LECTURE NOTES

ON

MICROPROCESSOR

&

MICRO CONTROLLER

SEMESTER–4TH

Prepared By: Abhaya Ghosh

Unit - 4

Microprocessor based system development aids

Interfacing :-

→ It is a mp used to connect it with various peripheral to perform various operation to obtain a desire o/p.

There are 2 types of interfacing.

(i) Memory interfacing

(ii) I/O interfacing

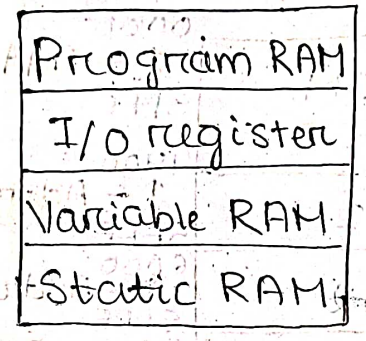
Memory Interfacing

→ It is used when the mp need to access the memory frequently for reading & writing data store in memory.

→ It is also used when writing & reading to a specific register or a memory chip.

I/O mapped I/O :-

→ Different address space require.
→ Separate read & write control signal.
→ I/O control signal is also used to control read & write operation.
→ Processor provide separate address range for memory or I/O.
→ Ex:- IN, OUT



Memory Interfacing :-

→ Microprocessor 8085 can access 64KB memory since address bus is 16 bit. But it is not always necessary to use full 64KB address space. The total memory size depends upon the application.
→ Generally EPROM is used as program memory & RAM is used as a data memory. When both EPROM & RAM are used the total address space 64KB that share by them.
→ The capacity of data memory & program memory depends on the application.
→ It is n't always necessary to select

one EPROM or one RAM, we can have multiple EPROM & multiple RAM as per the requirement of the application.

→ We can place EPROM or RAM anywhere in 64KB full address space but the program memory should be located from address 000H & rest address of 8085 MP.

→ It is n't always necessary to locate EPROM & RAM in consecutive memory location.

- The memory interfacing required:-

→ Select the chip

→ Identify the register

→ Enable the appropiate buffer.

→ MP system include memory device & I/O device. It is imp to note that mp can communicate with one device that a time, since the data, address & control buses are common for all devices.

→ In order to communicate with the memory or I/O device it is necessary to decode the address from the mp due each device can be access independently.

Address Decoding Technique

→ There are 2 types of address decoding technique. (i) Absolute / Full decoding
(ii) Partial / Linear decoding

→ With absolute decoding, we use all address line ($A_0 - A_7$) for decoding of I/O devices.

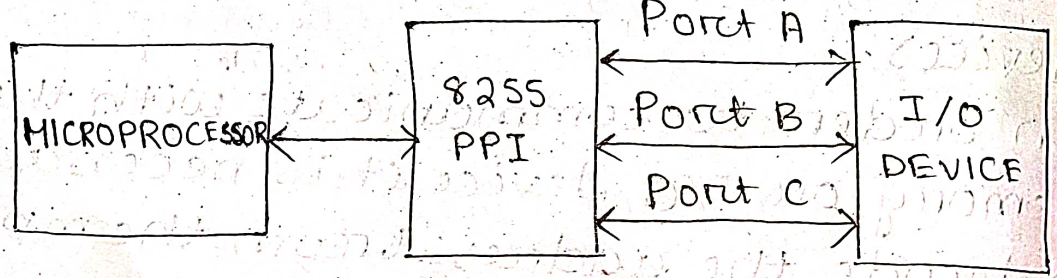
→ With partial decoding few address line are not used for decoding I/O device.

→ With absolute decoding we have fix address for decoding I/O.

→ With partial decoding multiple address may be decode.

→ Absolute & partial decoding may happen in case of memory interface.

8255 Programmable Peripheral Interface



→ 8255 is a peripheral programmable interface.

→ It was developed & manufacture by Intel corporation in 1970.

→ It is also called as 8255A.

→ 8255 PPI is the interface betⁿ mp & I/O device to connect peripheral device.

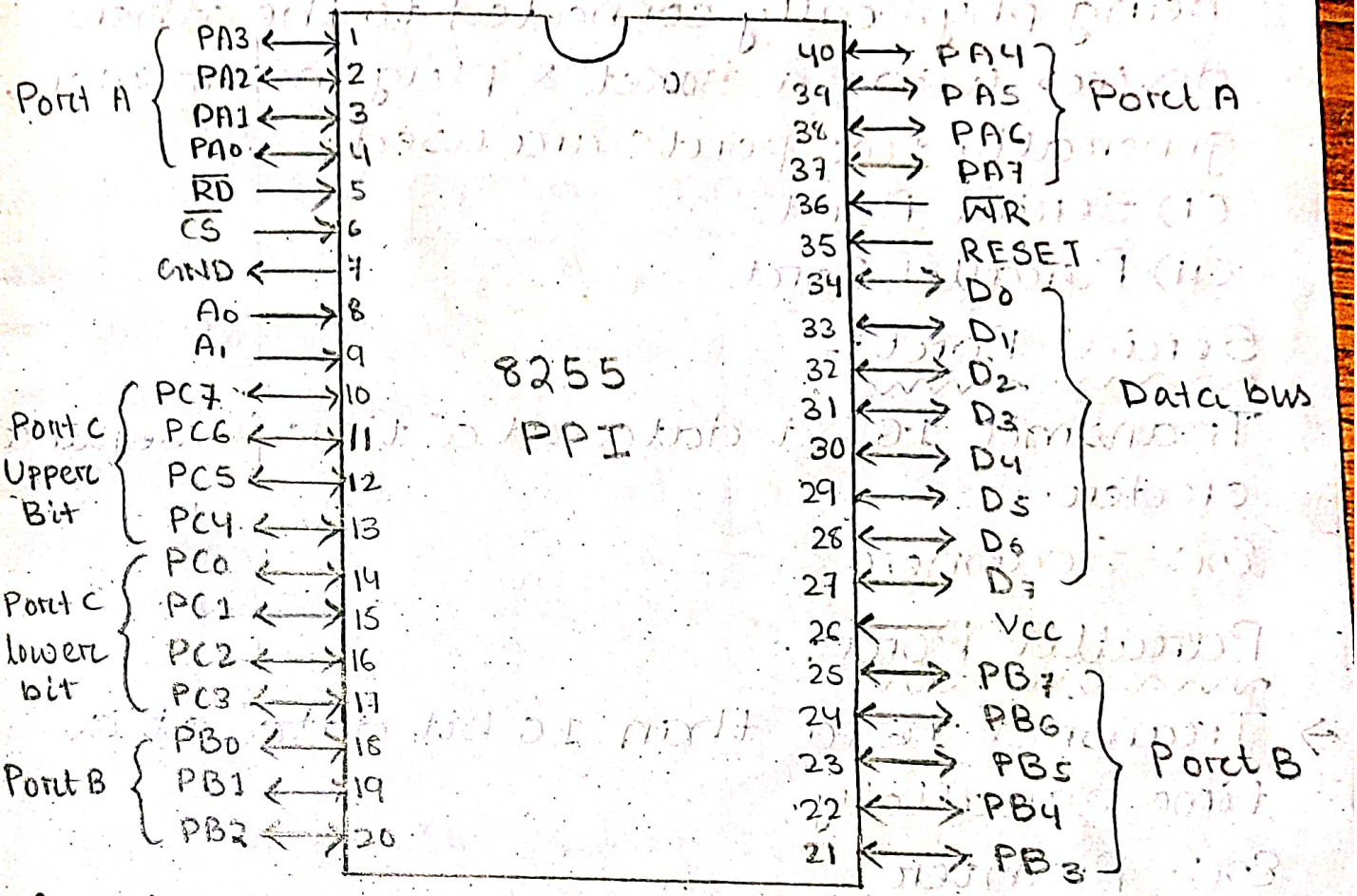
Feature of 8255 PPI :-

→ 8255 is design to work with various mp like :- 8085, 8086.

→ It is design to increase capacity of I/O interface.

→ It is used 3 ports i.e, port A, port B, port C.

Pin diagram :-



Continuing of A_0, A_1

A_0	A_1	Select
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Control register

→ The selection of I/O port & control word register is done by using A_0, A_1.

