Unit-7

Registers:-

As the flipflop is the basic memory unit. It is used for storing 1 bit memory information. The register consists of the set of flip flops which is used for storing a binary word. To store n bit binary word a set of n flipflops is used. There are different types of registers available in MSI circuits. The mostly used D flipflops are used as the registers. and such registers are called shift registers also. There are the different modes of operation for the registers.

* Series or serial operation
* Parallel operation

In case of serial operation, digits are put in sequence, one digit for each clock pulse. where as in case of parallel operation all digits get shifted simultaneously during a single clock pulse.

The block diagram for 4 bit register is as shown below

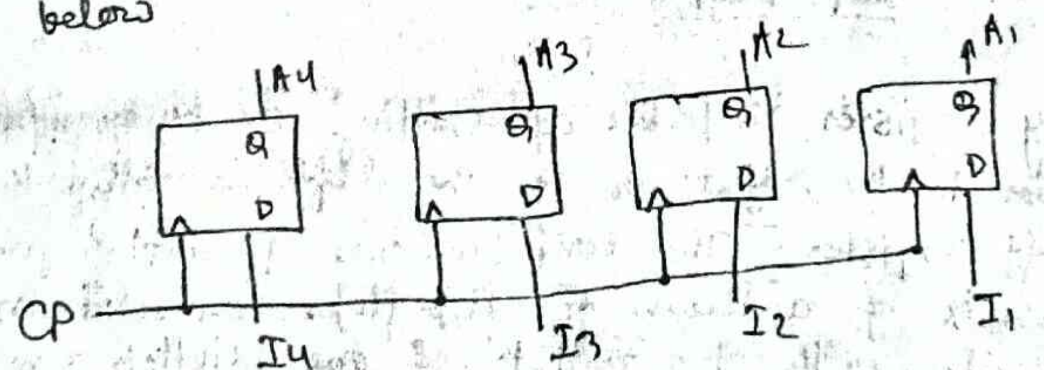


fig:- 4-bit register.

This is the register constructed by the 4-D flipflops and a common clock pulse input that enables all the flipflops so that the information presently available at the four inputs can be transferred into 4 bit register. There are different types of registers -

Serial in serial out (SISO)
serial in Parallel out (SIPO)
parallel In Serial out (PISO)
Parallel in Parallel out (PIPO)

In SISO data can be stored serially one bit at a time and output also transferred 1 bit at a time. In SIPO the data is stored serially but output is transferred in parallely.
Similarly in case of PISO the data inputs are provided parallel form & output is transferred in serial sequence where as in the PIPO form of register the data are taken in the parallel also the output is also transferred in parallel nature.

Shift Register:-

Any register capable of shifting its binary information either to the right or to the left is called the shift register. The configuration of shift register consists of a chain of flip-flops connected in cascade, with the output of one flipflop connected to the input of next flipflop. Also all the flipflops in the series receive a common clock pulse which cause the shift from one stage to next. The diagram is given as in the next page -

operations be performed by the shift register are the capabilities of these and are given as:

- A clear control to clear register to 0
- A Clock Pulse input for CP to synchronize all operations
- Shift right control to enable the shift right operation & the serial input & output associated to shift right
- Shift left control to enable the shift left operation & the serial input & output associated to shift left
- Parallel load control to enable a parallel transfer and the n input associated with the parallel transfer.
- n parallel output lines.
- A control state that leaves the information in the register unchanged even though clock pulses are continously applied.

Any shift register capable of both right & left shifting of information is called the bidirectional shift register and that shift only one direction is called unidirectional shift register. Also if the register has both shift and parallel load capabilities it is called a shift register with parallel load.

