DIGITAL ELECTRONICS

B TECH ELECTRICAL & ELECTRONICS ENGINEERING

PROFESSIONAL ENGINEERING CREDITS ASSESSMENT & CONTINUIOUS INTERNAL EVALUATION

PART I

UNIVERSITY EXAM QUESTION PATTERN – 6 MARK (EACH QUESTION

CARRIES 2 MARKS) ANSWER ANY 10 QUESTION

1. Define the term prime implicants and Essential prime implicants.

In Boolean logic, the term implicant has either a generic or a particular meaning. In the generic use, it refers to the hypothesis of an implication. In the particular use, a product term P is an implicant of a Boolean function F, denoted {\displaystyle P\leq F}, if P implies F.
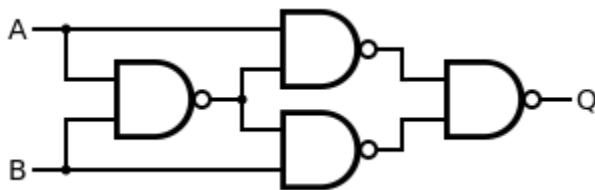
2. Draw the XOR logic using only NAND gates.

xor gate, now I need to construct this gate using only 4 nand gate

a b out0 0 00 1 11 0 11 1 0

the xor = (a and not b) or (not a and b), which is

$$\overline{A\quad B}+AB\quad \overline{A}\ \overline{B}+AB\overline{}$$

I know the answer but how to get the gate diagram from the formula?
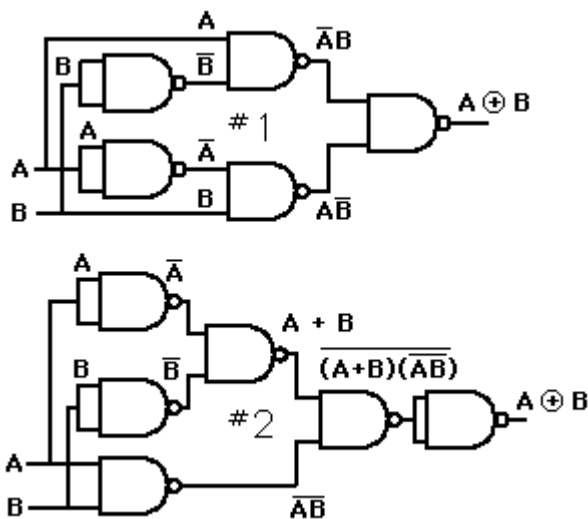

EDIT

I mean intuitively, to me, I should get this one if I do it step by step followed by the definition xor = (a and not b) or (not a and b).

$$\overline{A\ \overline{B}} \cdot \overline{AB} \qquad\qquad \overline{\overline{A\ \overline{B}} \cdot \overline{AB}} \qquad\qquad \overline{A}\ \overline{B} \cdot \overline{AB}$$

and xor will be constructed with 5 nand gates (first #1 image below)

my question is more like: imagine the first person in history figure out this formula, how can he or she (the thinking process) get the 4 nand soltuion from this formula, step by step.

$$\overline{A}\quad B + A\overline{B}$$

3. Express the switching function f (ABC) = B in terms of minterm.

Minterm Notation. ➤f = A'BC + AB'C' + AB'C + ABC' +ABC; The other way to represent f is: f (A,B,C) = m.

4. Define minterm & Maxterm. Give examples.

in which each appears exactly once in true or complemented form. e.g.: minterms of 3 variables: - Each minterm = 1 for only one combination of values of the variables, = 0 otherwise. Definition: a maxterm of n variables is a sum of the variables.

5. Simplify the given Boolean Expression F= x'+xy+xz'+xy'z

f=x'+xy+xz'+xy'z'f=x'+xy+xz'+xy'z'

We know a+a′b=(a+a′)(a+b)=a+ba+a′b=(a+a′)(a+b)=a+b

And a+a=aa+a=a

$\Longrightarrow$f=(x'+xy)+(x'+xz')+(x'+xy'z')f=(x'+xy)+(x'+xz')+(x'+xy'z')

f=(x'+y)+(x'+z')+(x'+y'z')f=(x'+y)+(x'+z')+(x'+y'z')

f=x'+y+z'+y'z'f=x'+y+z'+y'z'

f=x'+y+z'(1+y')f=x'+y+z'(1+y')

f=x'+y+z'[∵ 1+a=1]

6. Prove that the logical sum of all minterms of a Boolean function of 2 variables is 1

S = abc+ abc'+ ab'c+ ab'c'+ a'bc+ a'bc'+ a'b'c+ a'b'c' =
ab(c+c') + ab'(c+c')+ a'b(c+c')+ a'b'(c+c')=
ab+ab'+ a'b+a'b'=
a(b+b')+ a'(b+b')=
a+ a' = 1
We know that x+ x' =1

7. Show that a positive logic NAND gate is a negative logic NOR gate

2-22 Show that a positive logic NAND gate is a negative logic NOR gate and vice versa. ... A positive-logic NAND gate implements the function (xy)'. Hence a negative-logic NAND gate implements ((x'y')')' = (x+y)', which is NOR function.

8. If A & B are Boolean variables and if A=1 & A+B=0, Find B?

A=1

A+B=0

1+B=0

B=0-1

B=-1 Ans

9. What are universal gates implement AND gate using any one universal gate?

A universal gate is a gate which can implement any Boolean function without need to use any other gate type. The NAND and NOR gates are universal gates. In practice, this is advantageous since NAND and NOR gates are economical and easier to fabricate and are the basic gates used in all IC digital logic families.

10. What are the advantages of Schottky TTL family?

The advantage/disadvantage of Schottky TTL logic circuit over Standard TTL logic circuit is that:

- It provides low power consumption.
- It virtually eliminates saturation delay time.
- It provides simple circuitry.
- It gives low switching speed.

11. Define the term (i). Propagation delay (ii). Power dissipation

Propagation delay is defined as the flight time of packets over the transmission link and is limited by the speed of light. For example, if the source and destination are in the same building at the distance of 200 m, the propagation delay will be ~ 1 µsec.

The definition of power dissipation is the process by which an electronic or electrical device produces heat (energy loss or waste) as an undesirable derivative of its primary action.

12. Draw an active high tri-state Gate & write its truth table

Tri-state gates have additional circuitry via which the gate outputs can be enabled or disabled. This is very useful in digital systems where devices communicate via common wires called busses. Only one device can talk at a time; the others are disabled
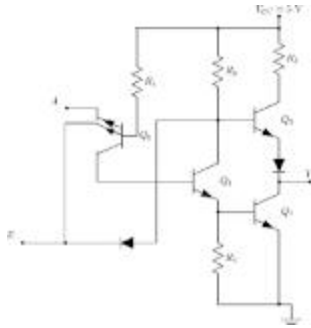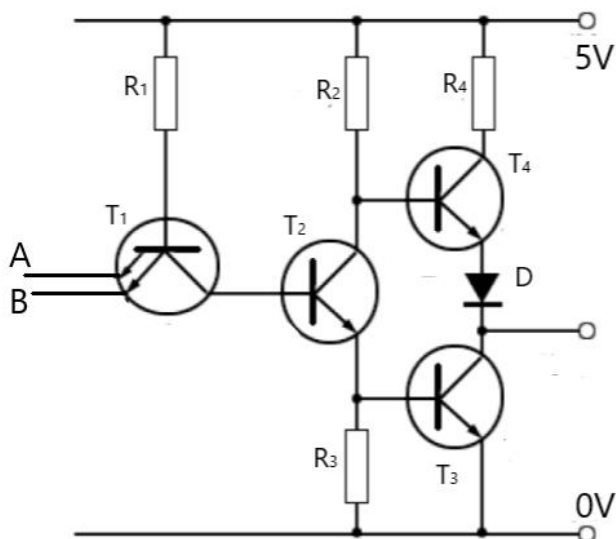
The NAND gate is a combination of an AND gate and NOT gate. They are connected in cascade form. It is also called Negated And gate. The NAND gate provides the false or low output only when their outputs is high or true.