8255 is a 40IC pin package & these pins can be divided into 4 categories.

(I) Port
(II) Data bus
(III) Power Supply
(IV) Control Signal

Port :-

→ It is generally a specific place for being physically connected to the other devices using a socket & plug in computer, generally two ports are used:

(I) Serial Port
(II) Parallel Port

Serial Port :-

Transmit 16 bit data at a time in sequence order.

Ex:- Scanner

Parallel Port :-

→ Transmit more than 16 bit data at a time parallelly.

Ex:- Printer

→ 8255 has 4 Port :- (I) Port A
(II) Port B
(III) Port C
(IV) Control Signal

→ $PA_0 - PA_7$ = Port A (8 bit)
→ $PB_0 - PB_7$ = Port B (8 bit)
→ $PC_0 - PC_3$ = Port C (lower) - 4 bit
→ $PC_4 - PC_7$ = Port C (upper) - 4 bit

Data bus

→ It is a 8bit bus i.e, D_0 - D_7 & it is also called bidirectional & pin occurring 27 to 34 in the 8255 diagram.

Power Supply:-

→ 8255 PPI uses +5V power supply as Vcc & 26 no. of pin use for the Vcc & 7 no. pin as ground (GND) pin.

Control Signal:-

$\overline{CS}$ (Chip select)

→ A low i/p signal enable, the communication bet^n 8255 & CPU.

$\overline{WR}$ (Write)

It is an active low signal. It performs write operation when the signal goes low, the mp write into a selected i/p or o/p port & a control word register.

$\overline{RD}$ (Read)

It is an active low signal. It performs read operation when the signal goes low, the mp read into a selected i/p or o/p port & a control word register.

RESET

It is an active high signal. It clear the control word register & save all port in the i/p port.

A_1, A_0

These are normally connected to the lower bit of address bus i.e, A_0 - A_1 & occupying the pin 8&9.

Operating mode of 8255 :-

There are 2 different operating mode for 8255 PPI. (i) Bit set or Reset mode
(ii) I/O mode

BSR mode :-

It is mainly used to define handshake signal.

I/O mode :-

It is used for input or output data transfer.

Control word Register

→ 8255 may be operated in one of the 2 mode (BSR or I/O mode) by initializing D_7 bit.
→ If D_7 bit = 1 then it is operating in I/O mode
→ If D_7 = 0 then it is operating in BSR mode

Input or Output Mode

This mode is divided into 3 types.
(i) mode 0
(ii) mode 1
(iii) mode 2

Mode 0

→ It is a simple I/O mode.
→ In this mode Port A, Port B, Port C are used as simple 8 bit I/O port.
→ This port couldn't need/require handshaking signal.
→ There is no need of BSR mode, that's why we are using I/O mode in this mode.

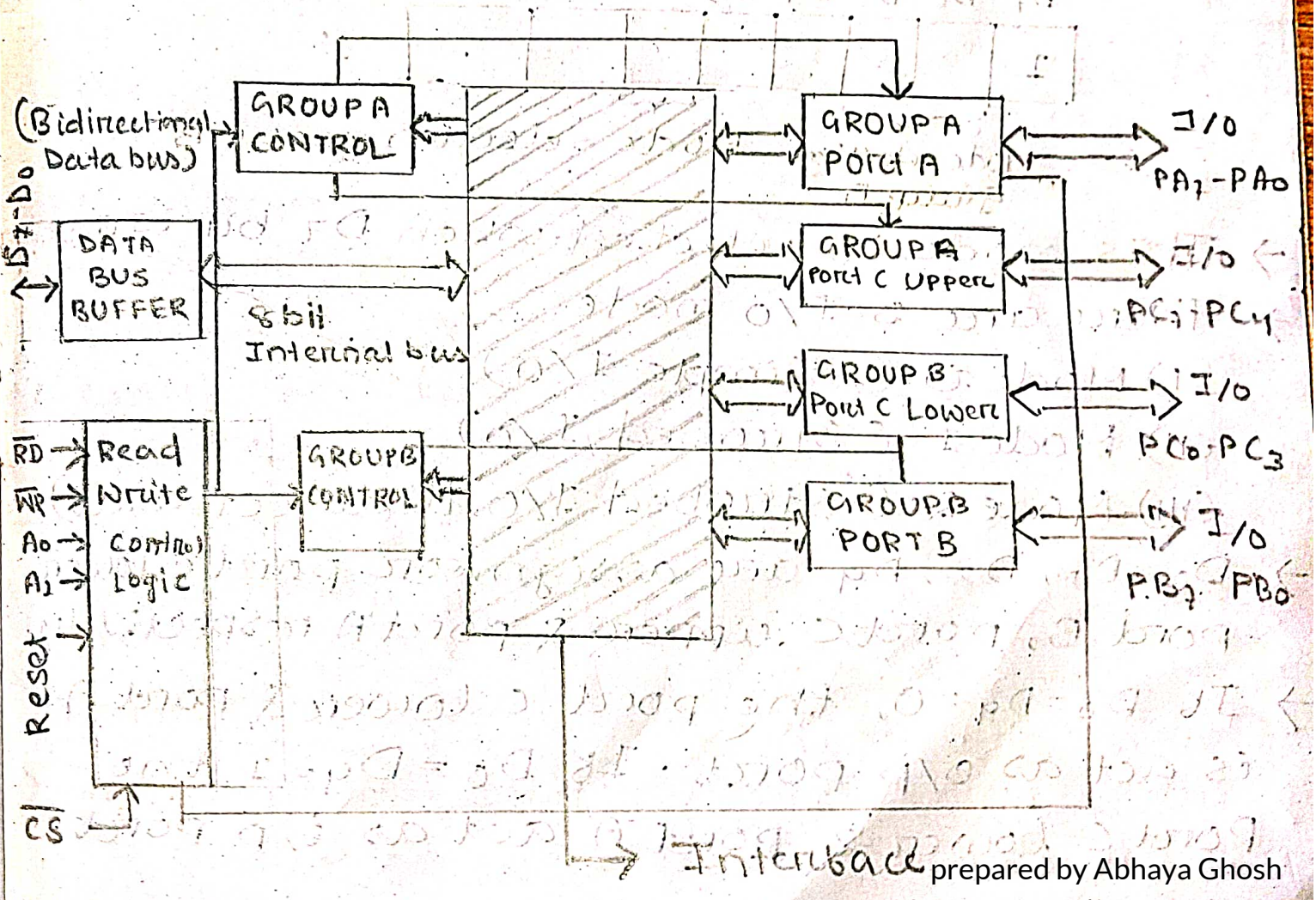
Mode 1

→ It is also known as strobed I/O mode.
→ When Group A & Group B are programmed to mode 1 then two port operated in strobed in mode, to this first all control word register to BSR mode to desire the store signal as handshaking.
→ After that the control word register is program to I/O mode.

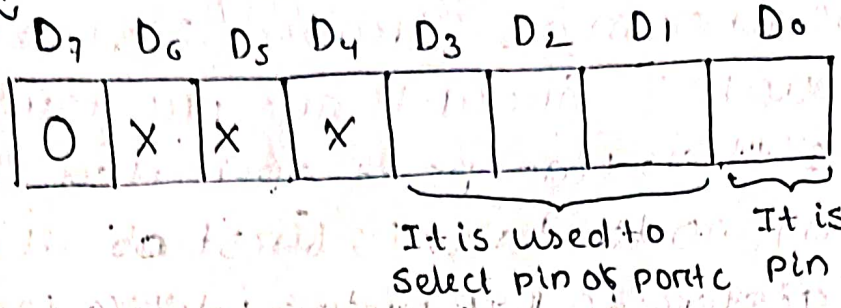
Mode 2

→ It is also known as bi-direction hand-shaking I/O.
→ In this mode port A is used as bi-direction mode & port B is either on mode 0 or mode 1.

Block Diagram of 8255 PPI



BSR



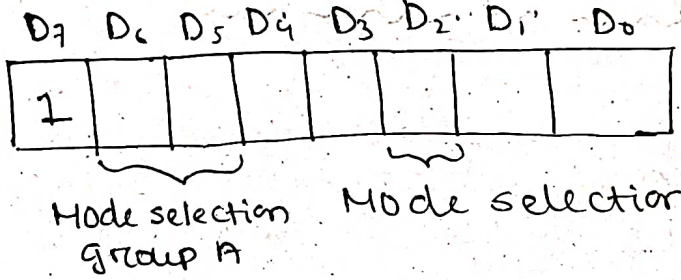
→ D_7 bit is always for BSR mode.