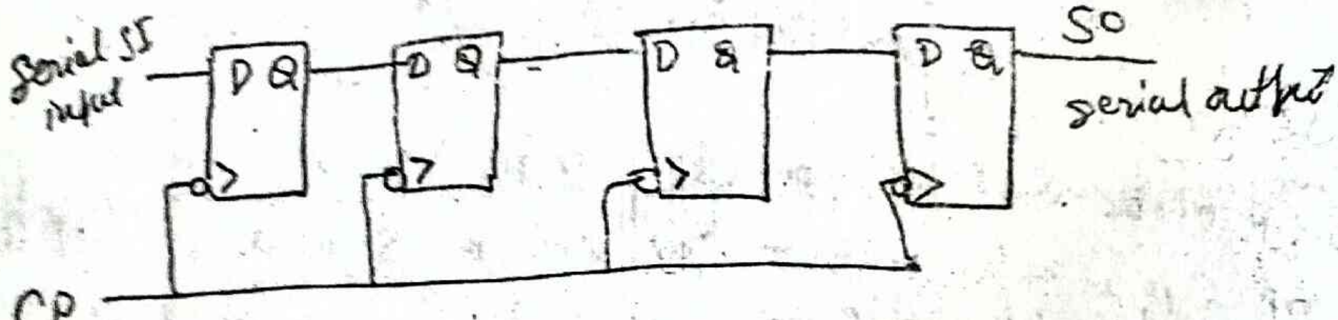


fig:- shift register

This register shifts its contents with every clock pulse during negative edge of pulse transition. This is indicated by small circle associated with the clock pulse input in all flip flops.

Counters:-

Sequential circuits that goes through a prescribed sequence of states upon the application of input pulses is called counter. OR it is a sequential circuit that counts the no. of input pulses that it receives. Generally it is constructed by using T or JK flipflop together with other combinational circuits. If we use n flipflops for counter then it will count 2^n states and after counting completed then the counter resets itself to the original states. There are two types of counters

* Synchronous counter
* Asynchronous counter / Ripple counter..

The synchronous counters are those in which the clock input is connected to all the flip flops individually so that they are simultenously clocked as shown in the figure

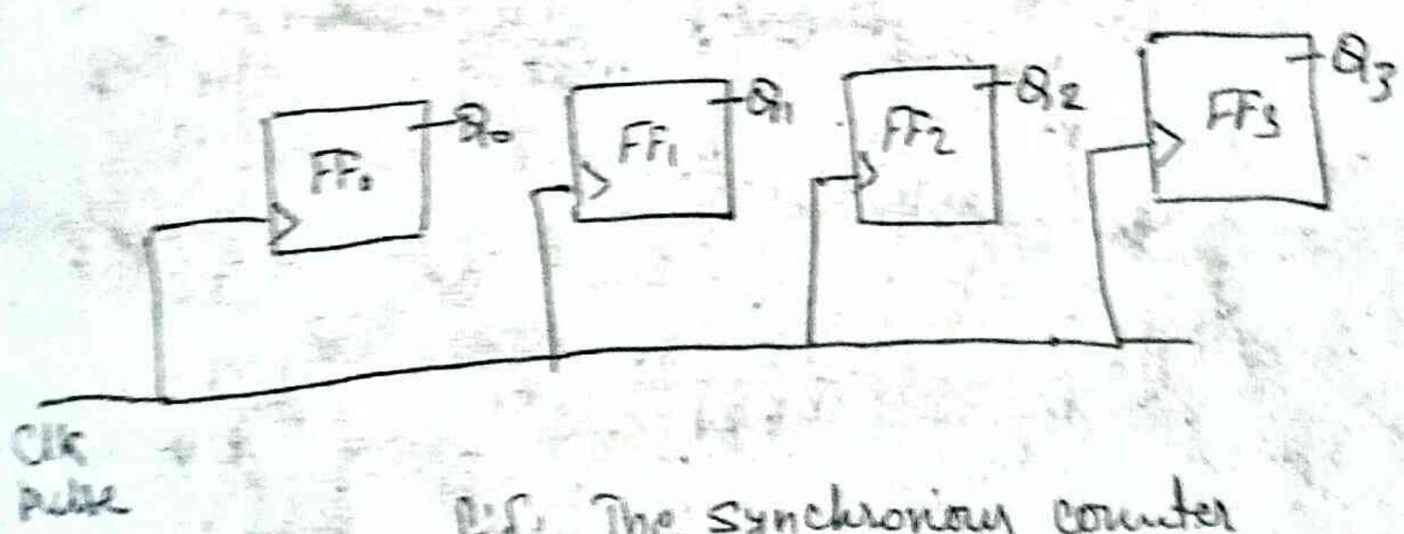


fig:- The synchronous counter

Asynchronous counters are also known as the ripple counters and are those in which the clock pulse is given as input to the first flipflop, the output of the first flipflop is fed as clock input to the second flip flop as shown in the figure given below.

