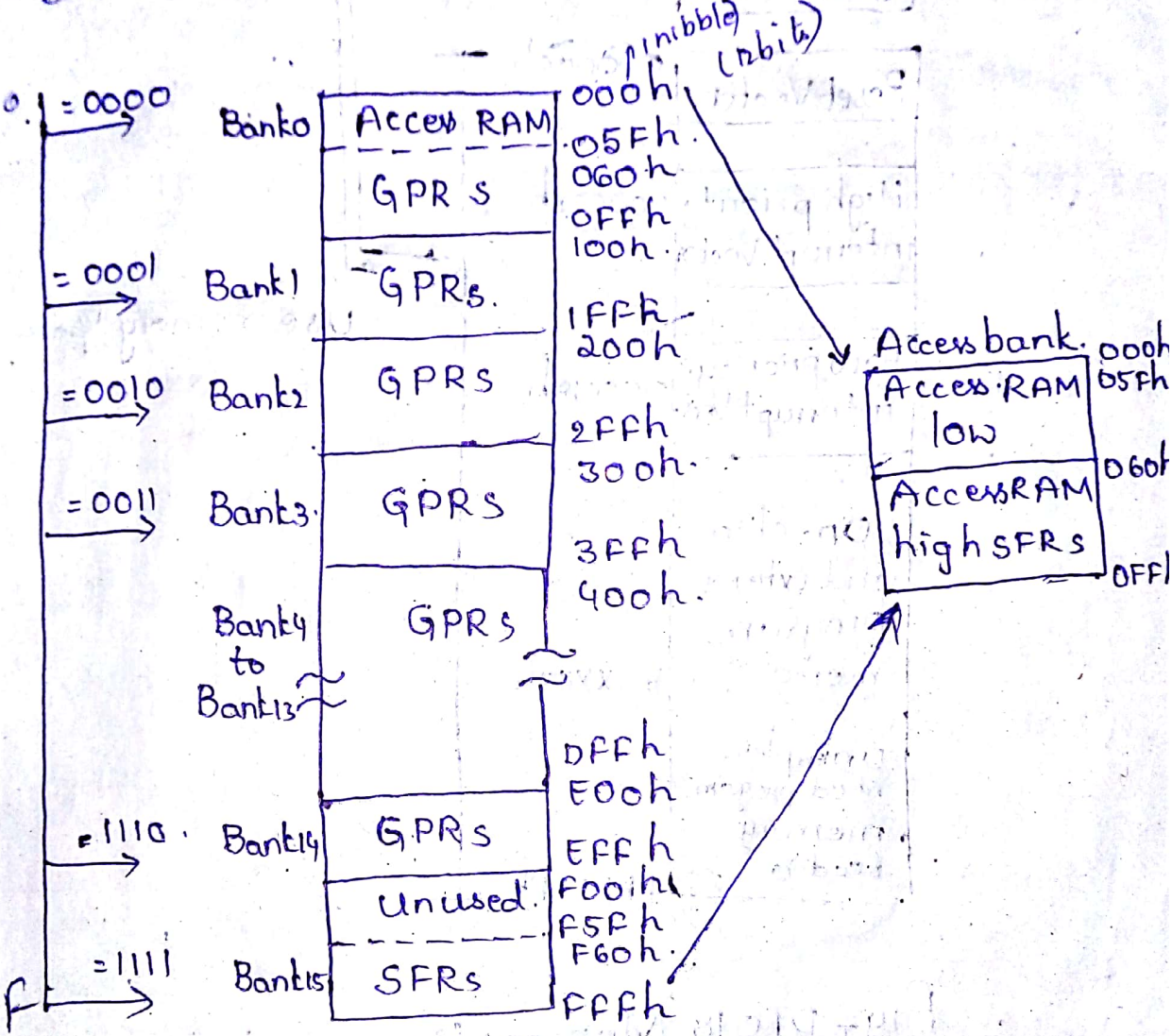
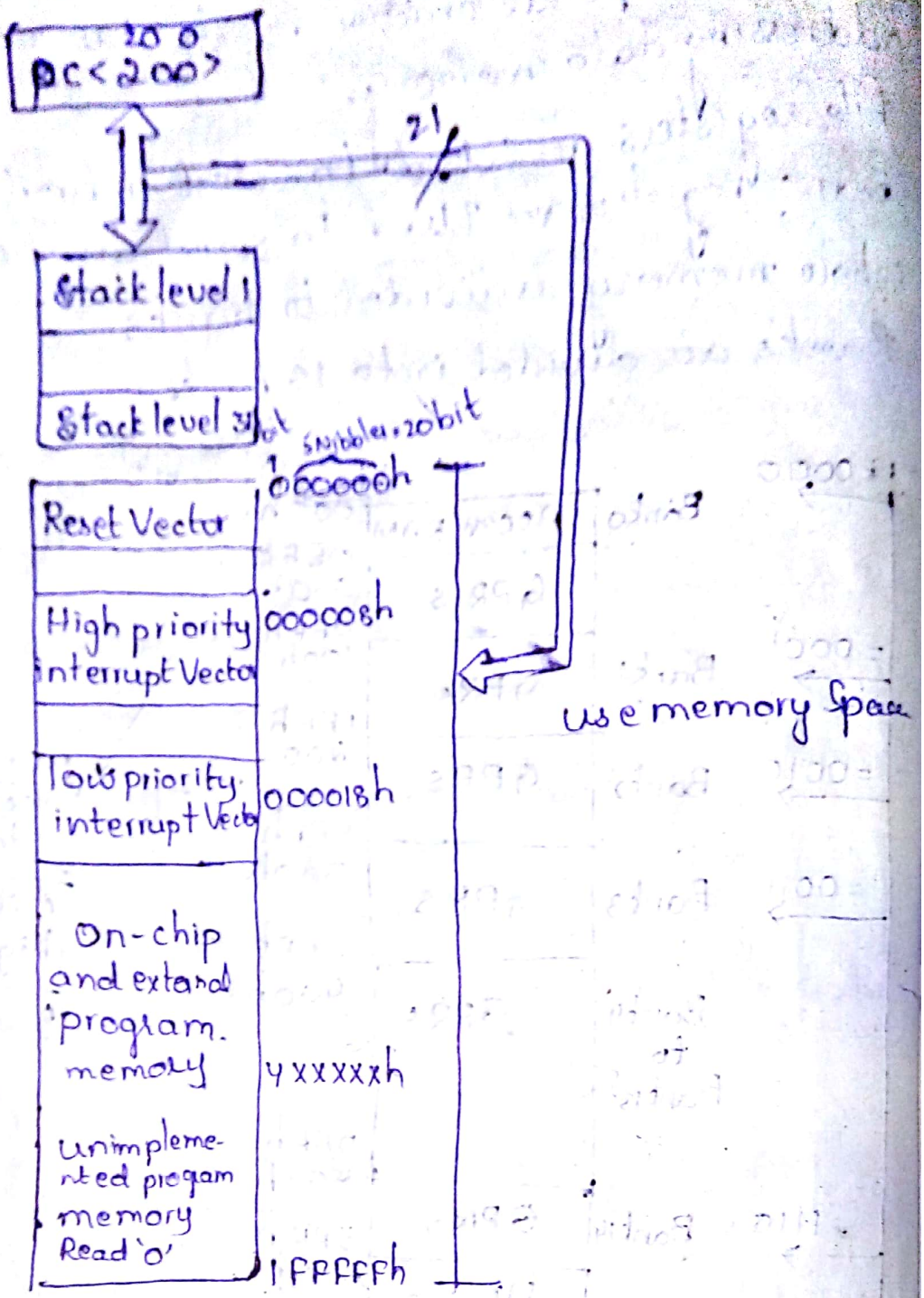