CARRIES 6 MARKS) ANSWER ANY 5 QUESTION

2..Explain the operation of 3 input TTL NAND gate with required diagram & truth table.

The above diagram is the circuit diagram of a TTL NAND gate.

From the diagram, we shall explain the working. Now, as seen, the transistor

$$T_1$$

has two emitters to allow two inputs into the transistor. Now, as connected the base voltage will be at 5V. if both inputs are logic 1 (usually means about 5V too), the potential difference across base and emitter would be zero or nearly. Hence, no current will flow and the transistor is turned off. So, the collector voltage would also be equal to about 5V. Hence, this potential can drive current through the emitter of the transistor

$$T_2$$

. This then will allow the collector voltage of the transistor

$$T_2$$

to fall.
Now due to the current flowing through the emitter, there would be a voltage drop across the resistor

$$R_3$$

. The desired voltage drop would be about

$$0.7V$$

. As seen, this is the input of the transistor

$$T_3$$

. Hence, the transistor is turned on. Due to saturation, the collector voltage will fall to about

$$0.2V$$

which is a logic 0. A
For the transistor

$$T4$$

, observe that the emitter voltage is made up of the entire voltage of the transistor

$$T3$$

plus the voltage drop across the diode D about

$$0.7V$$

. Hence the emitter potential would be

$$0.7+0.2=0.9V$$

. Now the base voltage of the transistor

$$T4$$

, would be the voltage across the base-emitter of

$$T3$$

and the voltage of the entire transistor (i.e.) voltage across emitter-collector. This would also be equal to about

$$0.9V$$

. Hence the emitter voltage and the collector voltage are equal. So the transistor

$$T4$$

will be turned off too. So the output is zero when both inputs are 1.

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

4. Compare & contrast the features of TTL & CMOS logic families

Comparison between TTL and CMOS technology:

TTL	CMOS
TTL stands for Transistor-Transistor Logic. The name is derived from the use of two Bipolar Junction Transistors or BJTs in the design of each logic gate.	CMOS stands for Complementary Metal Oxide Semiconductor.
TTL is a classification of integrated circuits.	CMOS is another classification of ICs that uses field effect transistors in the design.
The density of logic gates is less in TTL as compared to CMOS.	The primary advantage of CMOS chips to TTL chips is in the greater density of logic gates within the same material.
A single gate on a TTL chip can consume around 10mW of power.	An equivalent single gate in a CMOS chip can consume around 10nW.
TTL chips are lesser delicate and is not very suseptible to electrostatic discharge.	CMOS chips are a bit more delicate compared to TTL chips when it comes to handling as it is quite susceptible to electrostatic discharge.
TTL chips do not have CMOS logic.	There are CMOS chips that have TTL logic and are meant as replacements for TTL chips.
A logic gate in a TTL chip can consist of a substantial number of	A single logic gate in a CMOS chip can consist of as little as two Field

|	
parts as extra components like resistors are needed.	Effect Transistors.
TTL circuits consumes more power compared to CMOS circuits at rest.	CMOS circuits comsumes less power at rest.

5. List out the basic rules (laws) that are used in Boolean algebra expressions with example.

Boolean Expression	Description	Boolean Algebra Law or Rule
A + A = 1	A in parallel with NOT A = "CLOSED"	Complement
A . A = 0	A in series with NOT A = "OPEN"	Complement
A+B = B+A	A in parallel with B = B in parallel with A	Commutative
A.B = B.A	A in series with B = B in series with A	Commutative

7. Draw the schematic and explain the operation of a CMOS inverter. Also explain its characteristics

CMOS (Complementary Metal Oxide Semiconductor)

The main advantage of CMOS over NMOS and BIPOLAR technology is the much smaller power dissipation. Unlike NMOS or BIPOLAR circuits, a Complementary MOS circuit has almost no static power dissipation. Power is only dissipated in case the circuit actually switches. This allows integrating more CMOS gates on an IC than in NMOS or bipolar technology, resulting in much better performance. Complementary Metal Oxide Semiconductor transistor consists of P-channel MOS (PMOS) and N-channel MOS (NMOS). Please refer to the link to know more about the fabrication process of CMOS transistor.
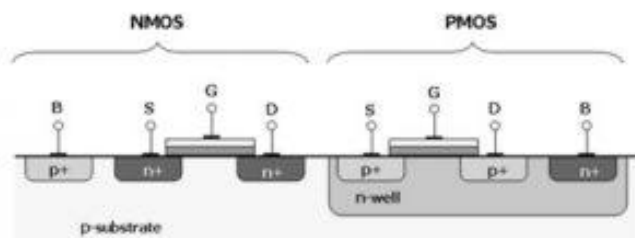
CMOS (Complementary Metal Oxide Semiconductor)

NMOS

NMOS is built on a p-type substrate with n-type source and drain diffused on it. In NMOS, the majority of carriers are electrons. When a high voltage is applied to the gate, the NMOS will conduct. Similarly, when a low voltage is applied to the gate, NMOS will not conduct. NMOS is considered to be faster than PMOS, since the carriers in NMOS, which are electrons, travel twice as fast as the holes.
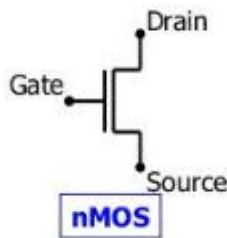
NMOS Transistor

PMOS

P- channel MOSFET consists of P-type Source and Drain diffused on an N-type substrate. The majority of carriers are holes. When a high voltage is applied to the gate, the PMOS will not conduct. When a low voltage is applied to the gate, the PMOS will conduct. The PMOS devices are more immune to noise than NMOS devices.
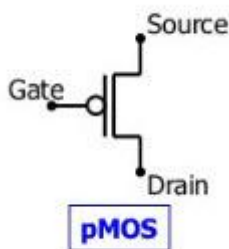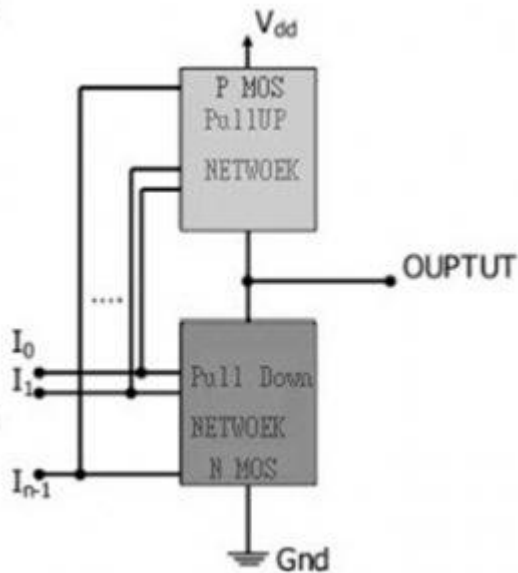
PMOS Transistor

CMOS Working Principle

In CMOS technology, both N-type and P-type transistors are used to design logic functions. The same signal which turns ON a transistor of one type is used to turn OFF a transistor of the other type. This characteristic allows the design of logic devices using only simple switches, without the need for a pull-up resistor.