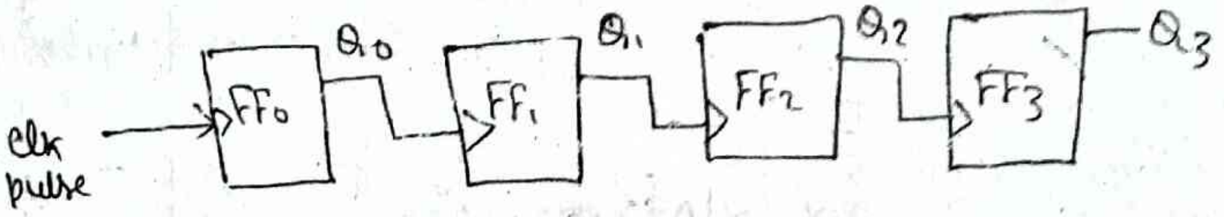
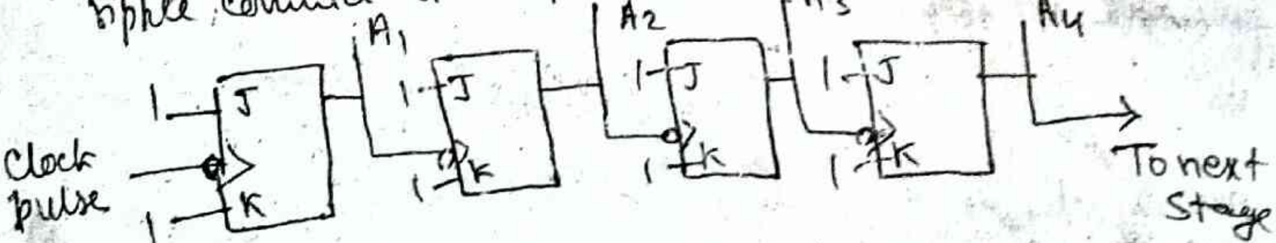


figure :- Ripple counter / Asynchronous counter.

<u>Binary ripple counter</u>:- Binary ripple counter consists of series of connection of complementing flip flops either (T flip flop or JK flip flop) with the output of each flipflop connected to the clock pulse input of the next higher order flipflop. The flip flop holding the least significant bit receive the incoming count pulses. The diagram of binary ripple counter is as follows-



<u>fig:- Binary ripple counter</u>

Here all the J-K inputs are equals to 1.

Uses of counter:-

* Counters are used for counting the number of times that certain event takes place.
* To control the fixed sequence of actions in the digital system
* To generate the clocks of different frequencies.
* To generate the timing signals.

4-Bit Ripple Counter:-

As we know that any n-bit ripple counter can count upto a maximum of 2^n states. If we connect four flip flops such that the output of first flipflop is fed to the clock input of the next as shown below, we get the 4-bit ripple counter.

