

16/11/18

we add decoder so that we can use less no. of lines \rightarrow Because from 2 i/p we can have 8 o/p \rightarrow we get the profit of resource saving.

let's suppose we have Memory like \rightarrow

8000
87FF
1000
17FF
4000
47FF

\rightarrow Here too many shadow memory \downarrow
Partially decoded

\rightarrow 6K memory with Partially decoded memory.

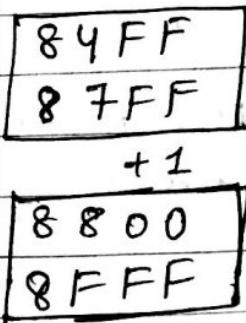
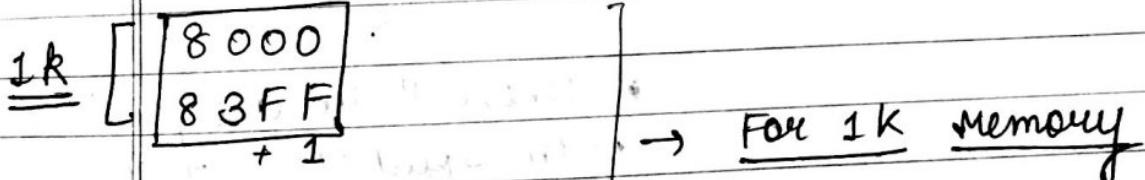
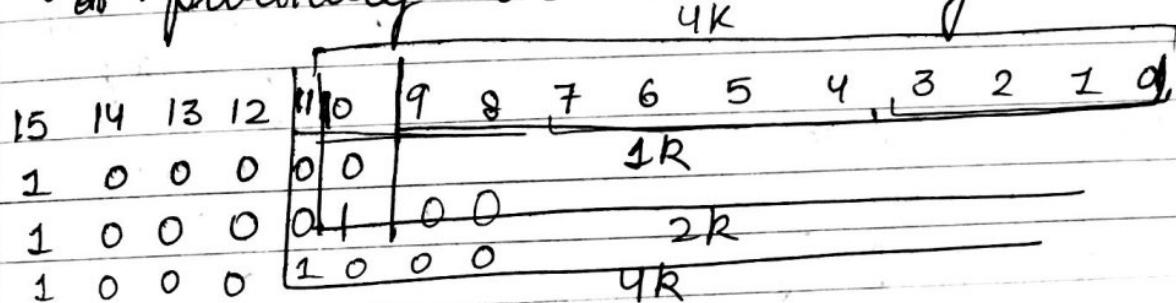
8000
87FF
9000
97FF

\rightarrow less shadow memory here

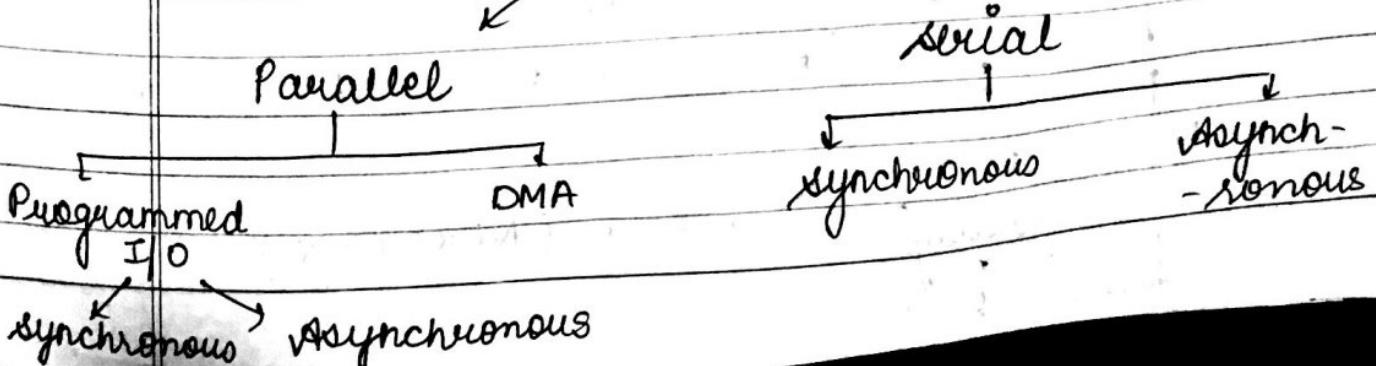
* with the help of select lines input lines are selected. (In case of multiplexer).

→ Those who have variations in them are selected as input lines and those points which are constant are used as select lines.

→ with the help of block diagram we can't judge whether we have fully decoded or partially decoded memory.

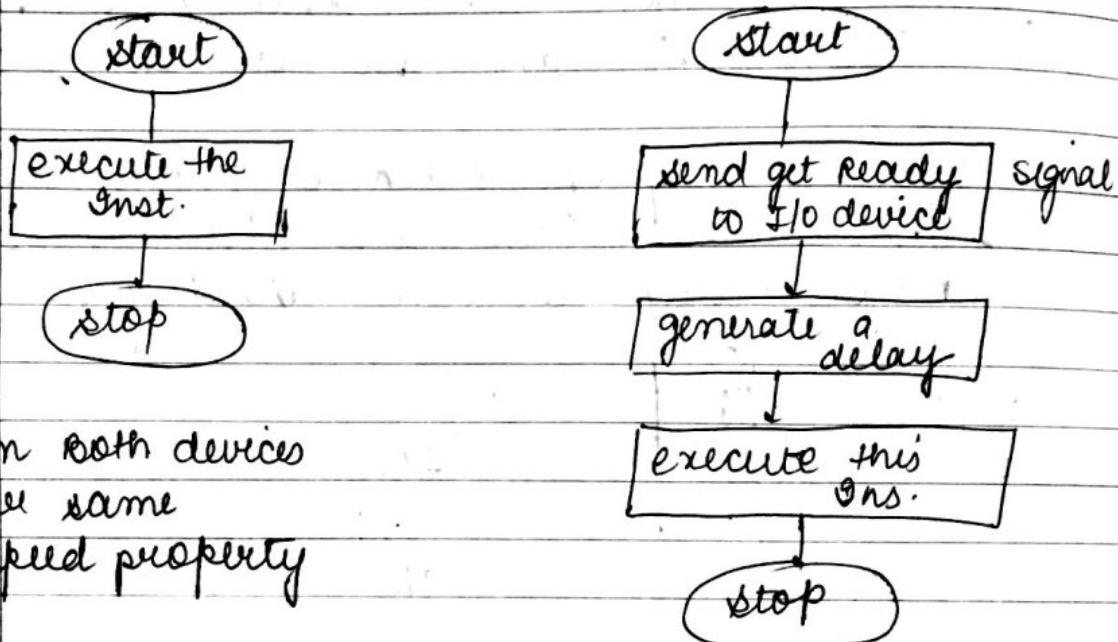


Data Transfer :-



Synchronous Mode of Data Transfer :-

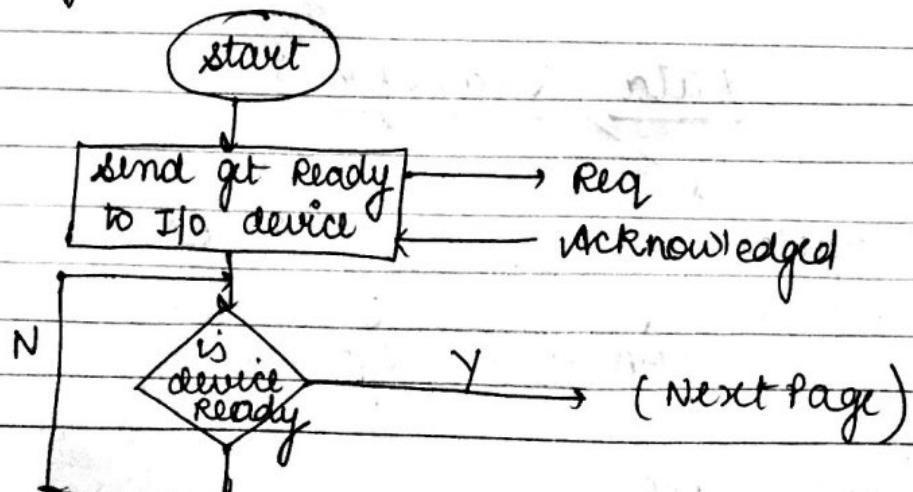
- compatible by speed
- If speed characteristic is precisely known