In CMOS logic gates a collection of n-type MOSFETs is arranged in a pull-down network between the output and the low voltage power supply rail (Vss or quite often ground). Instead of the load resistor of NMOS logic gates, CMOS logic gates have a collection of p-type MOSFETs in a pull-up network between the output and the higher-voltage rail (often named Vdd).

CMOS using Pull Up & Pull Down

Thus, if both a p-type and n-type transistor have their gates connected to the same input, the p-type MOSFET will be ON when the n-type MOSFET is OFF, and vice-versa. The networks are arranged such that one is ON and the other OFF for any input pattern as shown in the figure below.

CMOS offers relatively high speed, low power dissipation, high noise margins in both states, and will operate over a wide range of source and input voltages (provided the source voltage is fixed). Furthermore, for a better understanding of the Complementary Metal Oxide Semiconductor working principle

CMOS Characteristics

The most important characteristics of CMOS are low static power utilization, huge noise immunity. When the single transistor from the pair of MOSFET transistor is switched OFF then the series combination uses significant power throughout switching among the two stated like ON & OFF.

As a result, these devices do not generate waste heat as compared with other types of logic circuits such as TTL or NMOS logic, which usually use some standing current even they don't change their state.

These CMOS characteristics will allow for integrating logic functions with high density on an integrated circuit. Because of this, CMOS has become the most frequently used technology to be executed within VLSI chips.

The phrase MOS is a reference to the MOSFET's physical structure which includes an electrode with a metal gate that is located on the top of an oxide insulator of semiconductor material.

A material like Aluminum is used only once however the material is now polysilicon. The designing of other metal gates can be done using a comeback through the arrival of high-κ dielectric materials within the process of the CMOS process.

6. Implement the expression Y (A, B, C) = ΠM (0, 2, 4, 5, 6,) using only NOR-NOR logic
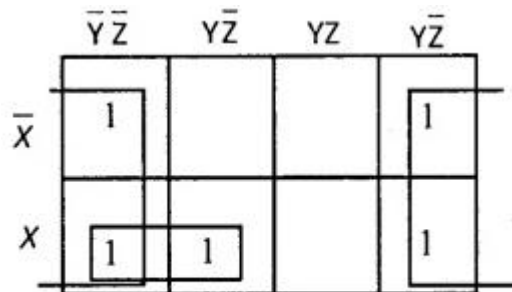
$F(x, y, z) = x\bar{z} + \bar{z}$

CARRIES 2 MARKS) ANSWER ANY 10 QUESTION

1. What is combinational circuit? Give examples.

A Combinational Circuit consist of logic gates whose outputs at any instant of time are determined directly from the present combination of inputs without regard to previous input.

There are the following characteristics of the combinational logic circuit:

- At any instant of time, the output of the combinational circuits depends only on the present input terminals.
- The combinational circuit doesn't have any backup or previous memory. The present state of the circuit is not affected by the previous state of the input.
- The n number of inputs and m number of outputs are possible in combinational logic circuits.

Examples of combinational circuits:

Adder,

Subtractor,

Converter,

Encoder/Decoder.

2. Draw the block diagram of n-bit parallel adder

Block diagram and Logic circuit diagram of a Parallel Binary adder can be given as,

Block diagram :-

Logic diagram :-
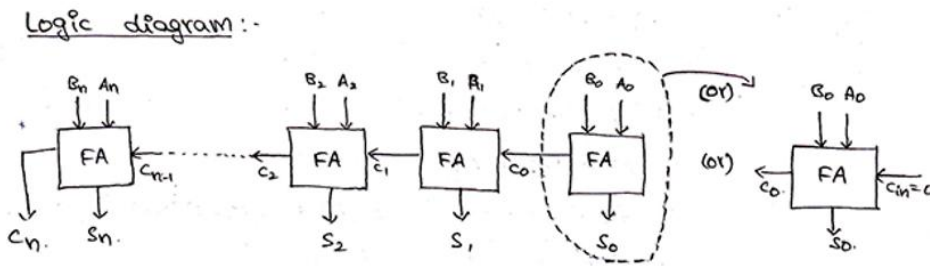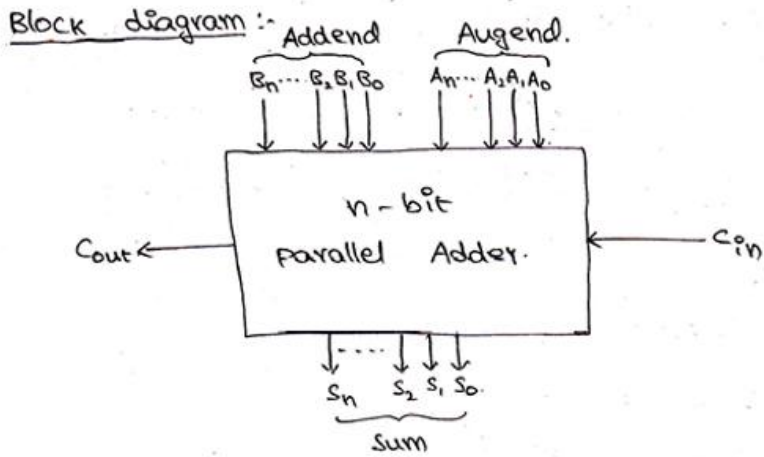
4-bit Binary Adder

3. Write an expression for borrow and difference in a full Subtractor circuit

Full-Subtractor Circuit

Construction of Full Subtractor Circuit:

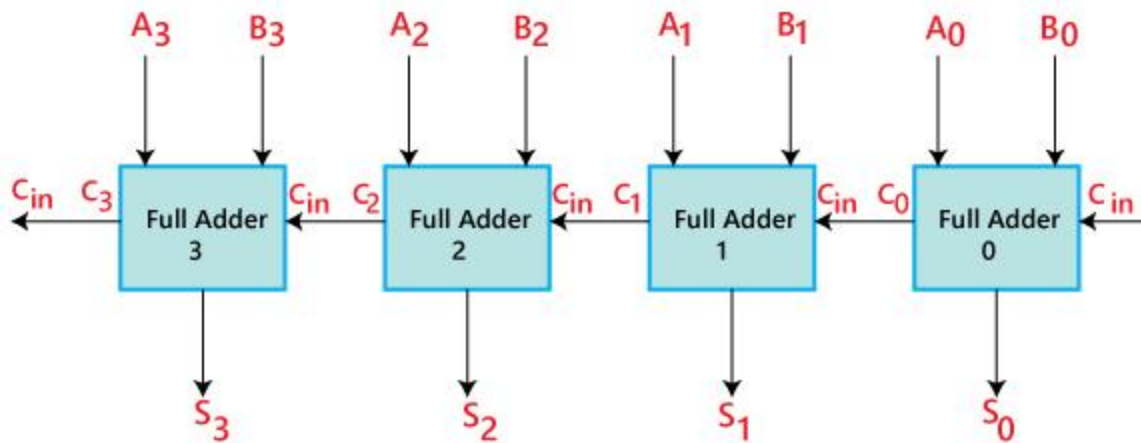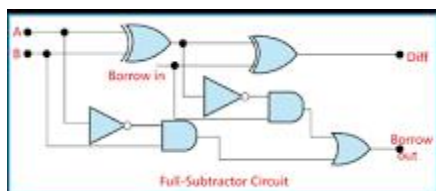The 'Diff' output of the first subtractor will be the first input of the second half subtractor, and the 'Borrow' output of the first subtractor will be the second input of the second half subtractor. The second half subtractor will again provide 'Diff' and 'Borrow'.