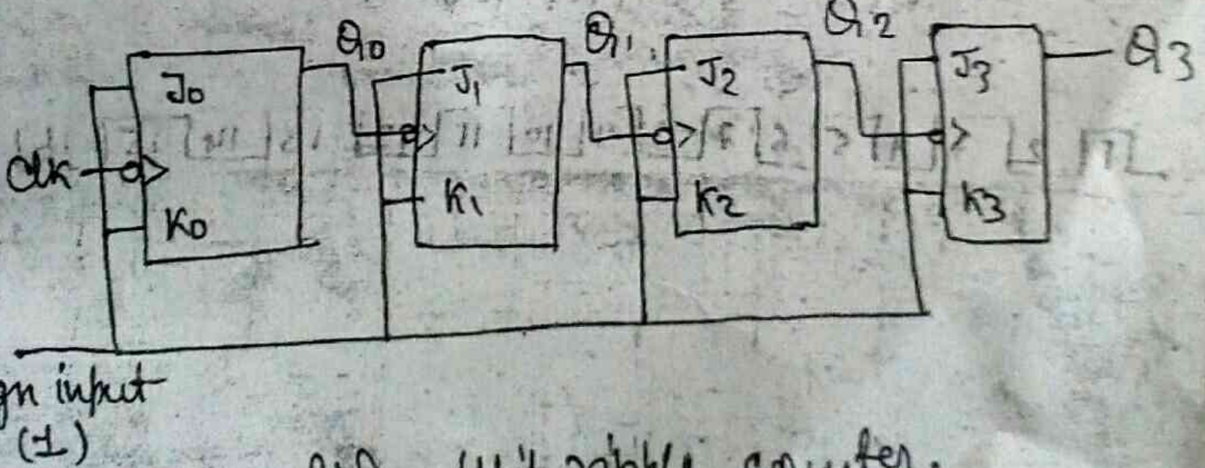


fig:- 4bit ripple counter.

This is the four bit ripple counter and it has $2^4 = 16$ states from 0000 to 1111 and then the counter reset to 0000 as shown in the truth table given before

Clock pulse	Q_3	Q_2	Q_1	Q_0	Count
initial	0	0	0	0	0
First	0	0	0	1	1
Second	0	0	1	0	2
third	0	0	1	1	3
fourth	0	1	0	0	4
fifth	0	1	0	1	5
sixth	0	1	1	0	6
seventh	0	1	1	1	7
eighth	1	0	0	0	8
ninth	1	0	0	1	9
tenth	1	0	1	0	10
eleventh	1	0	1	1	11
twelveth	1	1	0	0	12
13th	1	1	0	1	13
14th	1	1	1	0	14
15th	1	1	1	1	15
Sixteenth	0	0	0	0	Reset