→ BSR mode is always applicable to port C only (PC_0-PC_7).

→ Each line of port C can be set or reset by loading in control word register.

→ Bit D_6, D_5, D_4 are unspecified. Bit D_3, P_2, D_1 are used to select the pin of port C. D_0 bit is used to set or reset the pin of port C.

I/O Mode :-



→ This mode is selected when D_7 bit is one.

→ There are 3 I/O mode
(i) Mode 0 (Simple I/O)
(ii) Mode 1 (Strobed I/O)
(iii) Mode 2 (Strobed I/O bidirectional)

→ D_0, D_1, D_3, D_4 are assign for port C lower, port B, port C upper & port A respectively.

→ If $D_0 = D_4 = 0$, the port C lower & port A is act as o/p port. If $D_0 = D_4 = 1$, the Port C lower & port A act as i/p port.

→ If $D_1 = D_3 = 0$, then port B & portc upper act as o/p port. If $D_1 = D_3 = 1$, then port B & port c upper act as i/p port.

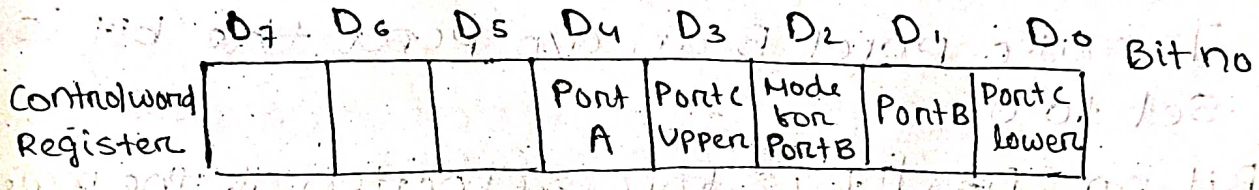
→ D_2 is used for the mode selection of group B (port B & port C lower).

→ When $D_2 = 0$, mode 0 is selected & when $D_2 = 1$, mode 1 is selected.

→ D_5 & D_6 are used for the mode selection of group A (Port A, port C upper).

→ Mode selected n bit D_2, D_5, D_6 are all 0 for mode 0 operation.

Control word



→ Control word is used for the programming port of 8255. The control word is written into the control word register.

→ No read operation of the control word register is allowed. If a particular port is to be made an i/p port the corresponding bit for the point is said to 1.

→ If a particular port is made an o/p port the corresponding bit for the port is set to 0.

Control Group

→ The 24 line of i/p port are divided into 2 groups i.e, group A & group B.

→ The group A contains port A & port C upper.

→ The group B contains port B & port C lower.

Bit no (D_0) [It is for port C lower]

→ To make port C lower an i/p port, the bit set to 1.

→ To make, port C lower an o/p port, the bit set to 0.

Bit no(D_1) [It is for port B]

→ To make port B an i/p port the bit is set to 1.

→ To make port B an o/p port the bit is set to 0.

Bit no (D_2) [It is for selection of the mode for the port B]

It is for the selection of the mode for the port B. If the port B operated as mode 0 the bit is set to 0 & for mode 1 operation of the port B, it is set to 1.

Bit no. (D_3) [It is for port C upper]

→ To make port C upper an i/p port, the bit is set to 1.

→ To make port C upper an o/p port, the bit is set to 0.

Bit no. (D_4) [It is for port A]

→ To make port A an i/p port; It is set to 1.

→ To make port A an o/p port, It is set to 0.

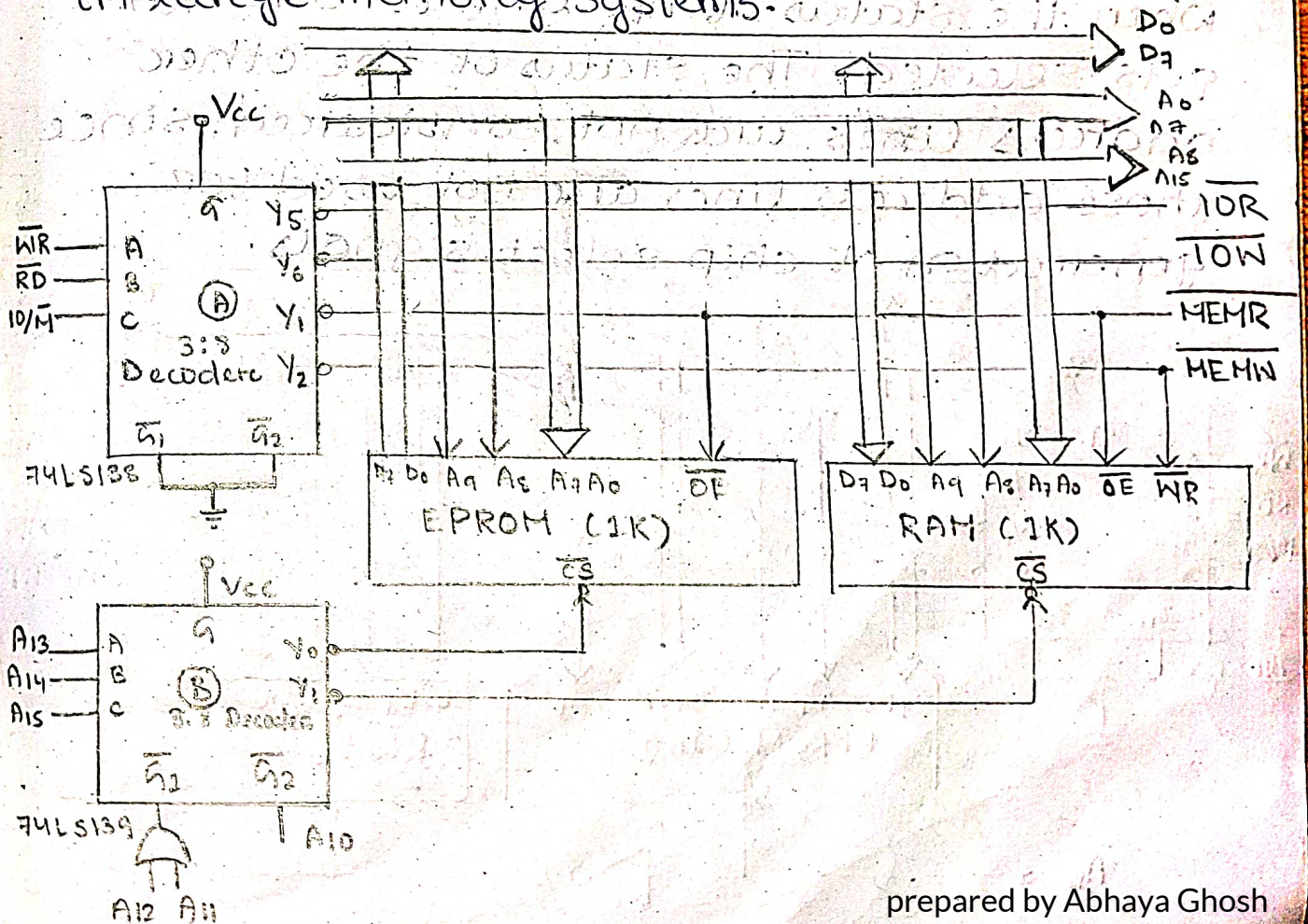
Bit no. D5, D6

→ This bit is used to define the operating mode of port A.