4. Differentiate a decoder from a Demultiplexer.

The main difference between demultiplexer and decoder is that a demultiplexer is a combinational circuit which accepts only one input and directs it into one of the several outputs. On the contrary, the decoder is a combinational circuit which can accept many inputs and generate the decoded output.

5..Design Half – adder using only NAND gates.

The minimum number of NAND gates required to design half adder is 5. The first NAND gate takes the inputs which are the two 1-bit numbers. The resultant NAND operated inputs will be again given as input to 3- NAND gates along with the original input.

6. What is code converter? List their types.

There are many other possible types of code converters known as BCD-to-seven-segment code converter, BCD-to-Gray code converter, BCD-to-excess-3 code converters, and so on.

7. Draw a logic diagram of 1 to 4 data distributor
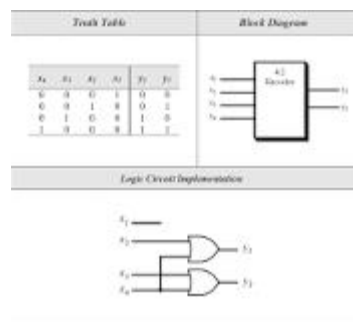
There are many other possible types of code converters known as BCD-to-seven-segment code converter, BCD-to-Gray code converter, BCD-to-excess-3 code converters, and so on.

10. Draw the block diagram of a 2's complement 4 – bit adder/ Subtractor.
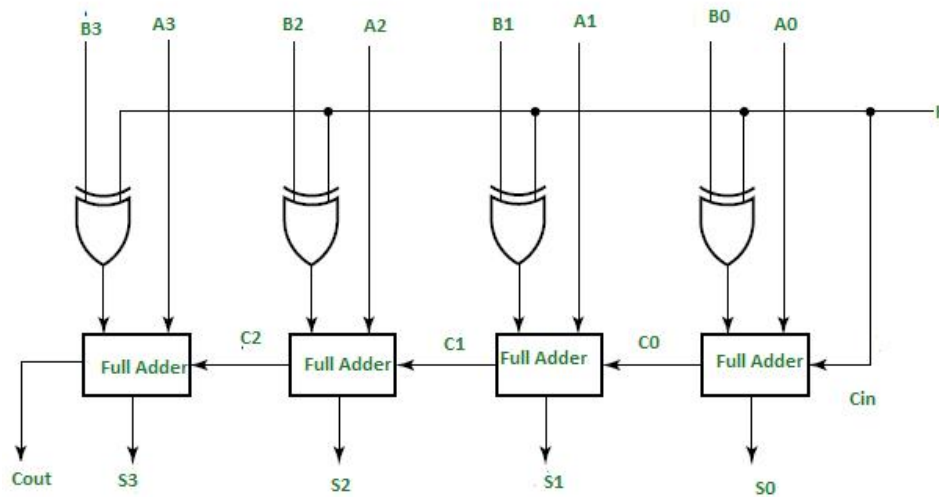
14. Design a single bit magnitude comparator to compare two words A and B

A magnitude digital comparator is a combinational circuit that compares two digital or binary numbers (consider A and B) and determines their relative magnitudes in order to find out whether one number is equal, less than or greater than the other digital number.
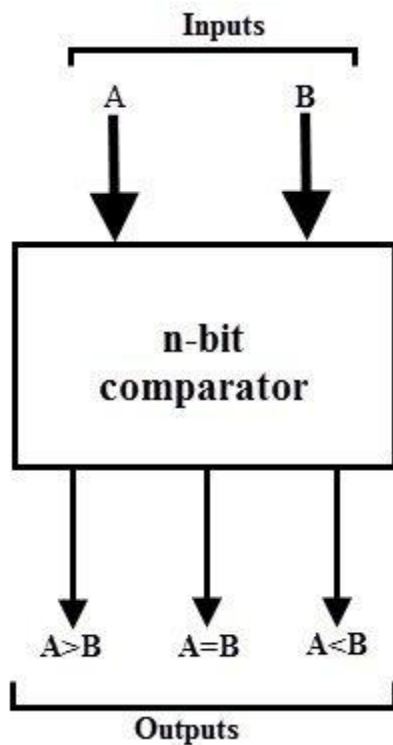
These are available in TTL as well as CMOS logic family ICs and some of these ICs include IC 7485 (4-bit comparator), IC 4585 (4-bit comparator in CMOS family) and IC 74AS885 (8-bit comparator).

Single Bit Magnitude Comparator

A_0	B_0	L	E	G
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

$$A0 < B0: L = \overline{A0}\, B0$$

$$A0 = B0: E = \overline{A0}\,\overline{B0} + A0\, B0$$

$$A0 > B0: G = A0\, \overline{B0}$$

It is to be noted that E can be realized as $\overline{(L + G)}$.
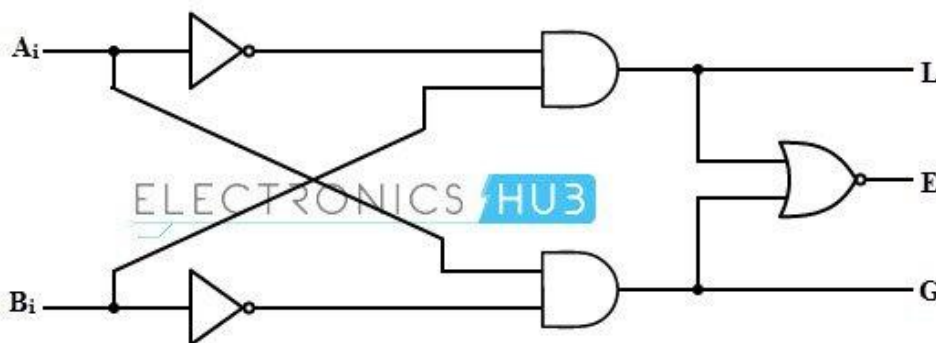
2-Bit Comparator

A 2-bit comparator compares two binary numbers, each of two bits and produces their relation such as one number is equal or greater than or less than the other. The figure below shows the block diagram of a two-bit comparator which has four inputs and three outputs.
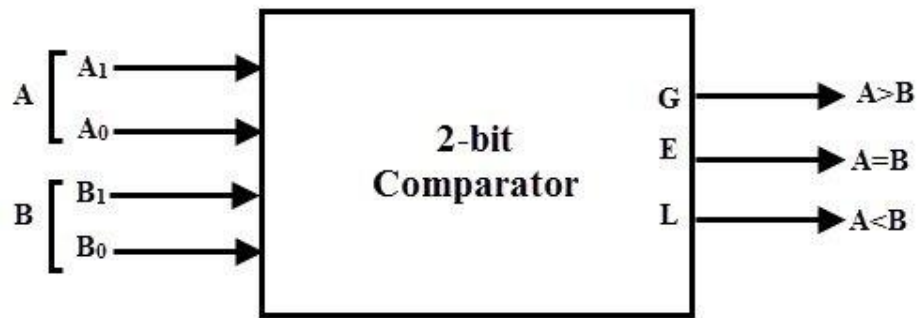
The first number A is designated as A = A1A0 and the second number is designated as B = B1B0. This comparator produces three outputs as G (G = 1 if A>B), E (E = 1, if A = B) and L (L = 1 if A<B).