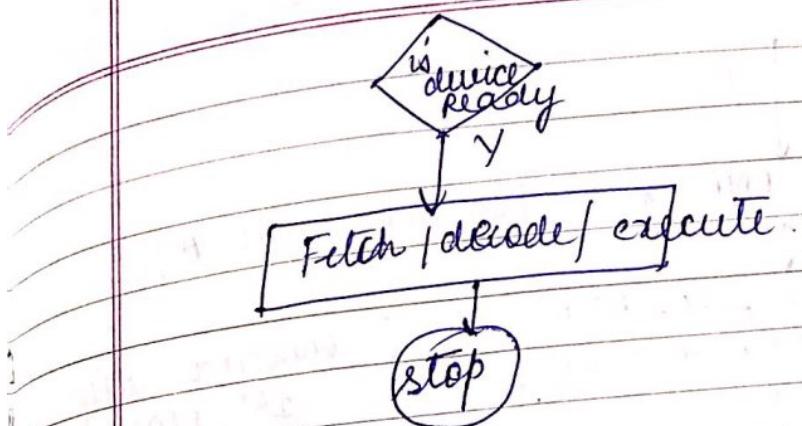


→ when there are problems with speed of 2 devices

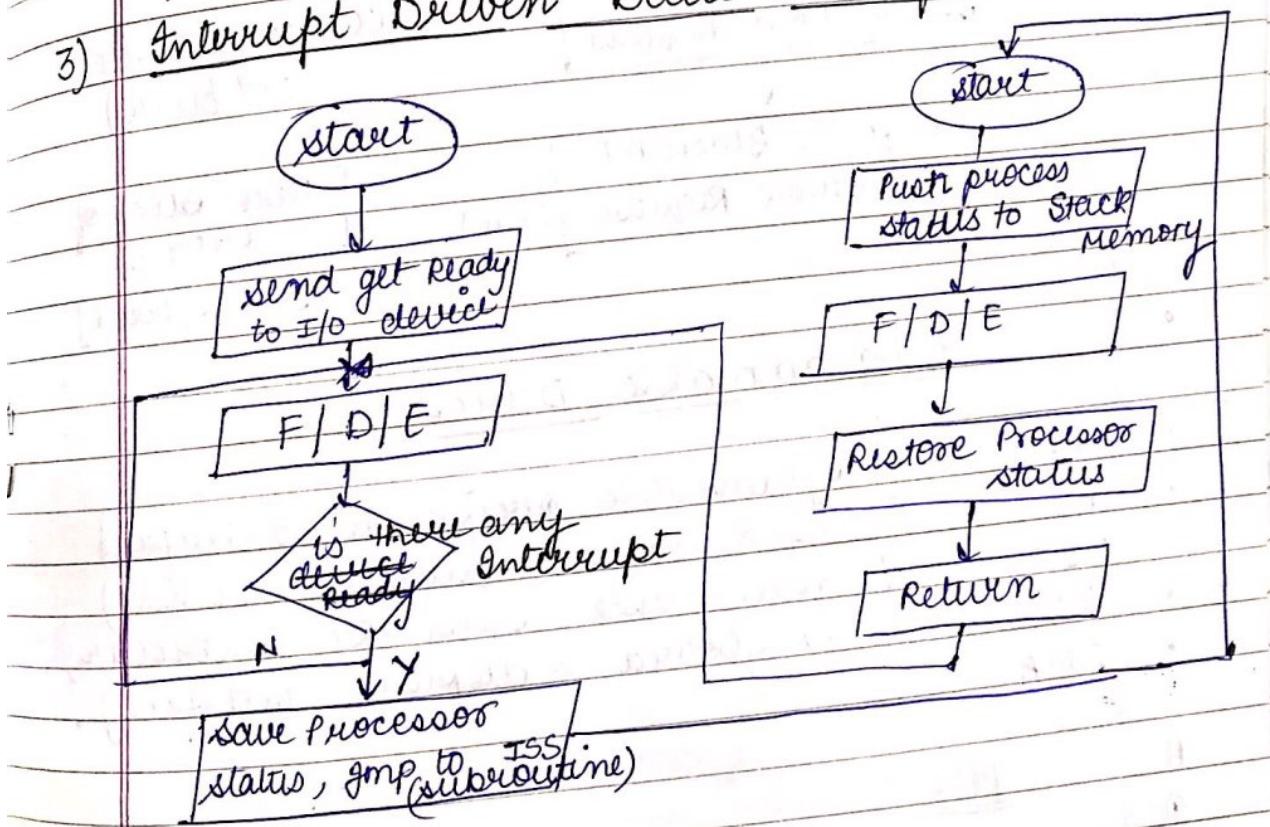
- If two diff events occurs at same frequency or clock that is synchronous event.