Fig: Data Memory Map for PIC 18

Each bank holds 256 bytes: Out of these banks Only One bank is active at a time.

Out of these 16 banks the data memory is having a memory Space for Unused purpose i.e F00h to F5Fh It holds the data temporarily.



Fig+ PIC 18 Memory Organisaton

The Stack is Used to hold the temporarily. Here the program memory Contains 31 Stack addresses which is to hold the retun addresses for Subroutine program.

CPU registers of PIC 18

Name	Description
1) TOSU	Top of Stack (Upper)
2) TOSH	Top of Stack (Higher)
3) TOSL	Top of Stack (lower)
4) STKPTR	Stackpointer
5) PCLATU	Upper program Counter latch.
6) PCLATH	Higher program Counter latch
7) PCL	Program counter lower byte
8) TBLPTRU	Table pointer upper byte
9) TBLPTRH	Table pointer Higher byte. High program Counter latch
10) TBLPTRL	Table pointer lower byte
11) TABLAT	Table latch.
12) PRODH	High product register.
13) PRODL	Low product register.
14) INTCON	Interrupt Control register.
15) INTCON2	Interrupt Control register2
16) INTCON3	Interrupt control register3.
17) INDF0	Indirect file register pointer0
18) POSTINC0	post-increment pointer0 (to GPR's)
19) POSTDEC0	post-decrement pointer0. (to GPR's)