Truth table

Inputs				Outputs		
A_1	A_0	B_1	B_0	A>B	A=B	A<B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

A>B

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

A=B

A_1A_0 \ B_1B_0	00	01	11	10
00	1	0	0	0
01	0	1	0	0
11	0	0	1	0
10	0	0	0	1

$$A > B: G = A_0\,\overline{B_1}\;\overline{B_0} + A_1\,\overline{B_1} + A_1\,A_0\,\overline{B_0}$$

$$A = B: E = \overline{A_1}\;\overline{A_0}\,\overline{B_1}\;\overline{B_0} + \overline{A_1}\,A_0\;\overline{B_1}\,B_0 + A_1\,A_0\,B_1\,B_0 + A_1\,\overline{A_0}\,B_1\,\overline{B_0}$$

$$= \overline{A_1}\;\overline{B_1}\,(\overline{A_0}\;\overline{B_0} + A_0\,B_0) + A_1\,B_1\,(A_0\,B_0 + \overline{A_0}\;\overline{B_0})$$

$$= (A_0\,B_0 + \overline{A_0}\;\overline{B_0})\,(A_1\,B_1 + \overline{A_1}\;\overline{B_1})$$

$$= (A_0 \text{ Ex-NOR } B_0)\,(A_1 \text{ Ex-NOR } B_1)$$

$$A < B: L = \overline{A_1}\;B_1 + \overline{A_0}\,B_1\,B_0 + \overline{A_1}\;\overline{A_0}\,B_0$$
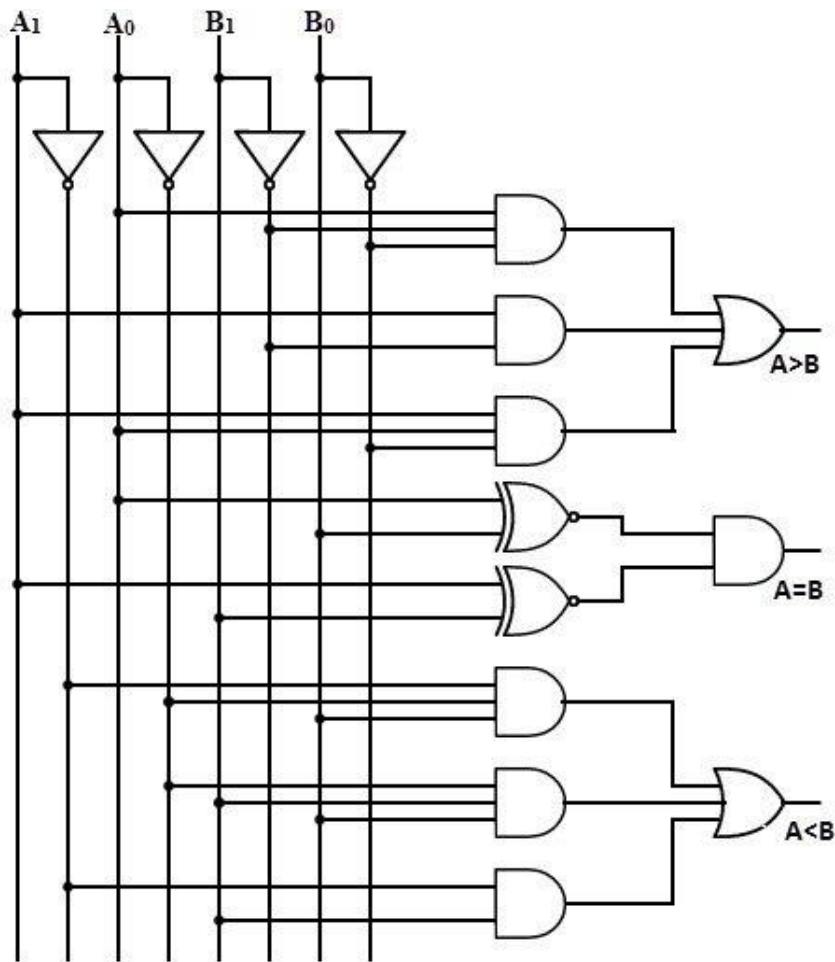
Using IC we can make 2 bit comparator as follows:
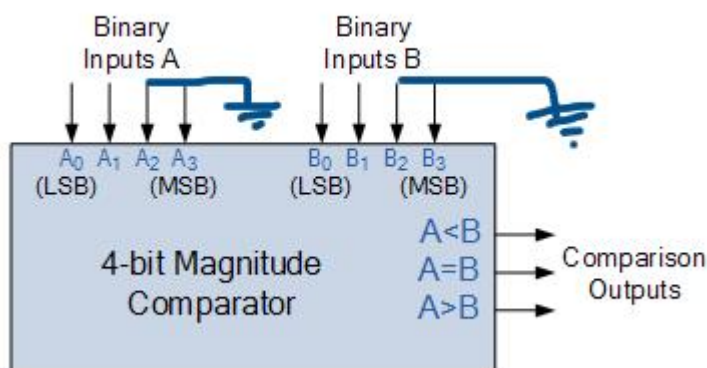
15. What is priority encoder?

A priority encoder is a circuit or algorithm that compresses multiple binary inputs into a smaller number of outputs. The output of a priority encoder is

the binary representation of the original number starting from zero of the most significant input bit.