Asynchronous Mode :-

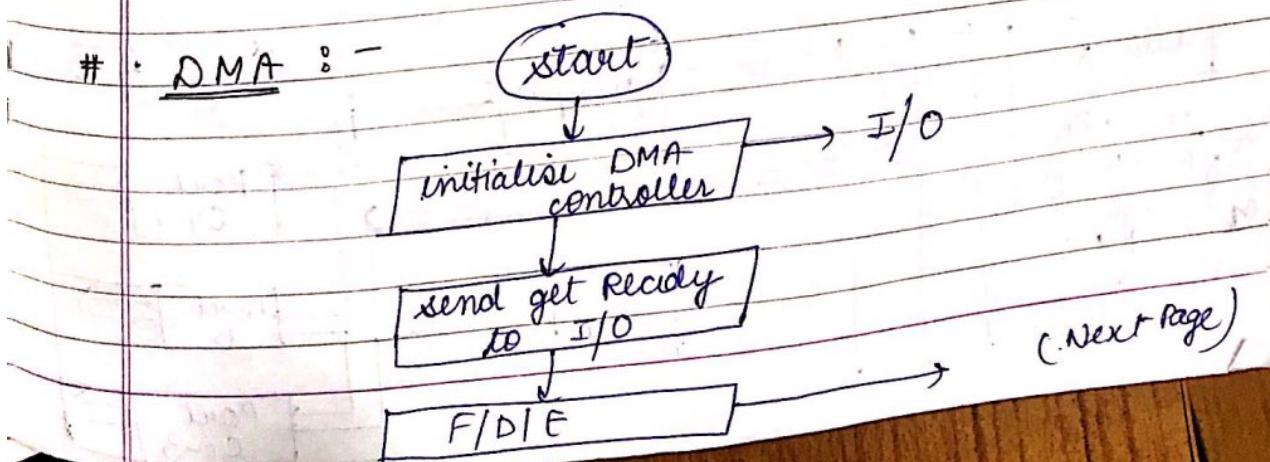


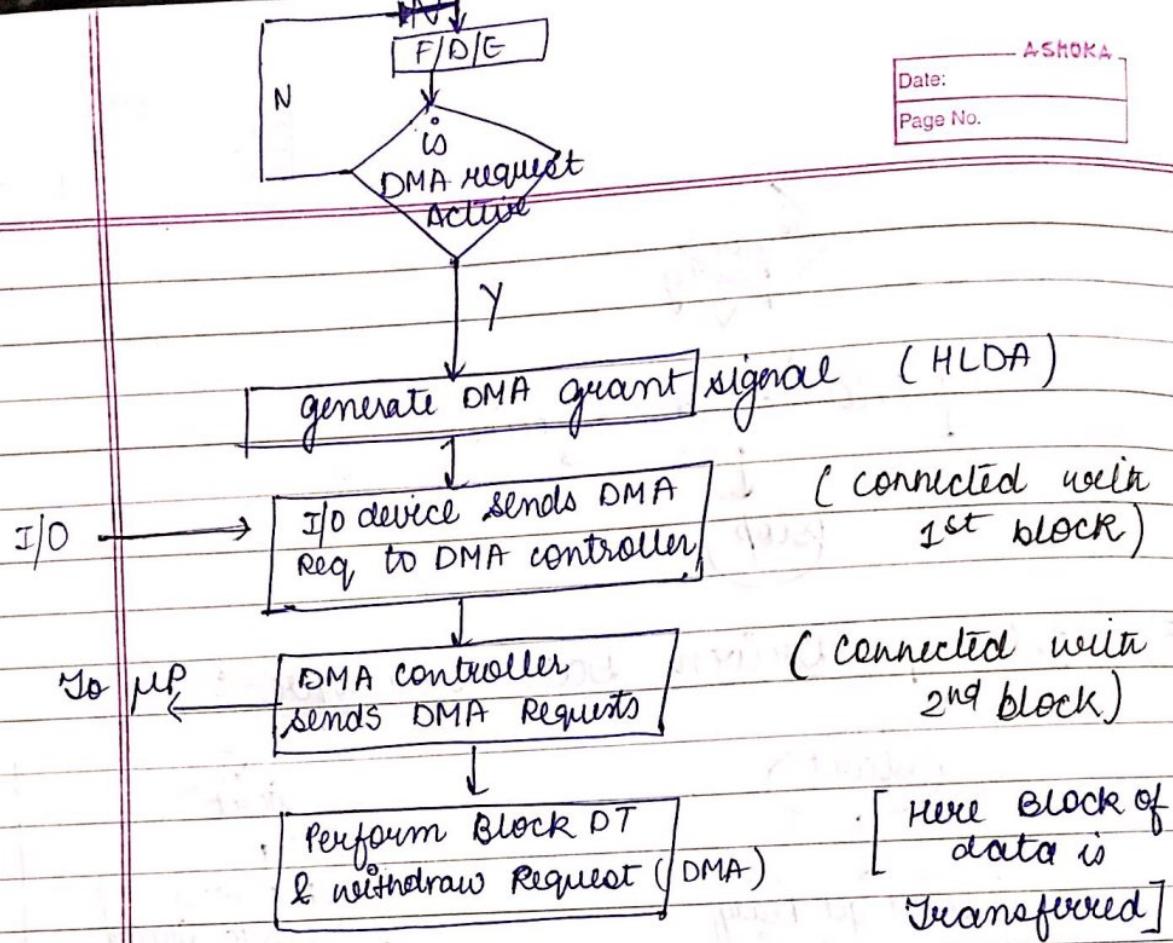


3) Interrupt Driven Data Transfer :-



DMA :-

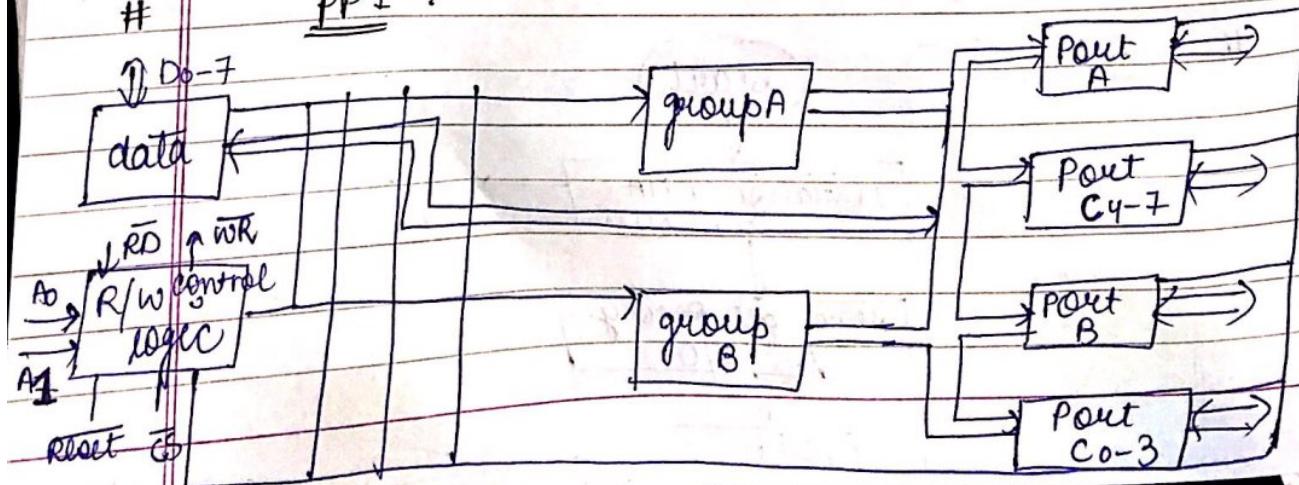




Programmable Devices :-

- 8255 : PPI (programmable peripheral Interface)
- 8259 : PIT (programmable Interval Timer)
- 8279 : PIC (programmable Interrupt controller).
- 8279 : KDI (Keyboard & display Interface).
- DMA

PPI :-



Programmed I/O / Programmable Devices :

- for 8259 specialisation is handling the interrupt

→ PPI : (8255 device)

When we have 555 timer which can work in 3 different modes than why we use programmable timer because In this system is fixed but in 555 timer we need to change the hardware again and again.

Programmable devices are our prime choice.

→ Peripheral → Input / output (By program we try to control I and o/p)
This device has 3 ports → divided in 2 groups.

- To instruct the chip that we need Port A or B or C or any Resources that we want to use via the chip

How to do Port selection ?

CS	A0	A1	
0	0	0	Port A (accessed)
0	0	1	Port B
0	1	0	Port C
0	1	1	<u>control word</u>

By this we can't access multiple ports at the same time. we can access one at a time.

control word is used to tell what we want to do with the Port we accessed. control register can be used to store the data.

All the Ports and control words are Register

Active mode → Go tell the μP what kind of commu.
all want to do and device is Ready
to perform that.

Date: _____
Page No. _____

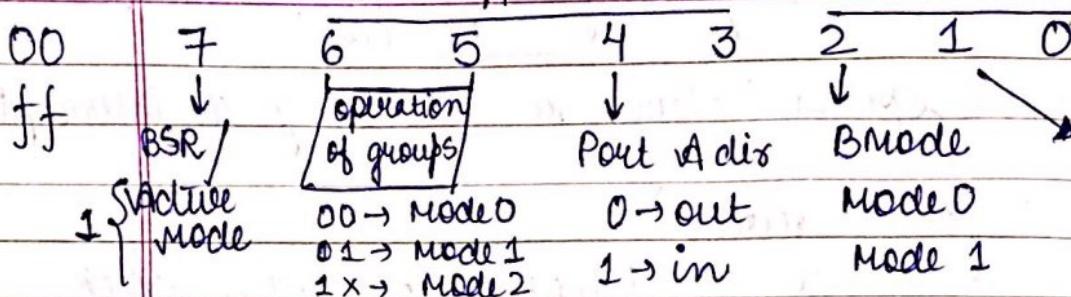
ASHOKA

Control word :-

A

B

control register



eg here we have address above 80 it means device is Ready. BSR is mode which is used only for Port C.