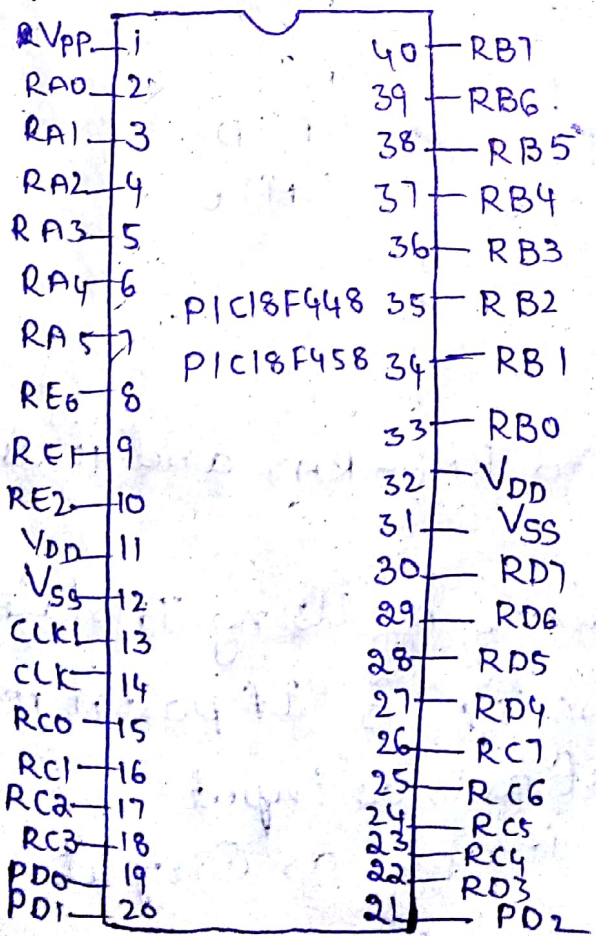
20) PREINC0	pre increment pointer 0 (to GPR's)
21) PLUSW0	Add WREG to FSR0
22) FSR0H	File Select register Zero high byte.
23) FSR0L	File Select register zero low byte.
24) WREG	Working register.
25) INDF1	Indirect file register pointer-1
26) POSTINC1	post increment pointer-1 (to GPR's)
27) POSTDEC1	post decrement pointer-1 (to GPR's)
	pre increment pointer-1 (to GPR's)
28) PREINC1	pre increment pointer 1 (to GPR's)
29) FSR1H	File Select register 1 high byte
30) FSR1L	File Select register 1 lower byte
31) BSR	Bank Select register.
32) INDF2	Indirect file register pointer 2.
33) POSTINC2	post increment pointer-2 (to GPR's)
34) POSTDEC2	post decrement pointer 2 (to GPR's)
35) PREINC2	pre increment pointer 1 (to GPR's)
36) PLUSW2	Add WREG to FSR2
37) FSR2H	File Select register 2 higher byte
38) STATUS	Status register.
39) FSR2L	File Select register 2 lower byte

I/O ports of PIC-18 Micro Controller

Audmo boards

Pins / chip	18pin PIC18F1220	28pin PIC18F2220	40pin PIC18F458	64pin PIC18F6652H	80pin PIC18F8525
Port A	X	X	X	X	X
Port B	X	X	X	X	X
Port C		X	X	X	X
Port D			X	X	X
Port E			X	X	X
port F				X	X
port G				X	X
port H				X	X
port J				X	X
port k					X
port L					X

Pin diagram of PIC 18



The no of I/o ports in PIC 18 family Varies according to the Selected number of PIC.

Examples PIC 18F448 / 58 family Consist 53 pins for reseved 5 ports (port A port B port C port D and port E)

It is not necessary to have all 8 ports pins are 8 pins

Here the port A is assigned 7 pins and port B, C, D are 8 pin and port E is having 3 pins.

Out of these ports has 3 Special functions registers associated with it. Those are Called port X, TRISX, latches LATX.

Port X	TRIS X	LAT X
Port A	TRIS A	LAT A
Port B	TRIS B	LAT B
Port C	TRIS C	LAT C
Port D	TRIS D	LAT D
Port E	TRIS E	LAT E

Port A

Here Port A is Copied $RA_0 - RA_7$ and it is Used for I/o pin

This port is enabling by Using Tristate A register Logic 0 If the register is If port A returns all Ones it acts as input otherwise it acts as Output

Port B

RB0 - RB7 for using these pins as both input & Output by enabling bits of tristate B register.

If this register returns all ones it acts as i/p. otherwise o/p.

Port C