The timing diagram for one cycle for the 4-bit ripple counter is shown as follow

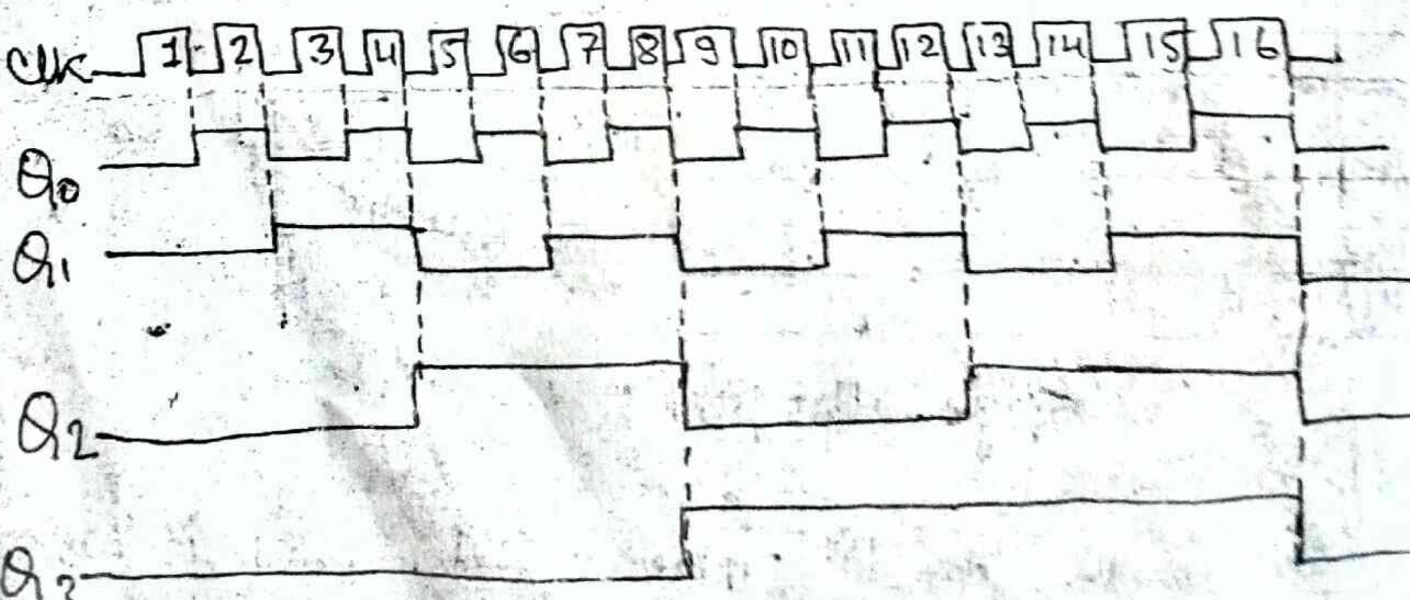


fig:- Timing diagram for 4-bit ripple counter

The memory unit

The part of the computer system that is capable of data storage either temporily or permenently is called the memory unit. Generally the registers present in the computers have the capability of storing data. The registers are either operational or storage. The operational registers are capable of storing the binary information in ~~its~~ their flip flop and also these are capable of the data processing task. Means the operational registers are capable of doing both string as well as the operational functions. Simillarly the storage registers are only used for storing the binary information in the temporary form. The memory unit of the computer system is the collection of storage registers.

Also the binary cells of registers in memory unit must have the following proporties.

-) It must have two reliable state for binary representation

-) It must have small size

-) The cost required for per bit storage should be as low as possible.

-) The time of access to memory register should be reasonably fast.

words is the collection of two, or four bytes. Simply it is the collection of binary information in terms of groups of bits. It is stored within the memory registers. It may either store some operands, instructions, or a group of alphanumeric characters or may be the binary coded informations

The memory unit is composed of different parts the memory address register (MAR) and memory buffer register (MBR).

The MAR specifies the memory word selected. Each word in memory is assigned a number identification starting from 0 up to the maximum number of words available. & The communication betn the specific memory word with the individual word the address of the word is transferred to the address register. An address register with n bits can specify upto 2^n memory words. Computer memory unit can range from 1024 words requiring an address register of 10 bits to $1048576 = 2^{20}$ words requiring a 20 bit address register.

State tables and diagrams

The effect of the previous inputs on the output is represented by the state of the circuit of the sequential circuit. The output of the sequential circuit at any time depends upon its current state and the input. The next state of the sequential circuit is also determined by the previous inputs and the current state of the circuit. The relationship that exists among the inputs, outputs, present states and the next states can be specified by state tables or the state diagrams.

The state table for the sequential circuit consists of the sections labelled present state, next state and output. The present state designates the state of flip flops before the occurance of a clock pulse. The next state shows the states of flipflops after the clock pulse, and the output section lists the value of the output variables during the present state. The flip flops can also be represented by the graphical symbol having the different states rather than the only graphical symbol, tables and the equations. The states of the flipflop are represented by the circles and the arrow headed arcs that indicates the transition between the different states. Consider the following diagram

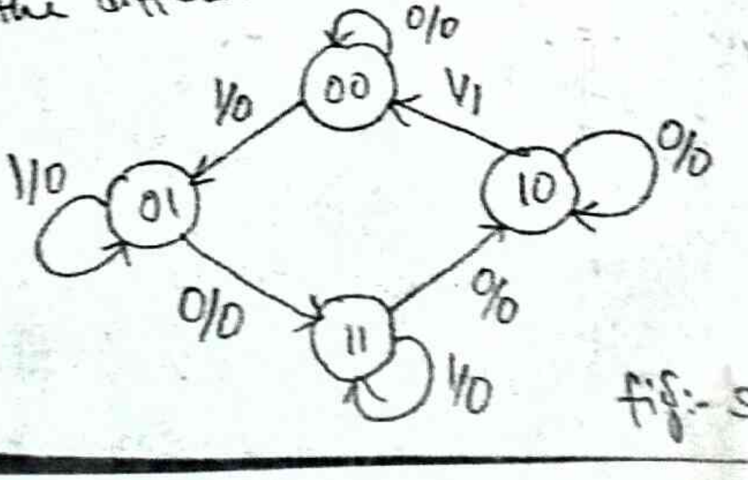
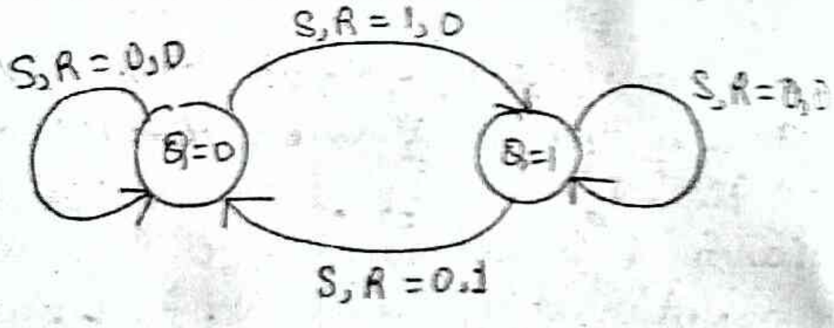