6 and 5 → used to tell the mode of group A in which mode we want to use it.

By 4 direction of port A

By 3 direction of Port C 4-7

[0 → out] [1 - In]

By 0 → Dir of PC 0-3 0 → out 1 → In

Port B works in 2 modes only because by one pin we can just set two modes.

eg

MVI A, 80

out ③ → It mean we need to send it to control word.

MVI A, 80

out ③ → By this bit we get to know about the port at which we need to send data.

let's see

① 0 9 0 0000

Mode → Active

Group A → Mode A

dir A → 0

dir PC 4-7 → 0

Group B → mode 0

Dir B → out

Dir PC 0-3 → out

By
Port C
we are
accompli
hands
mode

for F4
1111 0100

mode → active

group A → Mode 2

dir A → Input

Dir PC4-7 → output

Group B → Mode 1

dir B → output

dir PC3-0 → output

* difference b/w mode and direction →

1) direction is simple just I/p and o/p.

2) Mode / BSR
 \ Active / Mode 0
 \ Mode 1
 \ Mode 2

→ Mode 0 is simple I/O mode. → It have 3 ports A, B, C all of 8 Bits and use all ports and direction of all the ports is defined by Bit 4. How PPI will behave ~~is~~ is told us by Mode Number and in which direction we need to use them for that use another bit.

→ Mode 1 is Handshake I/O → Port A used for I/O data transfer

By Port C we are in Handshake mode → Firstly we will tell it by control signal that we are accompanying sending data and then by another control signal it will acknowledge us and it will disable all the write its signals. These type of DT are more trustworthy and Reliable. Signals used are called Handshake signals.

* same with group B. (in mode 1)

→ Mode 2 :- Bidirectional data by Port A . Port B either of 1 mode , Port C → for Handshake mode only group A can transfer data in Bidirectional mode. Here nobody knows initially what is the mode of direction and one bit it will get cleared whether it will be IN or output .

19/11/18

Port Address (011)

Port A → 0 (O/P) }
Port C → 0 (O/P) } MOB
(4-7)Slides :-Port B → 1 } Lower
Port C (0-3) → 1 } order Bytes

- 83 → 8003 (control Register)
 - Read 8001 (from Port B)
 Write 8000 (to Port A)

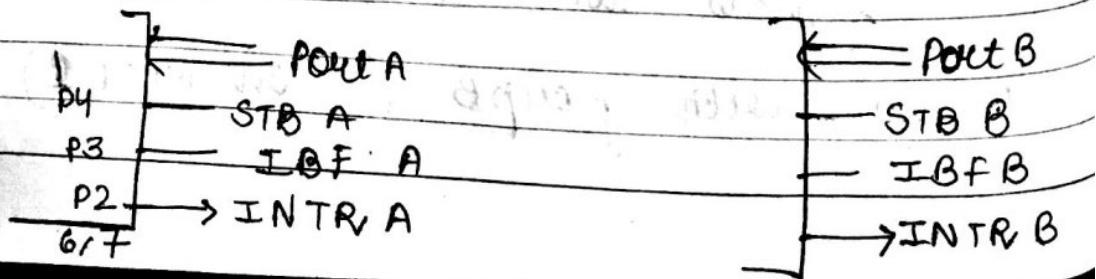
83 should go to control Register .

→ STB → Strobe [Basically its Request It Means S/S is requesting for data Transfer].
 IBF → Input Buffer Full .

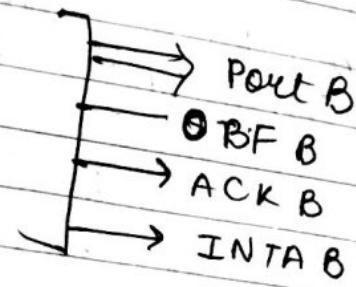
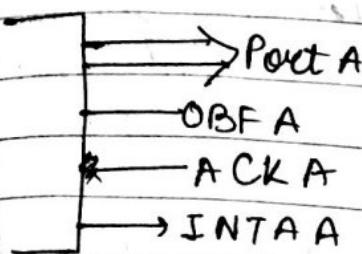
INTR A →

6 to 7 are not used .

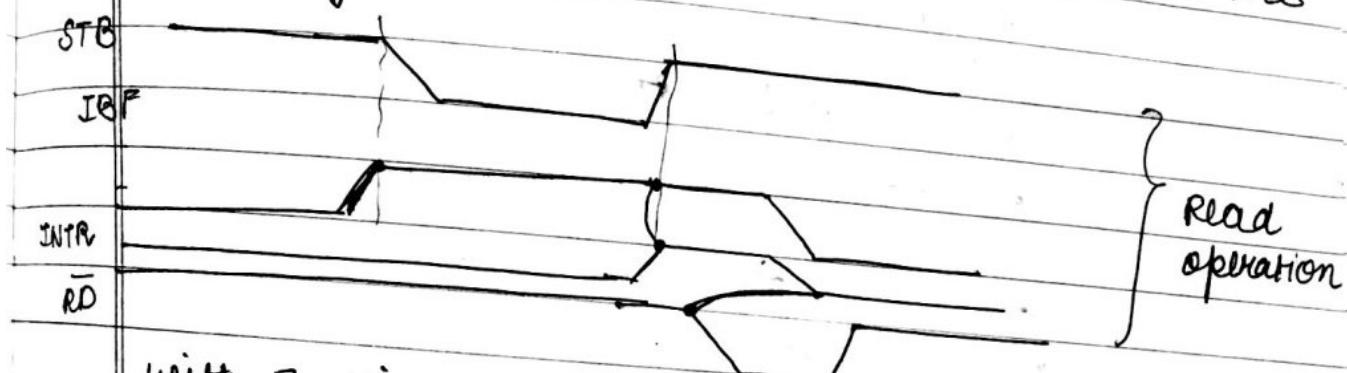
PC4 and PC5 used as control signals .

Mode 1 Port A :-

In data transfer, data can transfer in both directions.



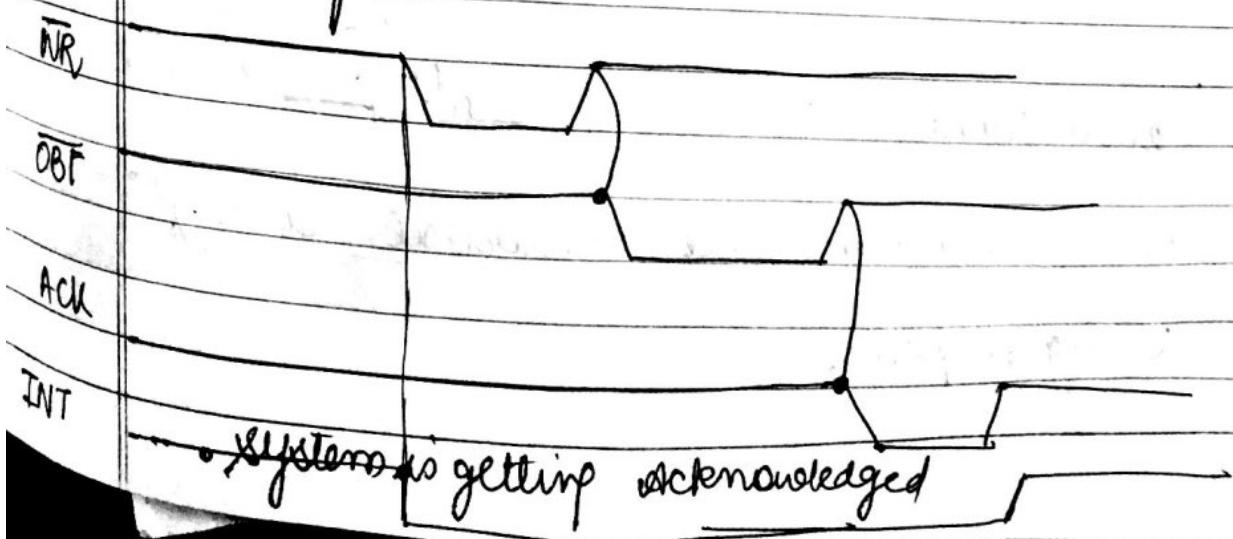
→ Stroke is a trigger signal that initiates the read signal.



with 3 signals Normal I/O
with INTR Interrupt driven I/O.

until Buffer is full we can't do Read operation
when Buffer is full and we get INTR then
Read operation gets started.