CARRIES 6 MARKS) ANSWER ANY 5 QUESTION

1. Implement the following function using suitable multiplexer F (A, B, C, D) = $\Sigma m(0,2,5,7)(1, 3, 4, 11, 12, 13, 14, 15$

)

2. Draw the logic diagram of a 2-bit by 2-bit binary multiplier and explain its operation.

Binary multiplication process: A Binary Multiplier is a digital circuit used in digital electronics to multiply two binary numbers and provide the result as output. The method used to multiply two binary numbers is similar to the method taught to school children for multiplying decimal numbers which is based on calculating partial product, shifting them and adding them together. Similar approach is used to multiply two binary numbers. Long multiplicand is multiplied by 0 or 1 which is much easier than decimal multiplication as product by 0 or 1 is 0 or same number respectively. Figure 1 below shows the block diagram of a 2-bit binary multiplier. The two numbers A1A0 and B1B0 are multiplied together to produce a 4-bit output P3P2P1P0. (The maximum product term can be 3 * 3 = 9, which is 1001, a 4-bit number).
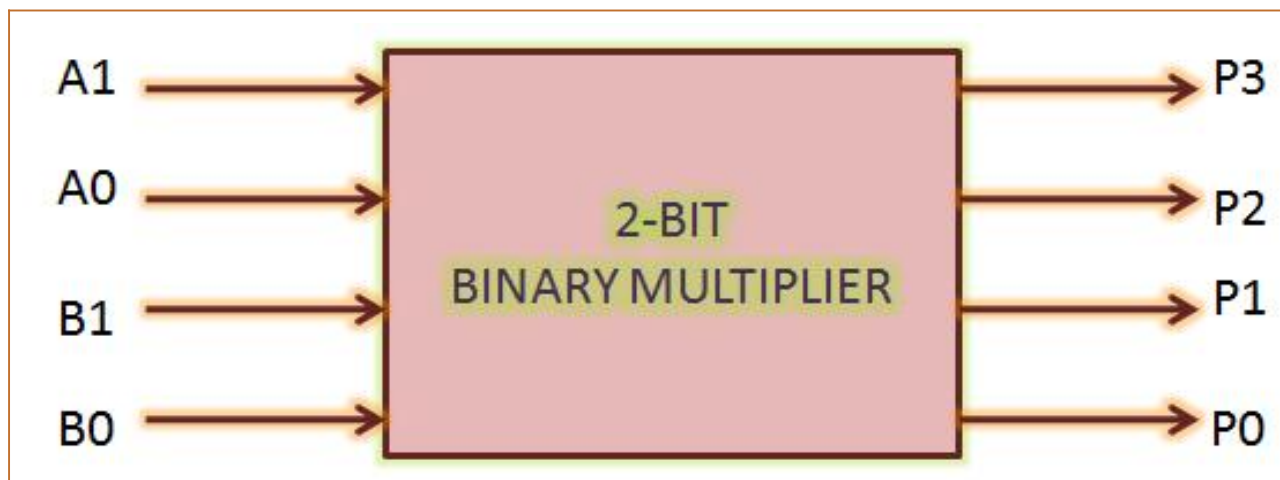
Let us take an example of multiplying two binary numbers as follows. The process is similar to multiplying two decimal numbers, with a difference that the resulting numbers are all binary.

$$110 = 6$$
$$X \quad 011 = 3$$

$$1\ 1\ 0 \qquad\qquad ; 110\ X\ 1$$
$$1\ 1\ 0\ x \qquad\quad ; 110\ X\ 1$$
$$0\ 0\ 0\ x\ x \qquad\quad ; 110\ X\ 0$$

$$1\ 0\ 0\ 1\ 0 = 18$$

Now, we have seen that multiplying a number with binary '0'produces all zeroes, and with '1' reproduces the number. So, multiplying two binary numbers is a straightforward job. It can be implemented without much difficulty using shifters, AND gates and adders.

2-bit binary multiplier circuit implementation: Let us implement a two bit binary multiplier. Let the two binary numbers be A_1A_0 and B_1B_0. The multiplication table will, then, look as:

$$\begin{array}{ccc} & A_1 & A_0 \\ X & B_1 & B_0 \end{array}$$

$$\begin{array}{ccc} & B_0A_1 & B_0A_0 \\ B_1A_1 & B_1A_0 & x \end{array}$$

```
--------------------------------------------------------------
        P3      P2        P1            P0
```

Thus, we get the partial products as:

      P0 = A0*B0

      P1 = A0*B1 xor A1 * B0   ; carry generated here goes to next stage

      P2 = A1*B1 xor (A0*B1) * (A1*B0)

      P3 = A1*B1 and (A0*B1) * (A1*B0)

Thus, we can see that a 2-bit binary multiplier can be implemented using two half-adders only.

Characteristics of a binary multiplication: As mentioned above, a binary multiplier is used to multiply binary numbers. In general, the characteristics of binary multiplication are as follows:

• To multiply two binary numbers, AND gates, shifters and adders are required.

• Product of N*M bit binary numbers in of (N+M) bits.

• N*M AND gates are required to generate partial products of two M*N bit binary numbers.

• Number of adders required = N+M-2

• Speed limiting factor here is to sum up partial products.

3. Draw the block schematic of Magnitude comparator and explain its operation

Magnitude Comparator is a combinational circuit capable of comparing the relative magnitude of two binary numbers. It is one of the two types of digital comparator.
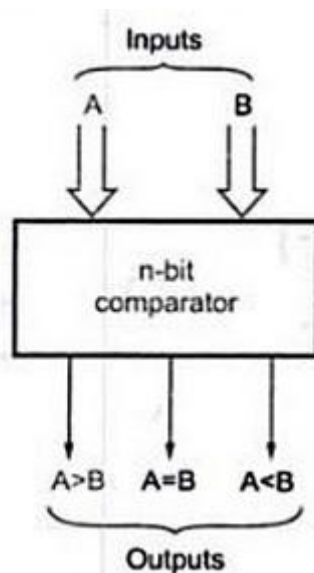
Figure(a): Block diagram of n-bit comparator

Figure(a) shows the block diagram of n-bit magnitude comparator. It accepts two n-bit binary numbers, say A and B as inputs and produces one of the outputs: A>B, A=B and A<B.

One of the outputs will be high depending upon the relative magnitude. That is, output A>B will be high if A is greater than B, output A=B will be high if A and B are equal, and output A<B will be high if A is less than B.

Its logic behaviour is same as adder. It does not return sum or carry.

Magnitude comparators are used in central processing units and microcontrollers.

This basic circuit for a magnitude comparator can be extended for any number of bits.

Four bit magnitude comparators are very popular circuits and are commercially available.

Examples: 74HC85 and CMOS 4063. These are four bit magnitude comparators.

Logic Diagram
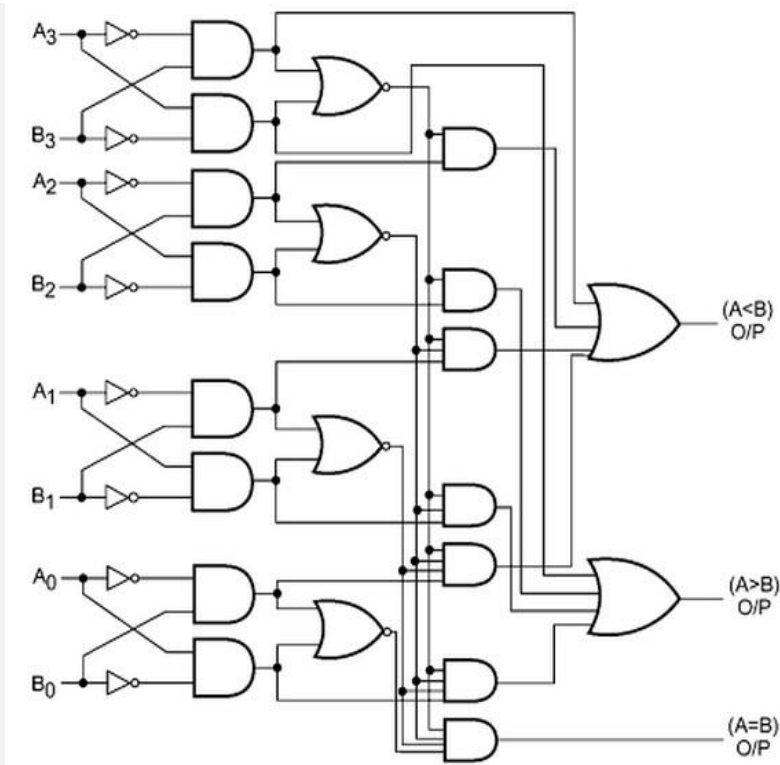
Figure(b): Four bit magnitude comparator

4. Draw & explain the block diagram of a 4-bit parallel adder / Subtractor
     To add two Hex codes, we need four full adders connected in cascade. This is because a <u>hex code</u> can be represented by four binary bits. The four full adders will connect via their CARRY outputs. And depending on the position of the bits, the full adders add, the SUM outputs of the full adders will be connected to the display.
The least significant bit will be connected to the LSB of the display. The most significant bit will be connected to the pin one bit before the MSB of the display. The carry output of the final full adder will be connected to the MSB pin of the display.