fig:- State diagram

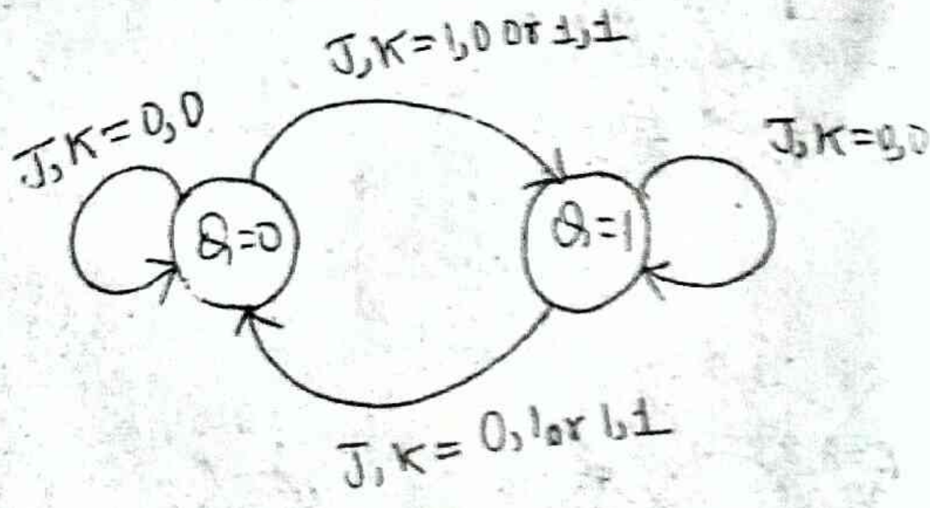
Here the binary number inside each circle indicates the state the circle represents. The directed lines are labelled with two binary numbers separated by a sl... The upper value indicates the input value and the lo... value indicate the output. Here the directed line from 00 state to 01 is labelled by 1/0 means if the sequent... circuit is in present state and the input is 1 then the next state is 01 and the output is 0. If it in a present state 00 and the input is 0 it will remain in that state. A directed line connecting a circle with itself indicates that no change of the state occurs. The state diagram provides exactly same information as the state table and is obtained directly from the state table.