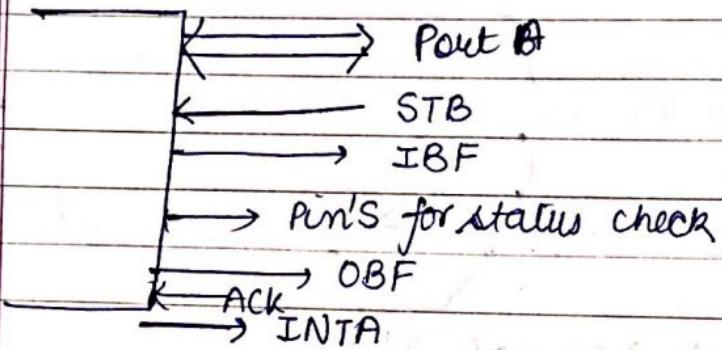
Write operation :-



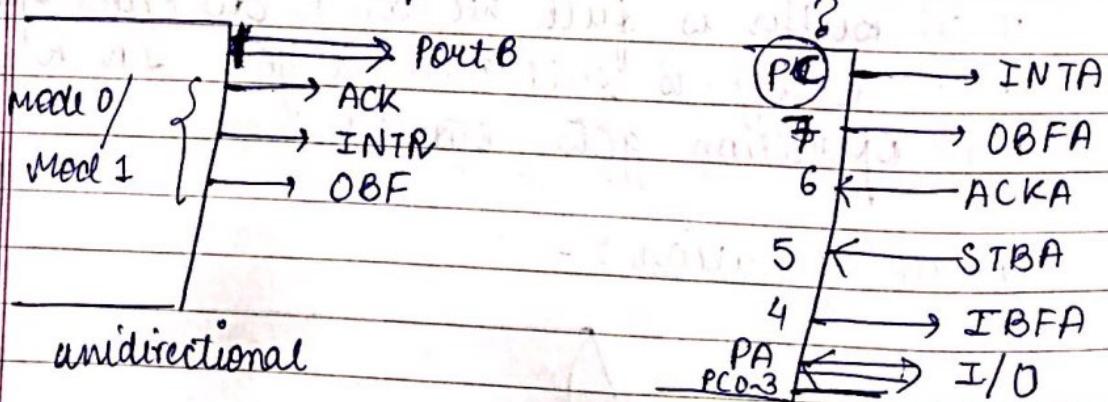
Until unless operation continues INTR remains high and as soon as data written part is completed and acknowledgement part is done then INTR Request / signal gets low.

Mode 2 Bidirectional Mode

same port is used in Read and Write mode we just keep some pins free and to make Port A bidirectional we can combine all the pins together.

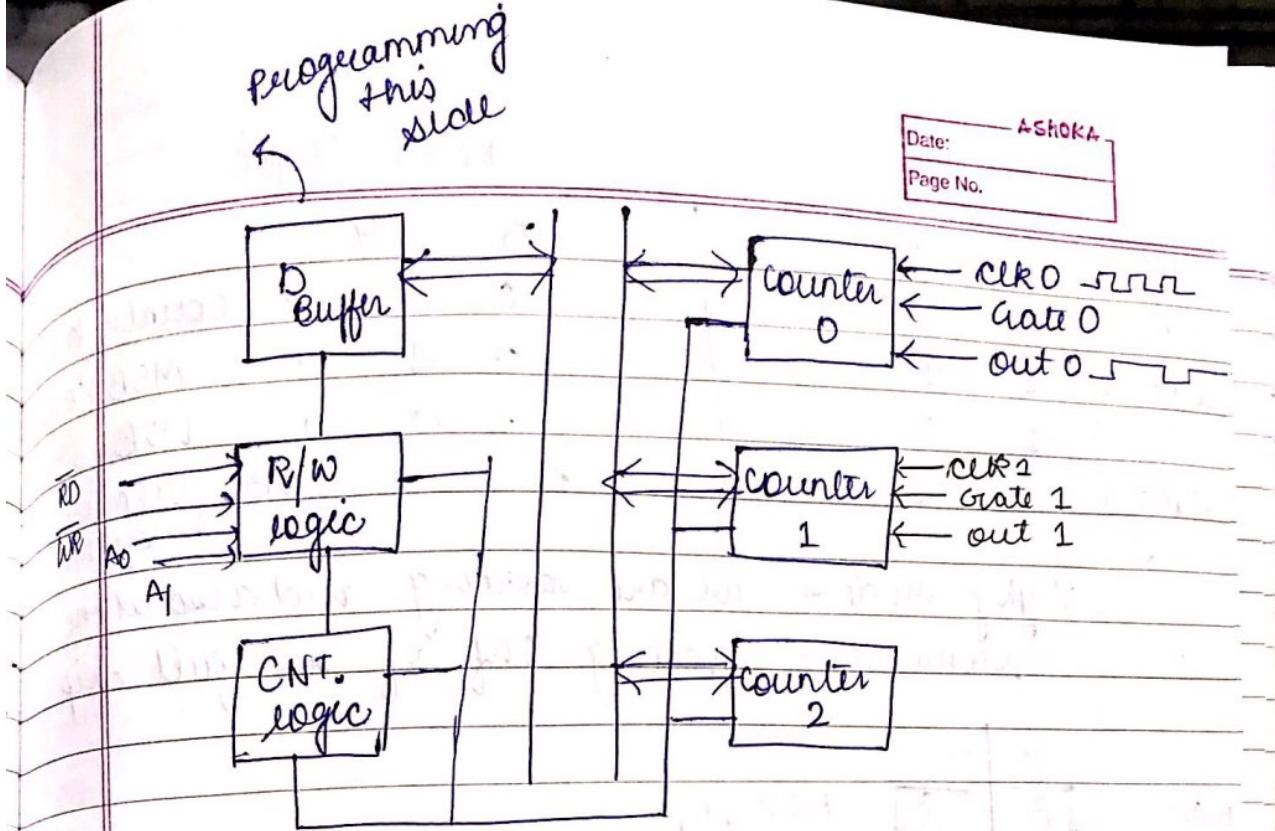


when strobe signal is low then only Read operation happens.