Mode of Port A	Bit no 6	Bit no 5
MODE 0	0	0
MODE 1	0	1
MODE 2	1	0/1

* Absolute Decoding / Full Decoding

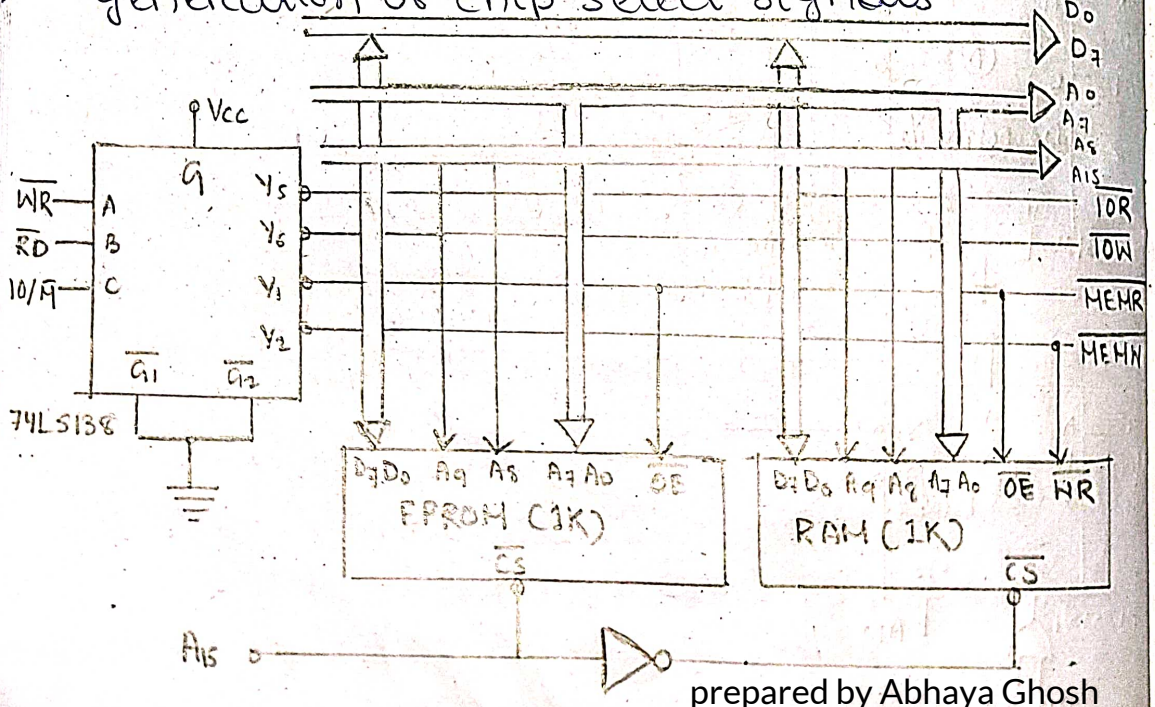
In this technique, all the higher address lines are decoded to select the memory chip & the memory chip is selected only for the specified logic levels on these higher order address lines, no other logic levels can select the chip. Below figure shows the memory interfacing in 8085 with absolute decoding. This is normally used in large memory systems.



Linear Decoding

In small systems, hardware for the decoding logic can be eliminated by using individual high-order address lines to select memory chips. This is referred to as linear decoding. Below fig. shows the addressing of RAM with linear decoding technique. This is also called partial decoding. It reduces the cost of decoding circuit, but it has a drawback of multiple addresses (shadow address).

A_{15} address line is directly connected to the chip select signal of EPROM & after inversion it is connected to the chip select signal of the RAM. Therefore, when the status of A_{15} line is 'zero', EPROM gets selected & when the status of A_{15} line is 'one' RAM gets selected. The status of the other address lines are not considered, since those address lines are not used for generation of chip select signals.



* Memory Interfacing in 8085:

→ Memory is an integral part of a mp system.

→ Memory interfacing in 8085 is used to access memory quite frequently to read instruction codes & data stored in the memory.

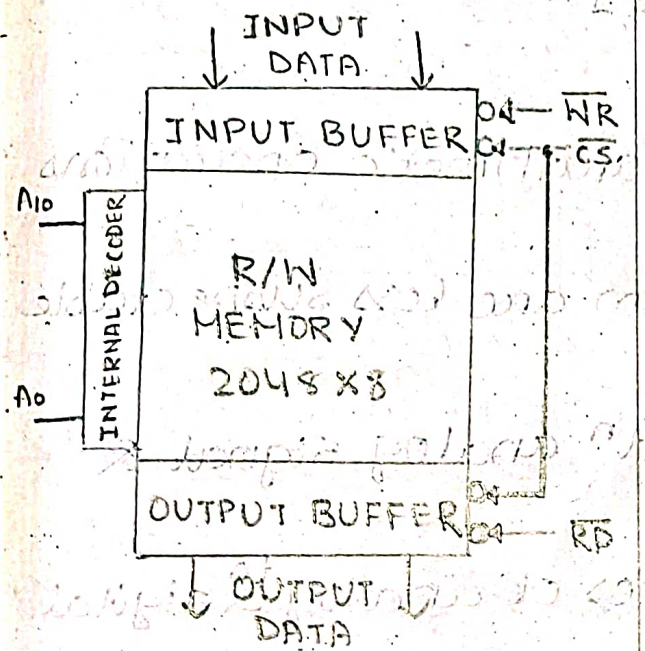
→ This read / write operations are monitored by control signals.

→ The mp. activates these signals when it wants to read from & write into memory.

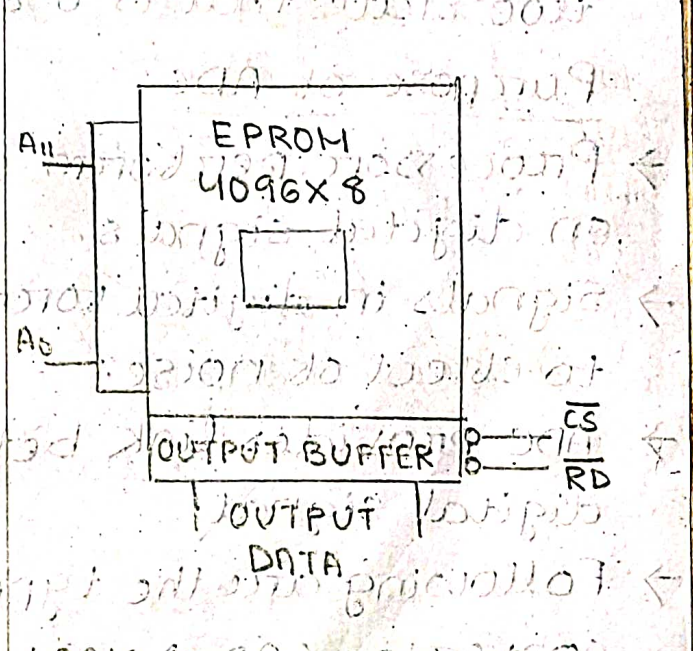
Memory structure & its requirement

→ Read/write memories consist of an array of registers in which each register has unique address.

→ The size of the memory is N x M as shown in below figure where N is the number of registers & M is the word length in number of bits.



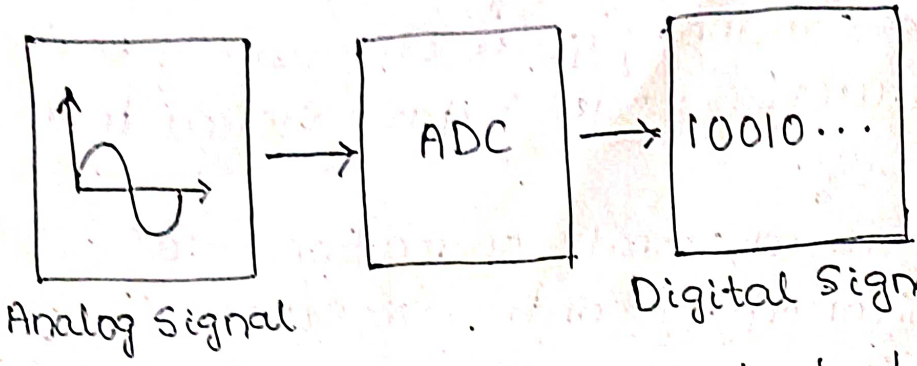
LOGIC DIAGRAM FOR RAM



LOGIC DIAGRAM FOR EPROM

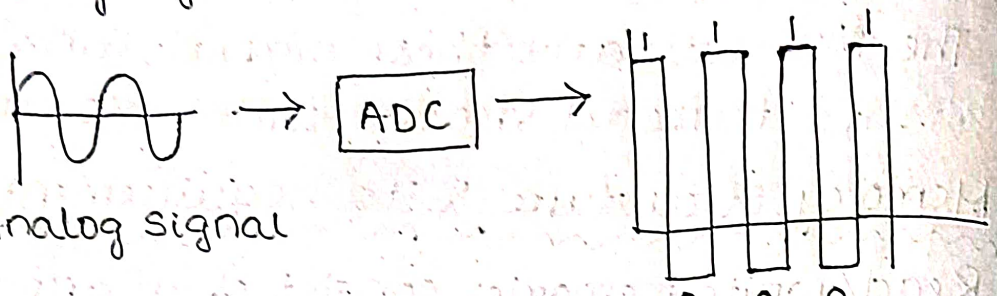
ADC & DAC with interfacing:-

ADC (Analog to Digital Converter)



Analog Signal

Digital Signal



Analog signal

Digital ~~Data~~/signal

→ It is an electronic IC which transforms analog signal (continuous) to digital signal (discrete).