We can use two hex 4×4 keypads to generate the input bits, or we can just add the bits manually
Each row of the keypad is connected to a full adder depending on its significance. The first full adder receives inputs from the first row of the hex keypad. The second full adder receives inputs from the second row of the hex keypad and the carry from the first and so on. The resultant combinational logic circuit is shown below.
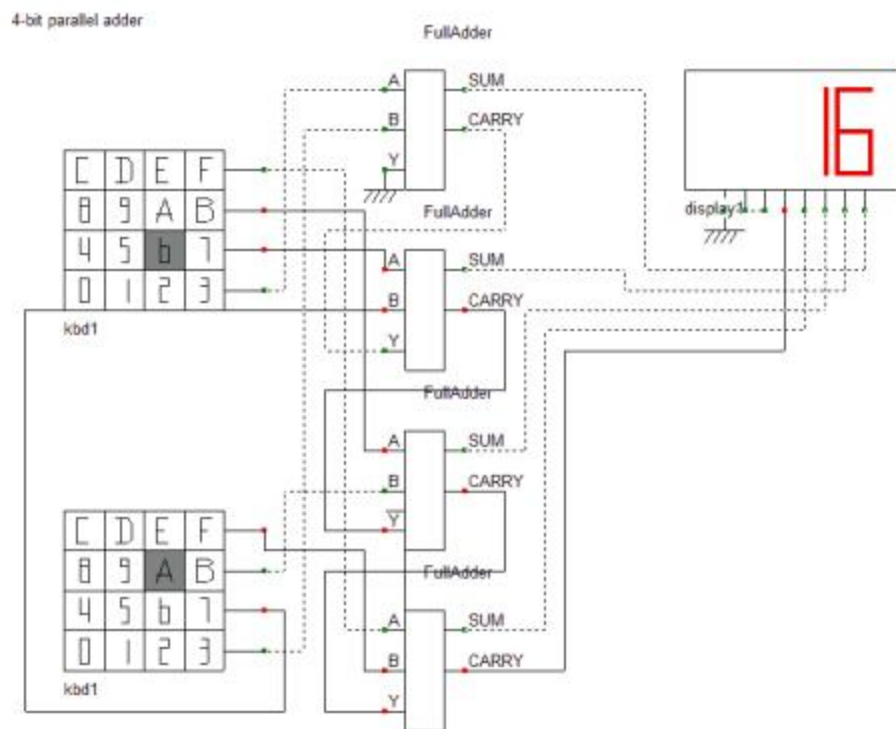
                                                            4-bit parallel
adder

5. Design & implement the conversion circuits for BCD to Excess – 3 code

6. Design a BCD to Gray code converter. Uses don't care

7. Explain the operation of carry look ahead adder with neat diagram

   A digital computer must contain circuits which can perform arithmetic operations such as addition, subtraction, multiplication, and division. Among these, addition and subtraction are the basic operations whereas multiplication and division are the repeated addition and subtraction respectively.

To perform these operations 'Adder circuits' are implemented using basic logic gates. Adder circuits are evolved as Half-adder, Full-adder, Ripple-carry Adder, and Carry Look-ahead Adder.
Among these Carry Look-ahead Adder is the faster adder circuit. It reduces the propagation delay, which occurs during addition, by using more complex hardware circuitry. It is designed by transforming the ripple-carry Adder circuit such that the carry logic of the adder is changed into two-level logic.

4-Bit Carry Look-ahead Adder

In parallel adders, carry output of each full adder is given as a carry input to the next higher-order state. Hence, these adders it is not possible to produce carry and sum outputs of any state unless a carry input is available for that state.

So, for computation to occur, the circuit has to wait until the carry bit propagated to all states. This induces carry propagation delay in the circuit.
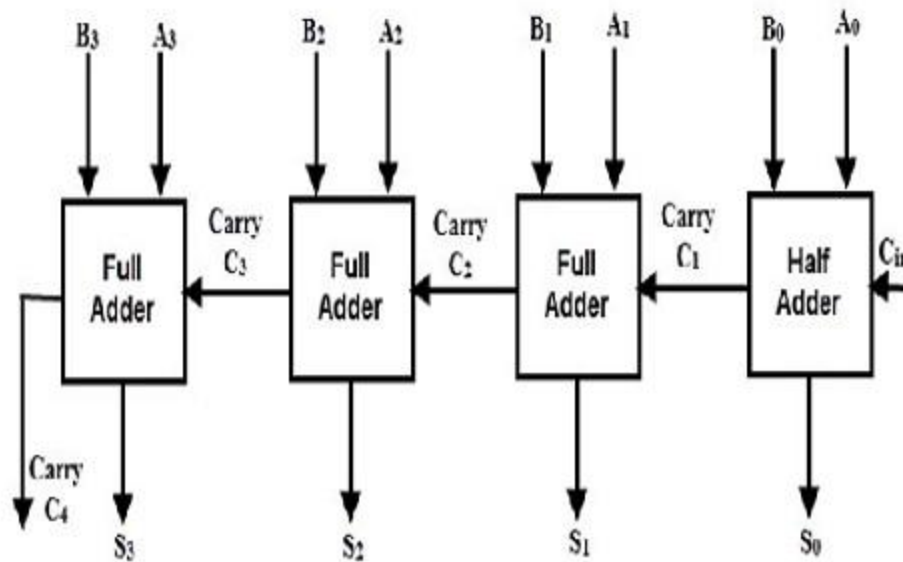
4-bit-Ripple-Carry-Adder

Consider the 4-bit ripple carry adder circuit above. Here the sum S3 can be produced as soon as the inputs A3 and B3 are given. But carry C3 cannot be computed until the carry bit C2 is applied whereas C2 depends on C1. Therefore to produce final steady-state results, carry must propagate through all the states. This increases the carry propagation delay of the circuit.

8. Draw and explain the BCD adder circuit.

The digital systems handles the decimal number in the form of binary coded decimal numbers (BCD). A BCD Adder Circuit that adds two BCD digits and produces a sum digit also in BCD. BCD numbers use 10 digits, 0 to 9 which are represented in the binary form 0 0 0 0 to 1 0 0 1, i.e. each BCD digit is represented as a 4-bit binary number. When we write BCD number say 526, it can be represented as

5	2	6
↓	↓	↓
0 1 0 1	0 0 1 0	0 1 1 0

Here, we should note that BCD cannot be greater than 9.