PIT (Programmable Interval Timer) :-

Timing Purpose,



Gate pin → Resetting of counter

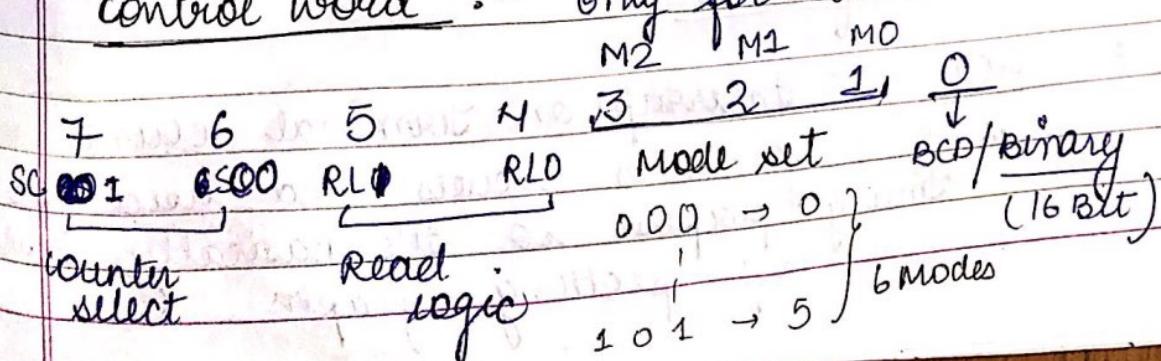
Input from clock

Instead of resetting i/p we reset counter.

CS	A1	AO	
0	0	0	counter 0
0	0	1	counter 1
0	1	0	counter 2
0	1	1	control word
1	X	X	

every IC have different control word but every programmable IC have a control word.

control word :- only for counter 0, 1, 2



Read logic

	7	6	:	5	4	
counter 0	0	0		0	0	→ counter latch
counter 1	0	1		0	1	→ MSB (8 bit)
counter 2	1	0		1	0	→ LSB ("")
Illegal	1	1		1	1	→ LSB followed by MSB

Flyby mode → we are working and also at the same time getting info of ~~not~~ full chip

Read logic $\frac{80}{10} \frac{5}{01}$ read logic

counter latch tells us about the status of counter. which counter are free and which are currently occupied.

23/11/18
#

M2 M1 M0

0 0 0 → Mode 0

0 1 0 → Mode 1

1 0 0 → Mode 2

1 0 1 → Mode 3

1 1 0 → Mode 4

1 1 1 → Mode 5

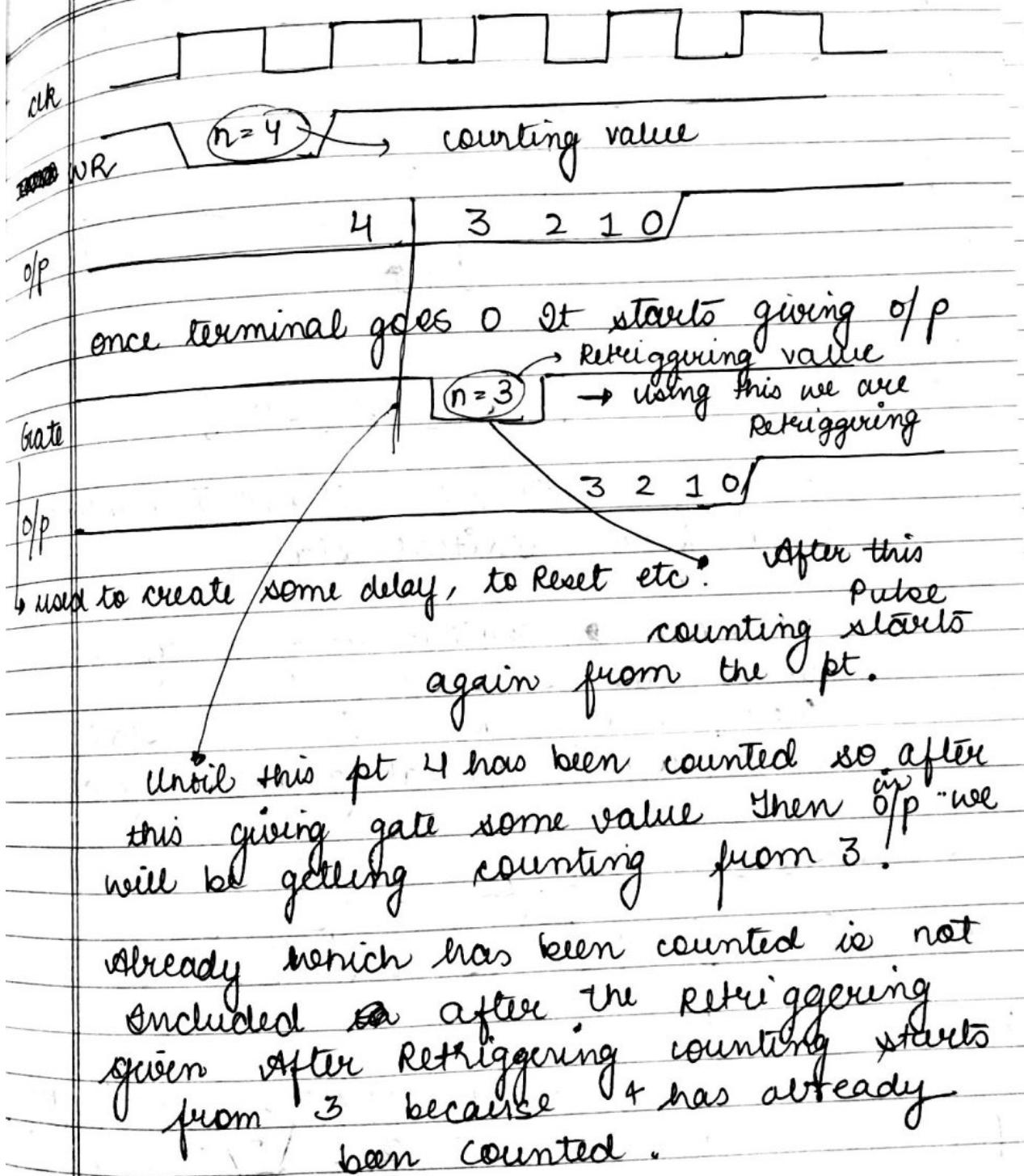
} only 6 modes

either Packed BCD or 16 bits Binary

8253 and 8254 are almost similar.

Modes of 8253 →

- 1) Mode 0 :- interrupt on terminal count
as PIT generates waves and used for timing purpose so it's basically used by plotting graphs.