→ Analog signals are directly measurable quantities while digital signals only have two states that is 0 & 1.

Purpose of ADC

→ Processors perform arithmetic operations on digital signals.

→ Signals in digital form are less susceptable to effect of noise.

→ ADC provides link betn analog signal & digital signal.

→ Following are the types of analog to digital converter (ADC):- Counter type, Successive approximation, Flash ADC, Sigma & delta.

Application of ADC :-

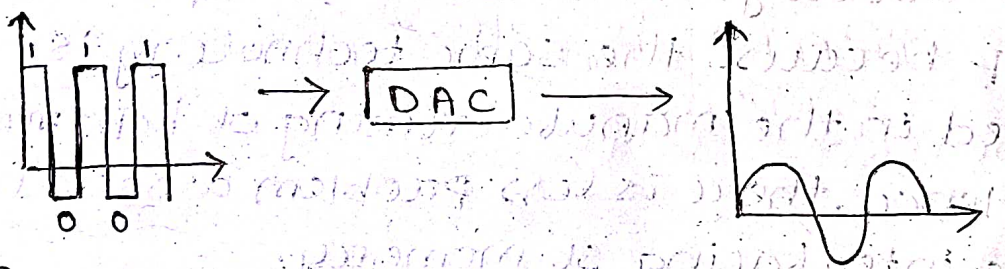
Digital voltmeter - It measures voltage in analog & convert to digital form using ADC for digital representation form.

Mobile Phone - Analog voice is converted to digital form for further processing (speech operation encoding etc.). Before it is converted, back to analog form for transmission.

Scanner - When we take photo or scan document the scanner uses ADC internally to convert analog information provided by picture / document into digital information.

Voice recorder - It uses ADC to convert analog voice information to digital information.

DAC (Digital to analog converter)



Digital signal Analog signal

→ To convert digital values to analog values.
→ It performs inverse operation of ADC.
→ There are 2 types of DAC
(i) Weighted Register
(ii) Resistive Divider

→ Following specifications are considered from the design, development as well as selection of DAC.

- Resolution
- Reference voltages
- Setting time
- Linearity
- Speed
- Errors

Application of DAC

→ Modem - Modem needs DAC to convert data to analog form. So that it can be carried over telephone wires.
→ Pointers
→ Digital audio
→ Digital motor control
→ Sound equipments

Data Transfer Schemes

In a mp-based system or in a computer data transfer takes place b/n 2 devices such as mp & memory, mp & I/O devices, memory & I/O devices. Usually, memories are compatible with mp because the same technology is employed in the manufacturing of both memories & mps. Hence, there is less problem associated with the interfacing of memory.

A mp-based system or a computer may have several I/O devices of different speed. A slow I/O device can't transfer data when mp issues instruction for the same because it takes some time to get ready. To solve the problem of speed mismatch b/n a mp & I/O devices a no. of data transfer techniques have been develop

The data transfer schemes are classified into the following two broad categories:
(i) Programmed data transfer schemes
(ii) DMA data transfer schemes.

Programmed Data transfer scheme:-
Programmed Data transfer schemes are controlled by the CPU. Data are transferred from an I/O device to the CPU or vice versa under the control of programs which reside in the memory.

These programs are executed by the CPU when an I/O device is ready to transfer data. The mp executes the program to transfer data. Programmed data transfer schemes are employed when small amount of data are tobe transfered. The programmed data transfer schemes are classified into the following 3 categories.
(i) Synchronous data transfer scheme
(ii) Asynchronous data transfer scheme
(iii) Interrupt driven data transfer scheme

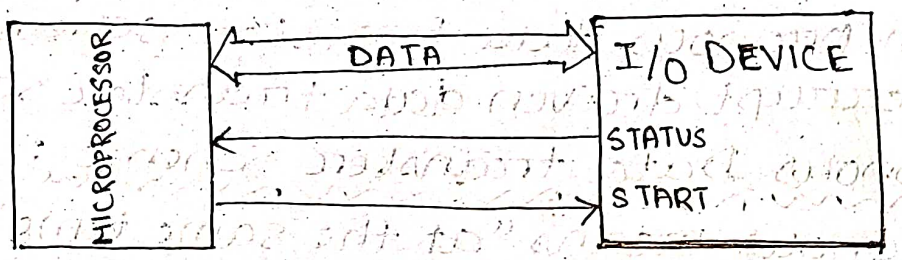
Synchronous Data transfer scheme:-
→ Synchronous means "at the same time".
→ The device which receives data are synchronized with same clock.
→ When the CPU, I/O devices match in speed, the synchronous data transfer is employed.
→ The data transfer with I/O devices performed executing IN or out instruction for I/O mapped I/O devices or memory read/write instruction for memory.
→ This technique is rarely used because I/O devices compatible with mp in speed are usually not available.

Asynchronous Data Transfer :-

→ Asynchronous means at regular intervals.
→ In this method, data transfer is not based on predetermined timing pattern.
→ This technique of data transfer is used when the speed of an I/O device doesn't match the speed of the mp.
→ In this technique, the status of the I/O device i.e, wheather the device is ready or not checked by the mp before the data transfer.

Purpose of Asynchronous Data transfer

→ Mp check the status signal before the data transfer.
→ Then mp indicates the I/O device to get ready.
→ When I/O device get ready the mp sends instructions to transfer the data.



→ A keyboard interfaced to a mp is an example of this type of data transfer scheme.

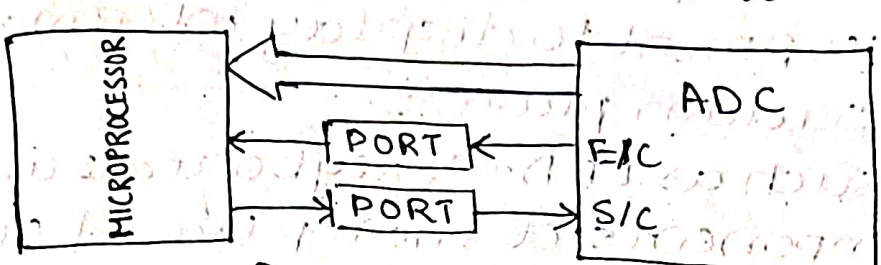
Asynchronous Data transfer scheme of an A/D converter

→ Asynchronous data transfer is used for slow I/O device.
→ An A/D converter has been transferred to the mp to transfer data in asynchronous mode.

Process:

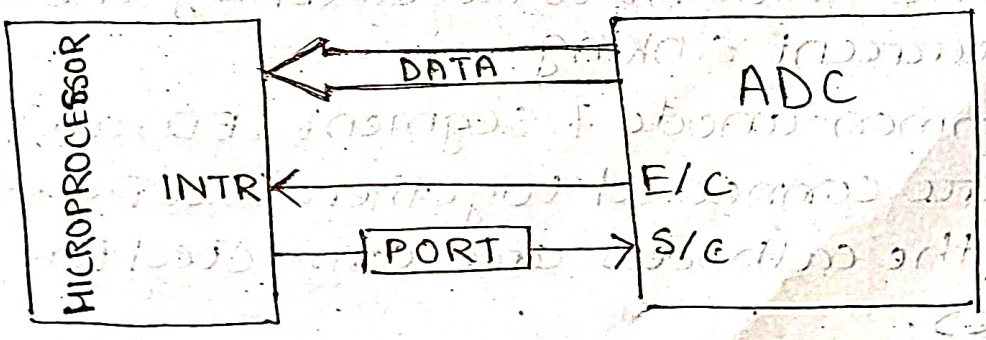
→ The mp sends a start of conversion signal S/C to the ADC.