State diagrams for different flipflops

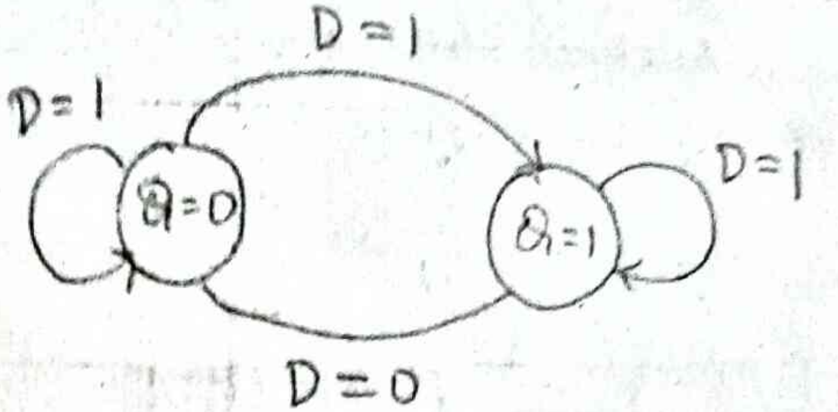
SR flip flop :-



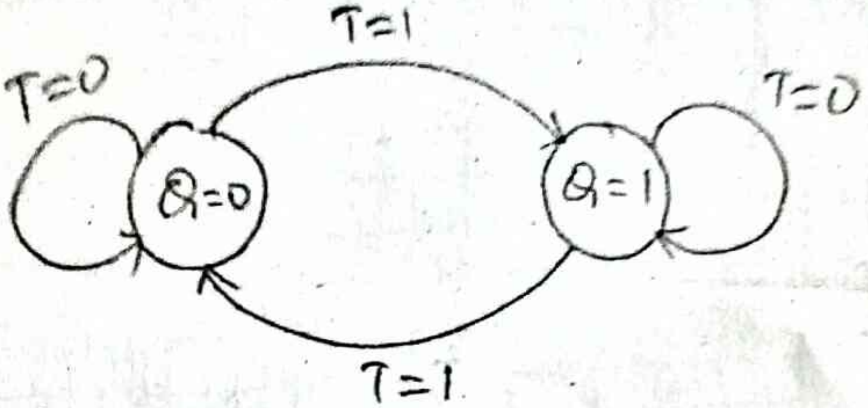
JK- flip flop :-



D flip flop:-



T-flipflop:-



we see from the table of the differend flipflops that all the flip flops have same number of the states and transitions. Each flip flop is in the set state when $Q = 1$ and in the reset state when $Q = 0$. Also each flipflop can move from one state to another or it can reenter the same state. The only difference between four tybes lies in the value of input signals that cause these transitions. The state diagram is considered very convenient way to visualise the operations of the flipflop or even of the large sequential components.

Design the circuit diagram that convert 4 bit binary number into its 2's complement

Soln Here the functional table is given by

inputs				outputs			
a	b	c	d	w	x	y	z
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

Now, calculating the value of w, x, y, z on the basis of minterms

w

	$c'd'$	$c'd$	cd	cd'
$a'b'$		1	1	1
$a'b$	1	1	1	1
ab				
ab'				

$$w = a'd + a'c + a'b + ab'c'd'$$

$$= a'(b+c+d) + ab'c'd'$$

$$= a \oplus (b+c+d)$$

Scanned by CamScanner

x

	$c'd'$	$c'd$	cd	cd'
$a'b'$		1	1	1
$a'b$	1			
ab	1			
ab'		1	1	1

$$x = b'c + b'd + bc'd'$$
$$= b'(c+d) + bc'd'$$
$$= b \oplus (c+d)$$

y

	$c'd'$	$c'd$	cd	cd'
$a'b'$		1		1
$a'b$		1		1
ab		1		1
ab'		1		1

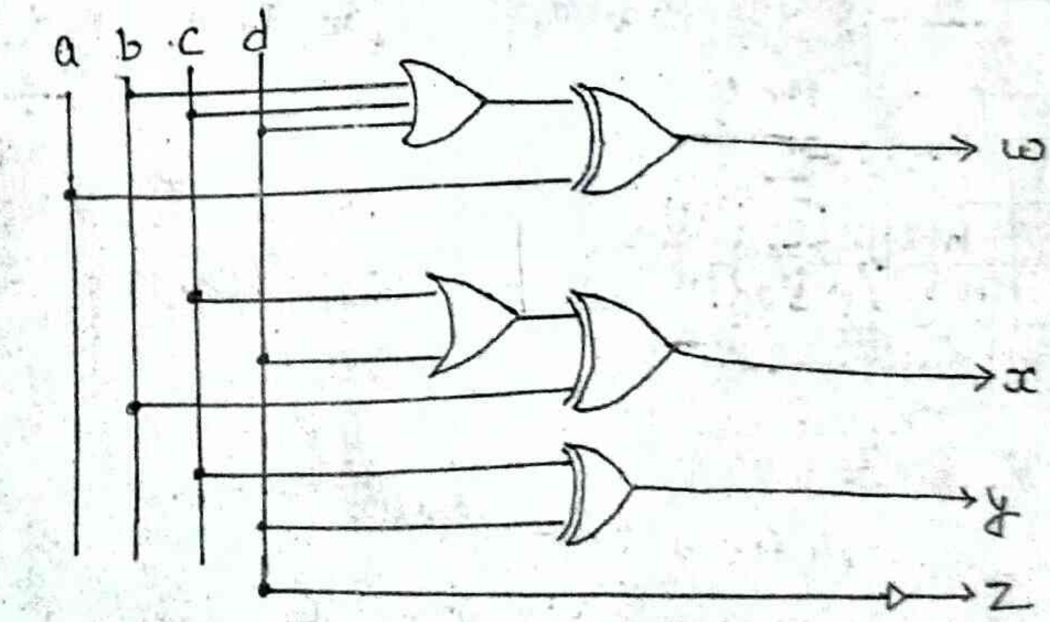
Z

	$c'd'$	$c'd$	cd	cd'
$a'b'$		1	1	
$a'b$		1	1	
ab		1	1	
ab'		1	1	

$$y = c'd + cd'$$
$$= c \oplus d$$

$$Z = d$$

Hence we can draw the circuit as follow -



Design a combinational circuit that accepts a three bit number and generates an output binary number equals to the square of input number.