a) # Mode 1 : One shot

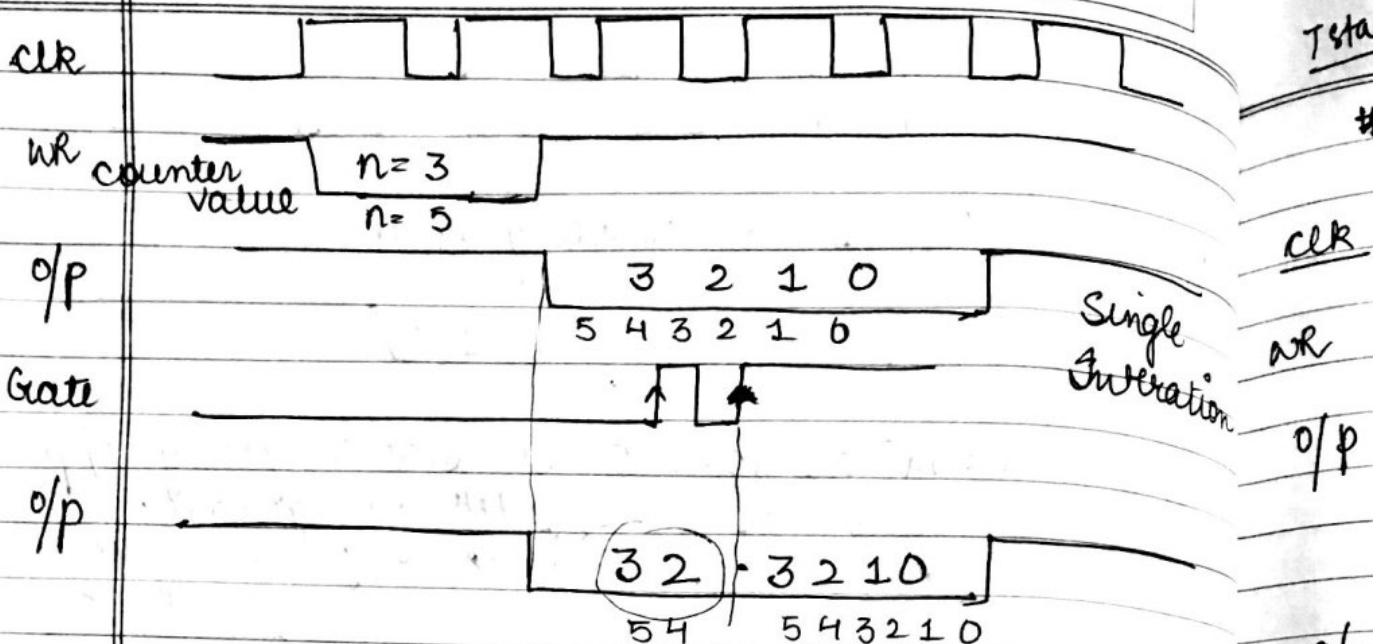
Hardware retriggen one shot

Gate used for Retriggering

Date: ASHOKA
Page No.

n²
T_{sta}

CLK



Once S/S goes Retriggered it starts counting again from initial state.

1T state \rightarrow whatever the frequency is it always takes 1T state.

One operation happens in 17 steps

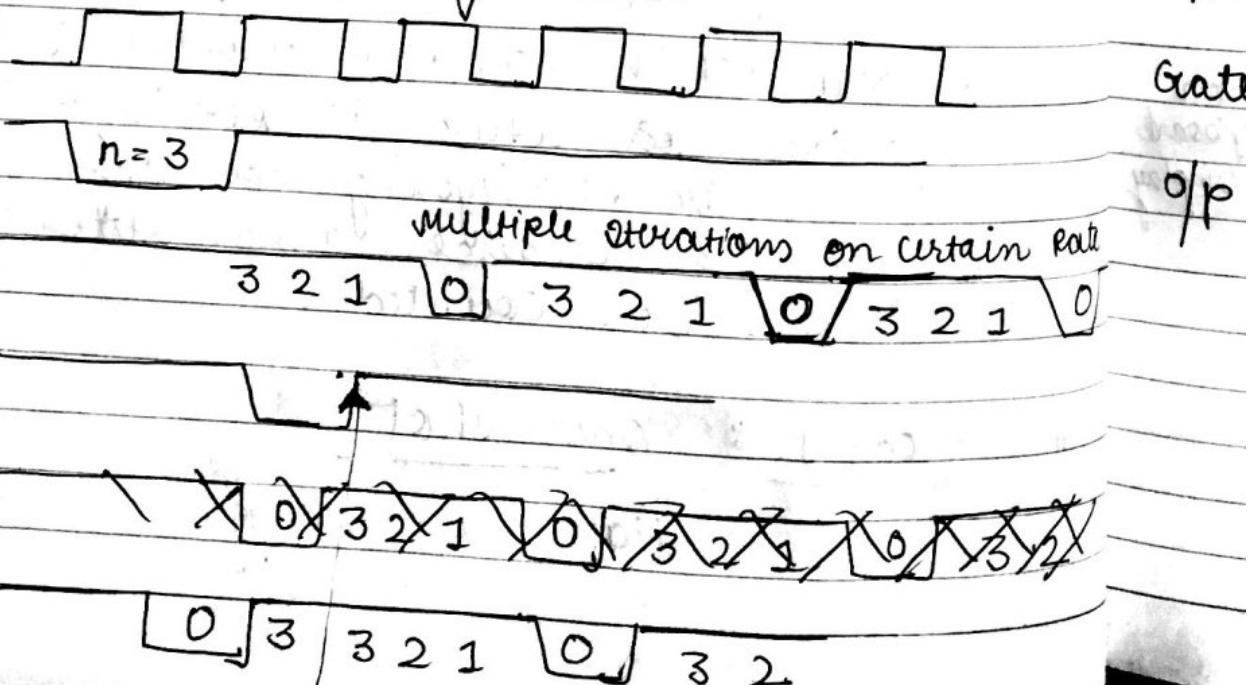
shots are used in generating PWM signal.

*

Mode 2 : Rate Generator :-

O/P

CLR



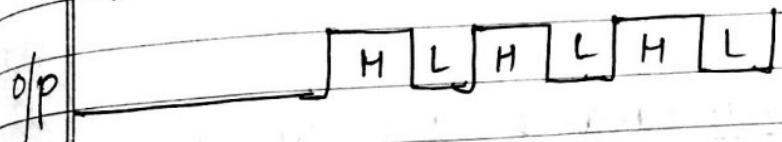
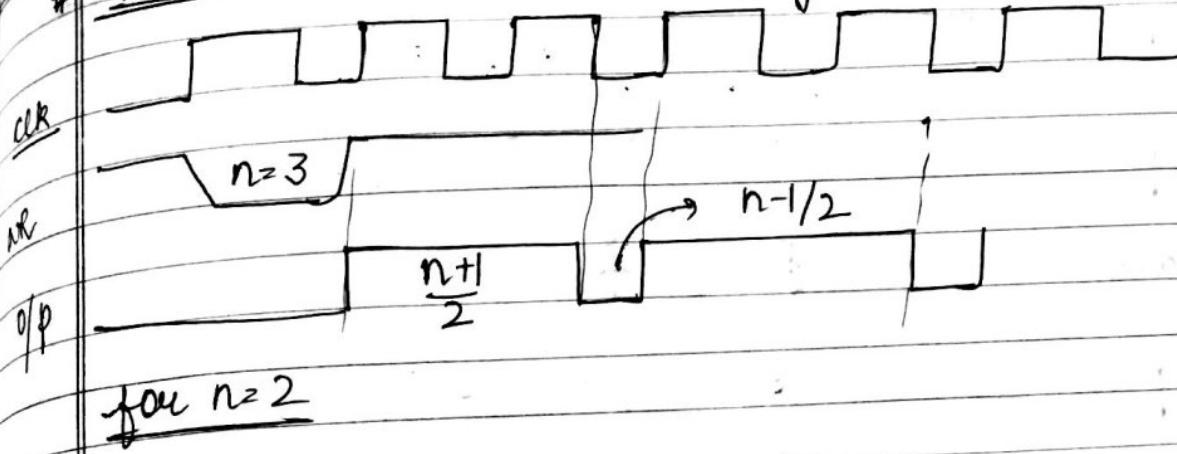
ASHOKA

Date:	
Page No.	