→ The ADC is so slow therefore it take more time to convert analog signal to digital signal.
→ When conversion is over ADC make end of conversion signal. It means E/C signal is high.
→ When E/C signal becomes high the mp issues instruction for data transfer.



Interrupt Driven Data transfer:-

→ In this scheme the mp initiates an I/O device to get ready & then it executes its main program instead of remaining in a program loop to check the status of the I/O device.
→ It just interrupt the normal processing sequence of the mp.
→ On receiving an interrupt the mp complete the current instruction at hand & then attends the I/O device.
→ It saves the content of the Program counter on stack then take a subroutine called ISS.
→ It executes ISS to transfer data from or to the I/O device.
→ Different ISS are to be provided for different I/O device.
→ After completing the data transfer the mp return back to the main program.



Interfacing Seven Segment Displays

→ Interface the 8085 mp system with 7 segment display through the programmed I/O port 8255.

→ 7 segment displays is often used in the digital electronic equipment to display information regarding certain process.

→ I/O devices such as LEDs & Keyboards are essential components of the mp based or microcontroller based system.

→ 7 segment LEDs often used to display BCD no.s (0 to 9) & a few alphabets.

→ A group of 8 LEDs physically mounted in the shape of the no eight plus a decimal point.

→ Each LED is called a segment & labeled as 'a' through 'g'.

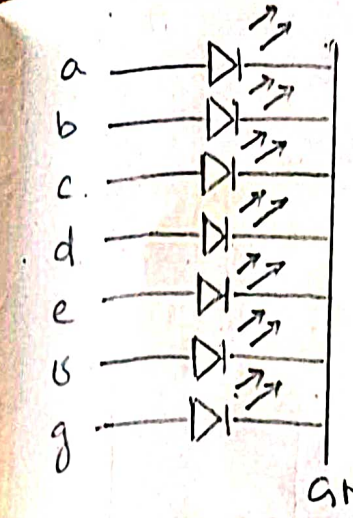
→ Commonly used o/p peripherals in embedded systems are LEDs, 7 segment LEDs & LCDs, the simplest is LED.

Two ways of connecting LEDs to I/O ports

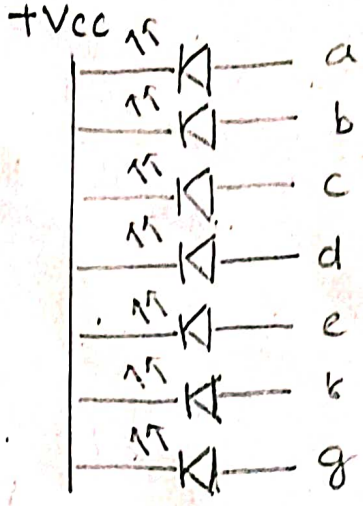
(i) LED cathode are grounded & logic 1 from the I/O port turns on the LEDs. The current is supplied by the I/O port called current sourcing.

(ii) LED anodes are connected to the power supply & logic 0 from the I/O port turns on the LEDs. The current is received by the chip called current sinking.

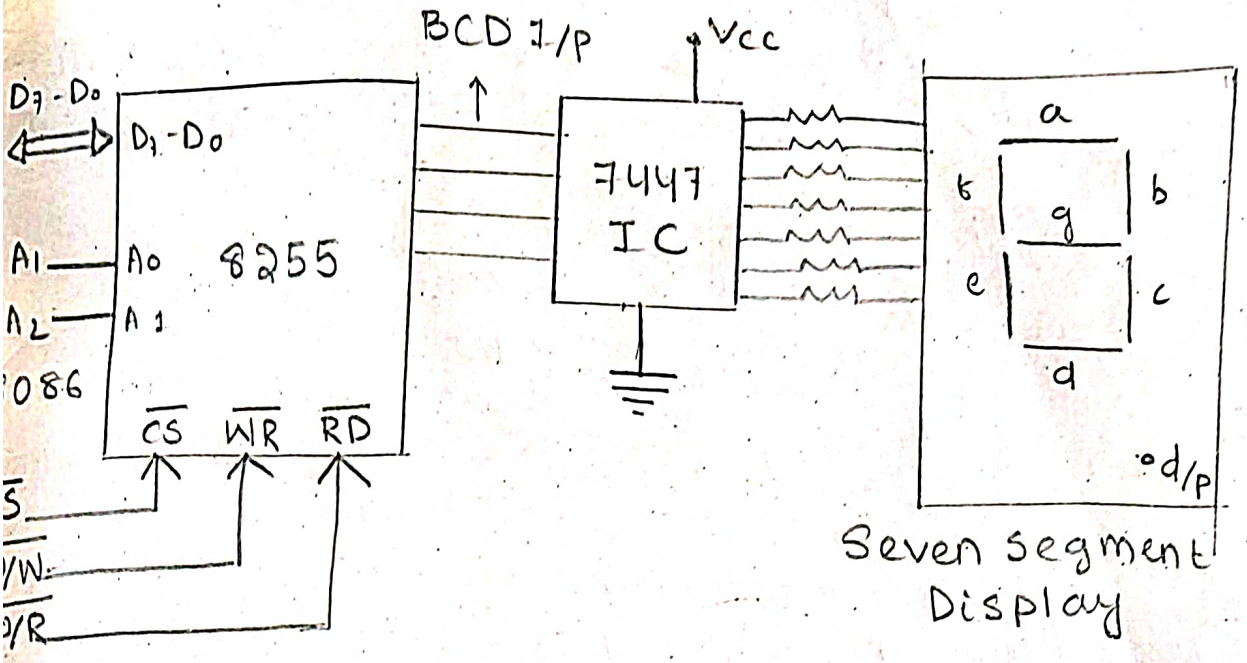
→ In a common anode 7 segment LED, all anodes are connected together to a power supply & the cathodes are connected to data lines.



Common anode



Common cathode



Seven segment Display

4.10

* Interfacing Stepper Motor with 8085

→ Stepper Motor is an electromechanical device that rotates through fixed angular steps when digital inputs are applied. It is suitable for precise position, speed & direction control which are required in automation system.

→ The angle through which stepper motor rotates with a fixed angle for each digital data is called step angle.

→ Different Stepper motor has different step angle. The more frequently used stepper motor has step angle of 0.9 degrees & 1.8 degrees.

→ Depending on the sequence applied to stepper motor, it can be classified in 2 categories:

(I) 4-Step Sequence / Full step Sequence

(II) 8-Step Sequence / Half step Sequence

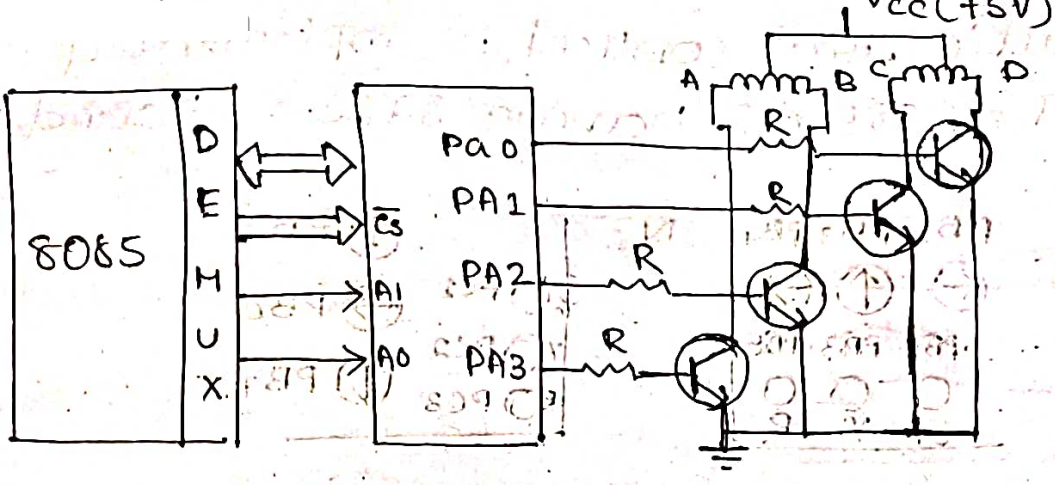
Excitation Sequence for clockwise & anti-clockwise rotation of stepper motor