<u>Soln</u> Here the truth table of the operation is

inputs			outputs					
a	b	c	u	v	w	x	y	z
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	1	0	0	0	0	1	0	0
0	1	1	0	0	1	0	0	1
1	0	0	0	1	0	0	0	0
1	0	1	0	1	1	0	0	1
1	1	0	1	0	0	1	0	0
1	1	1	1	1	0	0	0	1

Now calculating the value of u v w x y & z on the basis of minterms we get

u	b'c'	b'c	bc	bc'
a'				
a			1	1

$u = ab$

v	b'c'	b'c	bc	bc'
a'				
a	1	1	1	

$v = ab' + ac$

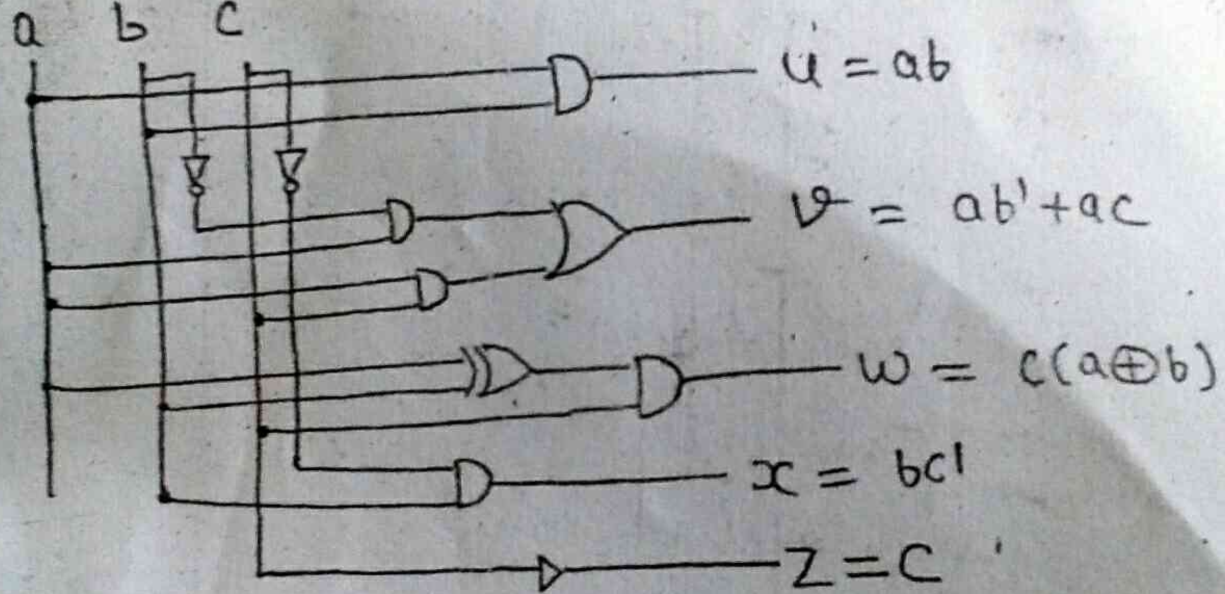
w	b'c'	b'c	bc	bc'
a'			1	
a		1		

$w = a'bc + ab'c$
$= c(a \oplus b)$

x	b'c'	b'c	bc	bc'
a'				1
a				1

$x = bc'$

Here the output of y all are equal to zero and output of z are as same as the inputs of c. Hence we can write $y = 0$ and $z = c$ and we draw the circuit as follows —


a
b
c
u = ab
v = ab' + ac
w = c(a⊕b)
x = bc'
z = c