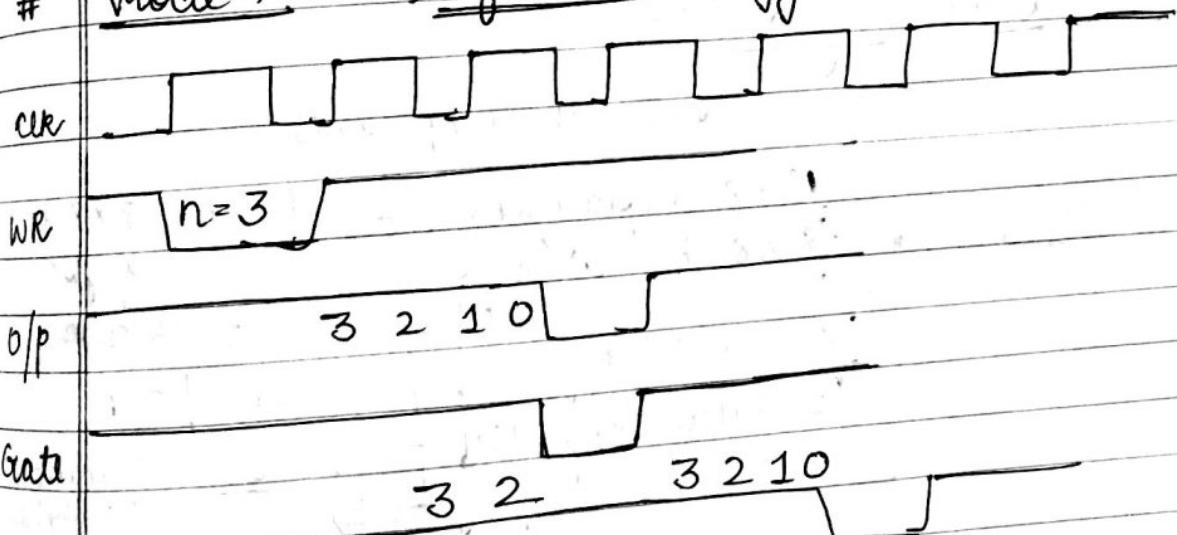
O/p n = odd
 $\left(\frac{n+1}{2}\right) \rightarrow \text{High}$
 $\left(\frac{n-1}{2}\right) \rightarrow \text{Low}$

n = even
 $n/2$ High $n/2$ Low

Mode 3 square wave generator :-

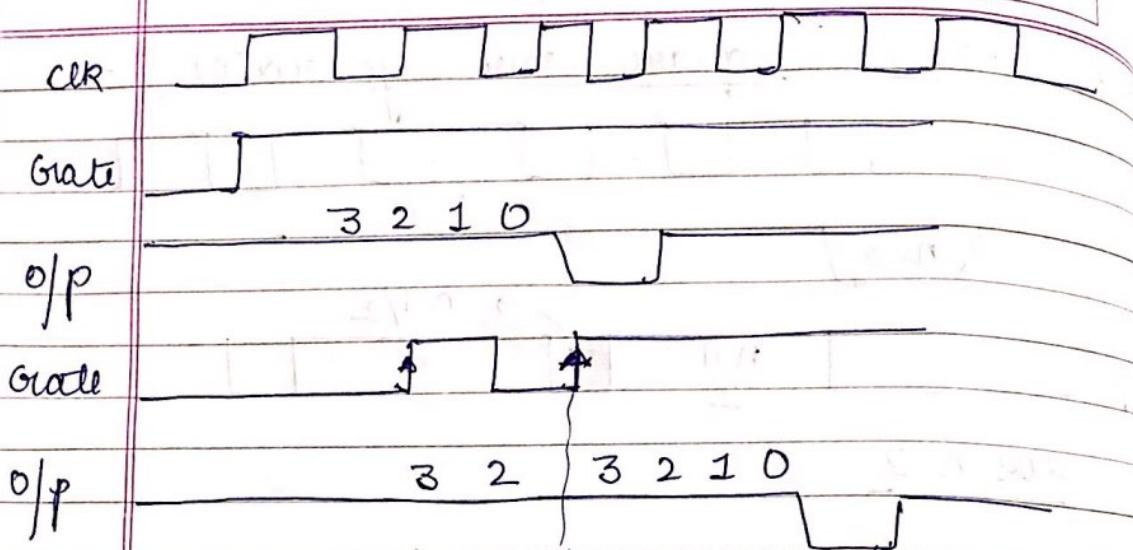


Mode 4 → software trigger strobe strobe



Mode 5 → hardware trigger strobe strobe

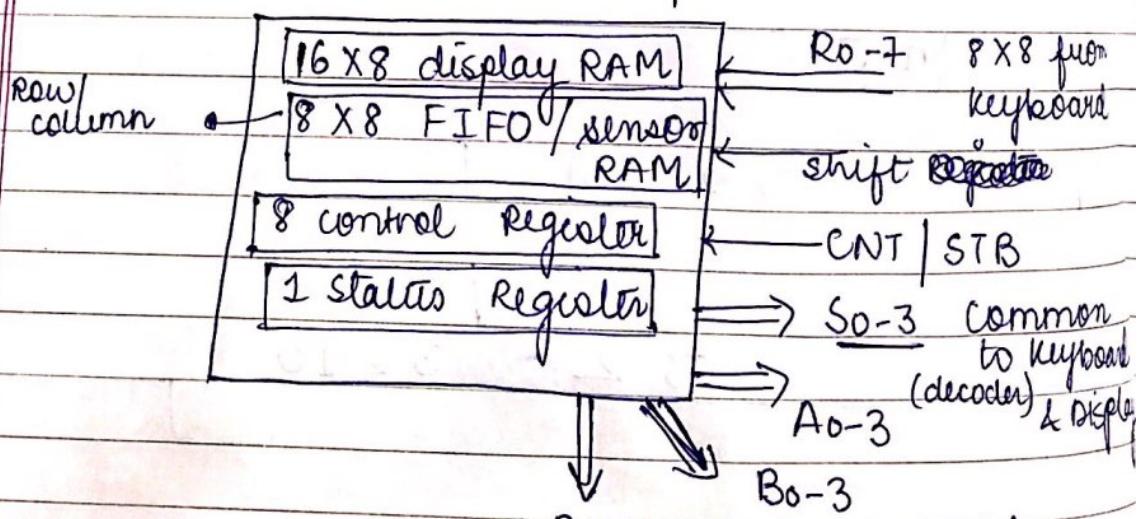
Here we does not use WR signal
 It works only when gate signal gets high then counter operation starts



K.D.I (Keyboard & display Interface)

Block Diagram :-

Internal Components



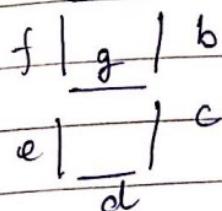
BD (Block display)

Shift → Right / left shift

S0-3 (select lines) → To make certain combinations for keyboard & display.

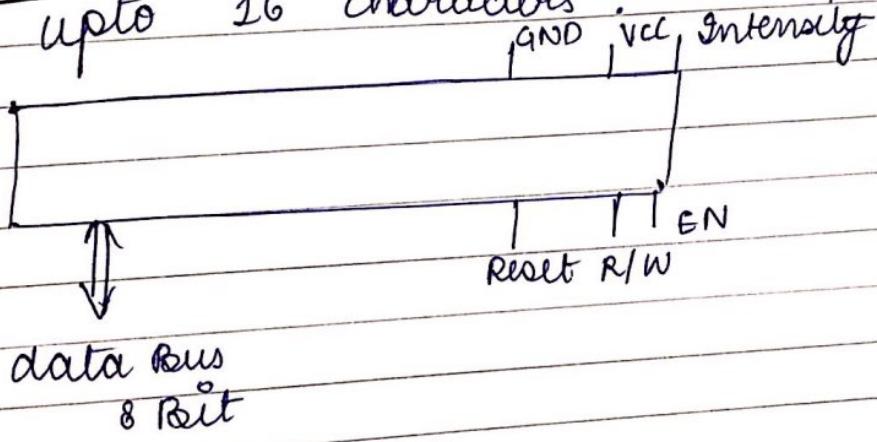
With the IC we can connect keyboard & display

for display we have
led
LCD (16x2)
7 segment display.
7 segment display :-
a



some nos in 8 bit
to get the no. what
we want.

LCD (16x2) → 2 rows and can print
upto 16 characters



DMA :- 8237 :- channels here