Motion	Steps	A	B	C	D
Clockwise	1	1	0	0	0
	2	0	1	0	0
	3	0	0	1	0
	4	0	0	0	1
	5	1	0	0	0

Motion	Steps	A	B	C	D
Anti-clock wise	1	1	0	0	0
	2	0	0	0	1
	3	0	0	1	0
	4	0	1	0	0
	5	1	0	0	0

Program:-

```
LXI SP, 2800H
MVI A, 80H  (Initialize 8255)
OUT 83H (CWR)
MVI B, 32H
Repeat: LXI H, 2100H
        MVI C, 04H
Back:   MOV A, M
        OUT 80H (Port a)
        CALL Delay
        INX H
        DCR C
        JNZ Back
        DCR B
        JNZ Repeat
        HLT
```

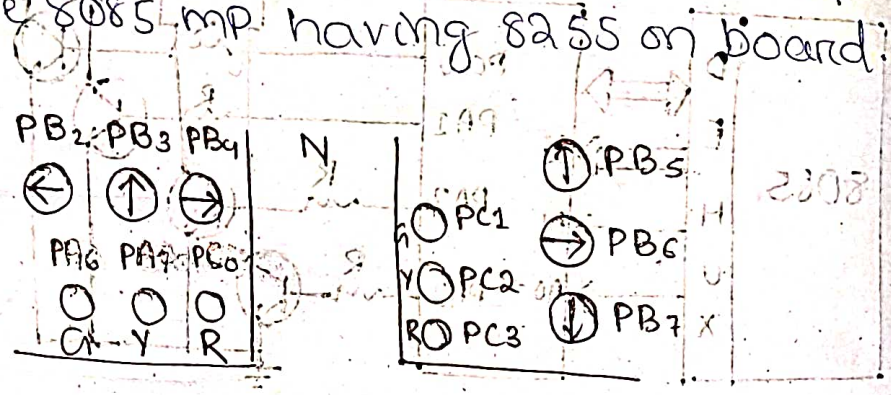


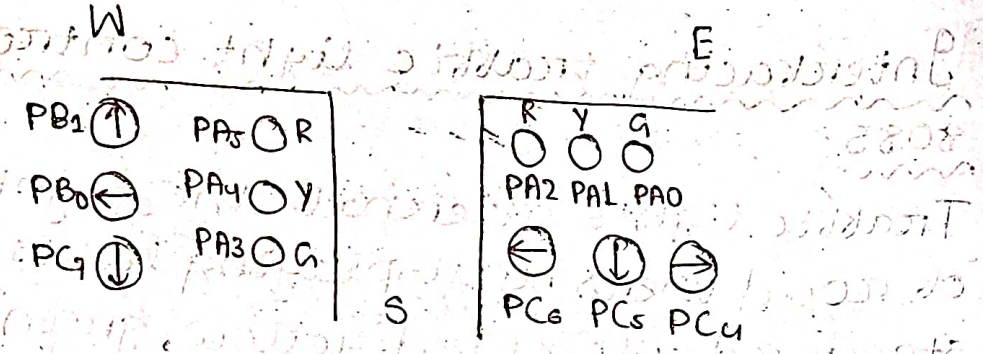
4.9 Interfacing traffic light control with 8085

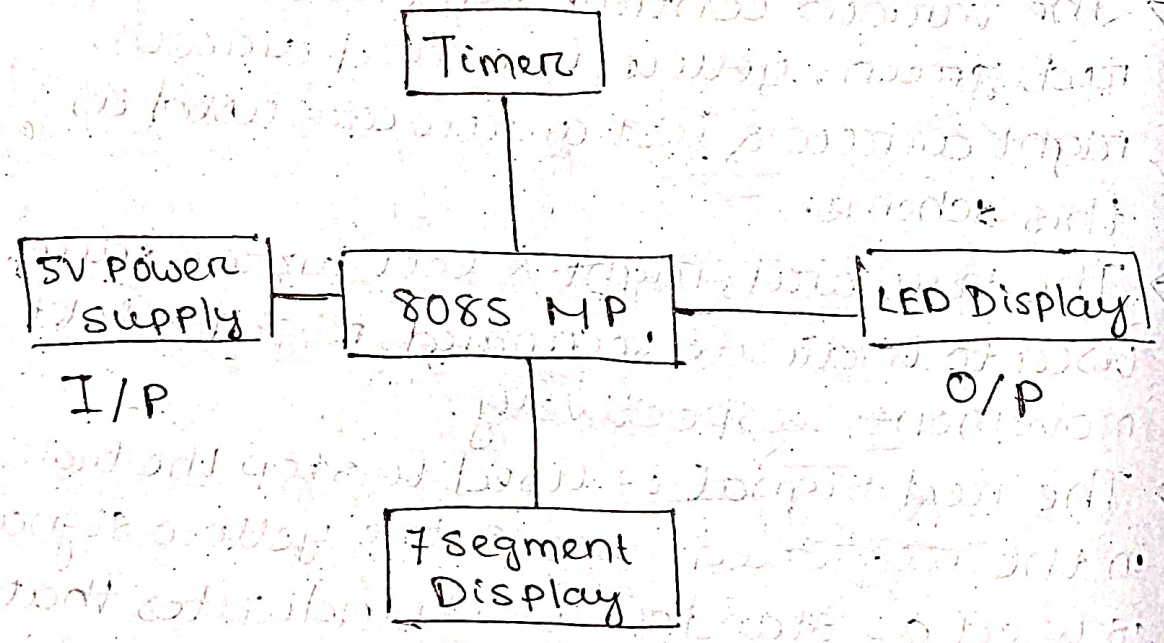
→ Traffic lights alternate the right of way of road users by displaying lights of a standard color (red, yellow, green).

→ The various control signals such as red, green, yellow, forward arrow, right arrow & left arrow are used in this scheme.

The traffic light control is implemented using the 8085 mp having 8255 on board:



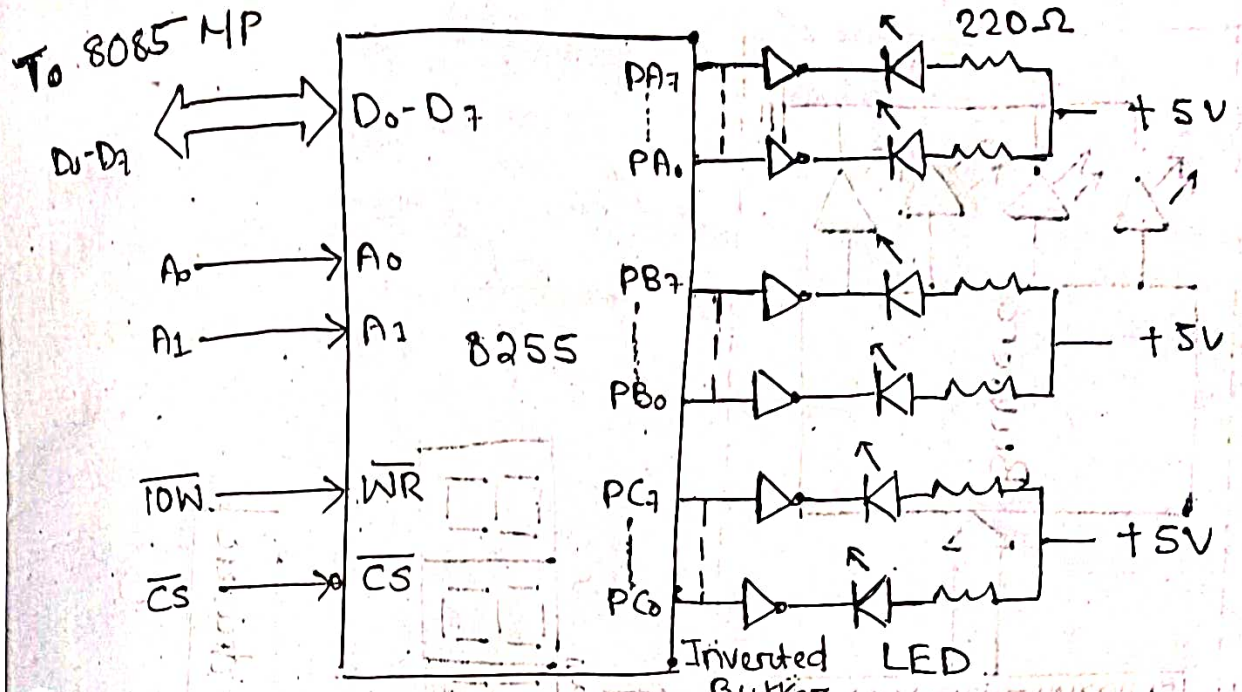




Displays waiting Time

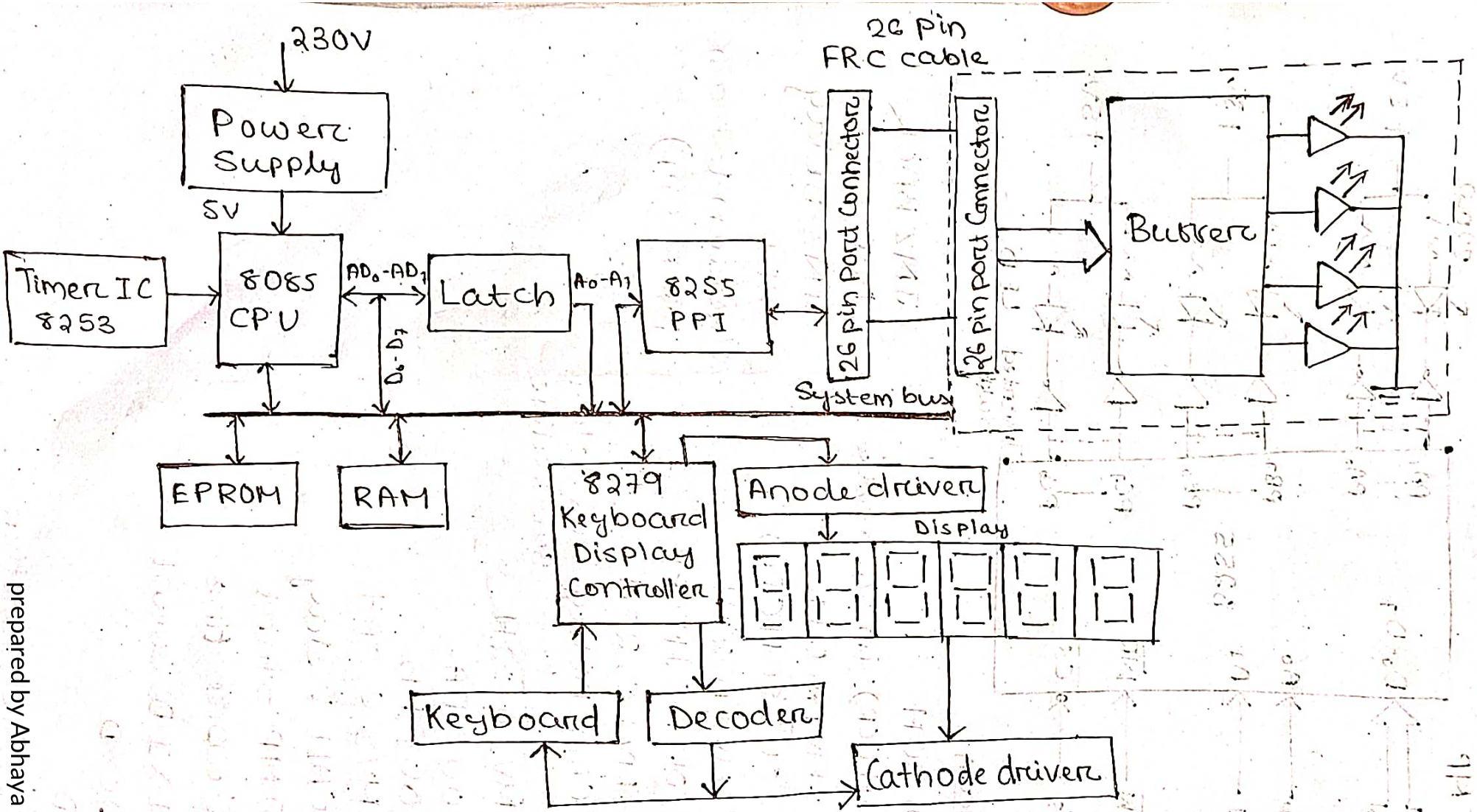
Each signal is controlled by a separate pin of I/O ports. The total no. of logic signals required for this arrangment is 24. The PPI device 8255 is used to interface these 24 logic signals with the lamps. The logic '0' & '1' represent the state of lamp. Logic '1'

represents ON & '0' represents OFF. All ports of 8255 are used as o/p ports.

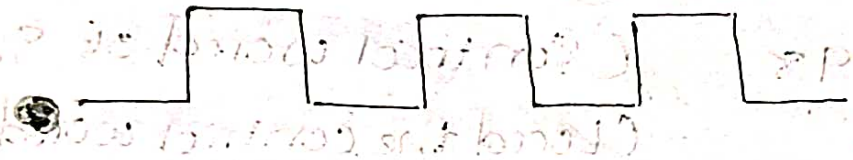


Program

```
MVI A, 80H
OUT 83H (CR): in O/P mode
Start : MVI A, 00H
        OUT 80H (PA)
        MVI A, 24H
        OUT 81H (PB)
        MVI C, 28H
        CALL Delay: call delay
                    subroutine
        MVI A, 12H
        OUT PA
        OUT PB
        MVI C, 0AH
        CALL Delay
        JMP Start
Delay Subroutine
Delay: LXI D, count
BACK: DCX D
      MOV A, D
      ORA E (Check whether
              count 0)
      JNZ BACK
      DCR C
      JNZ Delay
      RET
(Return to main
 program)
```

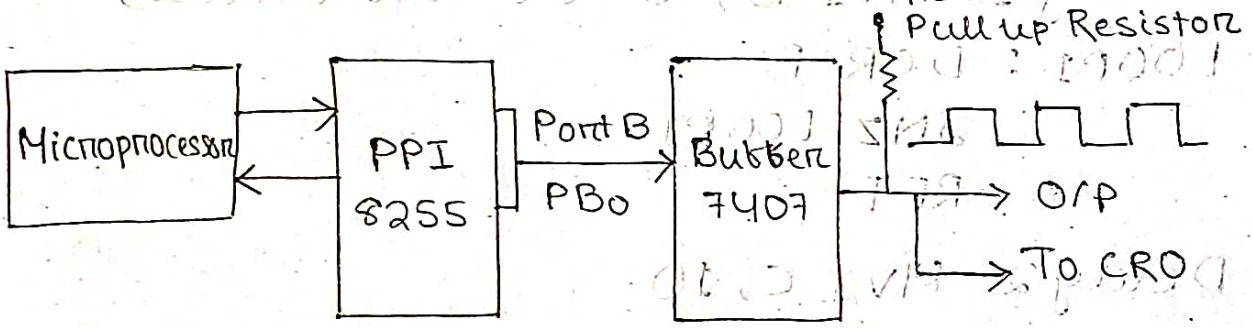


Generate Square waves on all lines of 8255



→ A square wave is a non-sinusoidal periodic waveform in which the amplitude alternates at a steady frequency betn fixed minm & maxm values, with the same duration at minm & maxm.

→ A square wave or pulse can easily be generated by mp. The mp sends high & then low signals to generate square wave or pulse. It can be generated using I/O port or SOD line or timer/counter.



The pin PB0 of the Port B of 8255 is used for taking O/P. This is connected to a buffer 7407. The final O/P is taken from the buffer terminal. The control word used in the program is 98H to make Port B an O/P port.

Program.

```
LXI SP,6000  (Initialize SP to use CALL)
MVI A,98     (Control word of 8255)
OUT 83       (Load the control word onto 8255)
LOOP: MVI A,00  (Data for low pulse)
      OUT 81    (Send PB0 as low)
      CALL Delay1
      MVI A,01
      OUT 81    (Send PB0 as high)
      CALL Delay2
      JMP LOOP
```

Subroutine

```
Delay1: MVI B,10  (Count value)
LOOP1: DCR B
       JNZ LOOP1
       RET
Delay2 MVI C,10
LOOP2: DCR C
       JNZ LOOP2
       RET
```