The addition of two BCD numbers can be best understood by considering the three cases that occur when two BCD digits are added.

Sum Equals 9 or less with carry 0

Let us consider additions of 3 and 6 in BCD.

```
    6        0  1  1  0       ←  BCD for 6
+   3        0  0  1  1       ←  BCD for 3
  ___       _____
    9        1  0  0  1       ←  BCD for 9
```

The addition is carried out as in normal binary addition and the sum is 1 0 0 1, which is BCD code for 9.

Sum greater than 9 with carry 0

Let us consider addition of 6 and 8 in BCD

```
    6        0  1  1  0       ←  BCD for 6
+   8        1  0  0  0       ←  BCD for 8
  ___       _____
   14        1  1  1  0       ←  Invalid BCD number
```

The sum 1 1 1 0 is an invalid BCD number. This has occurred because the sum of the two digits exceeds 9. Whenever this occurs the sum has to be corrected by the addition of six (0110) in the invalid BCD number, as shown below

```
       6            0  1  1  0        ←  BCD for 6
   +   8            1  0  0  0        ←  BCD for 8
     ___           _____
      14            1  1  1  0        ←  Invalid BCD number
                +   0  1  1  0        ←  Add 6 for correction
                   _____
 0  0  0  1         0  1  0  0
                                      ←  BCD for 14
  _____/         _____/
      1                  4
```

After addition of 6 carry is produced into the second decimal position.

Sum equals 9 or less with carry 1

Let us consider addition of 8 and 9 in BCD

8					1	0	0	0	←	BCD for 8
+ 9					1	0	0	1	←	BCD for 9
17	0	0	0	1	0	0	0	1	←	Incorrect BCD result

In this, case, result (0001 0001) is valid BCD number, but it is incorrect. To get the correct BCD result correction factor of 6 has to be added to the least significant digit sum, as shown below

8					1	0	0	0	←	BCD for 8
+ 9					1	0	0	1	←	BCD for 9
17	0	0	0	1	0	0	0	1	←	Incorrect BCD result
+	0	0	0	0	0	1	1	0	←	Add 6 for correction
	0	0	0	1	0	1	1	1	←	BCD for 17

Going through these three cases of BCD addition we can summarise the BCD addition procedure as follows :

1. **Add two BCD numbers using ordinay binary addition.**
2. **If four-bit sum is equal to or less than 9, no correction is needed. The sum is in proper BCD form.**
3. **If the four-bit sum is greater than 9 or if a carry is generated from the four-bit sum, the sum is invalid.**
4. **To correct the invalid sum, add 0110_2 to the four-bit sum. If a carry results from this addition, add it to the next higher-order BCD digit.**

Thus to implement BCD Adder Circuit we require :

- **4-bit binary adder for initial addition**
- **Logic circuit to detect sum greater than 9 and**
- **One more 4-bit adder to add 0110_2 in the sum if sum is greater than 9 or carry is 1.**

The logic circuit to detect sum greater than 9 can be determined by simplifying the boolean expression of given BCD Adder Truth Table.

Inputs				Output
S_3	S_2	S_1	S_0	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Table 3.10

Fig. 3.31

$$Y = S_3 S_2 + S_3 S_1$$

With this design information we can draw the BCD Adder Block Diagram, as shown in the Fig. 3.32.
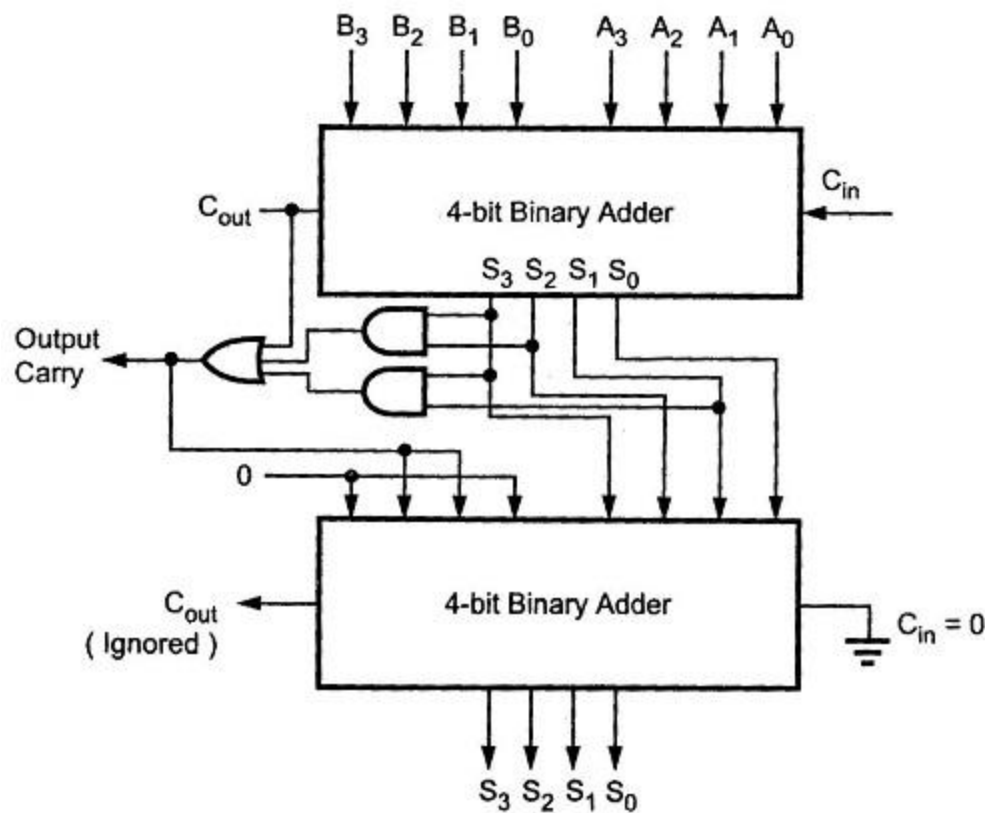
Fig. 3.32 Block diagram of BCD adder

9. Design a seven segment decoder circuit to display the numbers from 0 to 3.

10. Design & explain the working of Gray to BCD converter.

Gray codes are used in rotary and optical encoders, Karnaugh maps, and error detection. The hamming distance of two neighbours Gray codes is always 1 and also first Gray code and last Gray code also has Hamming distance is always 1, so it is also called *Cyclic codes*. You can convert a Gray code to Binary number using two methods.

Using Karnaugh (K) - map –

You can construct Gray codes using other methods but they may not be performed in parallel like given above method. For example, 3 bit Gray codes can be contracted using K-map which is given as following below:
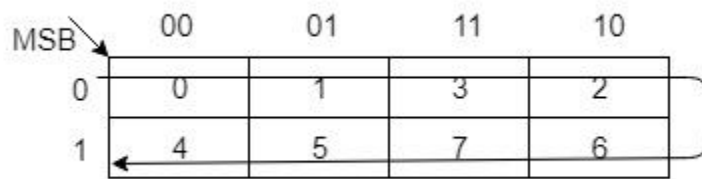
Decimal	Gray Code	Binary
0	000	000
1	001	001
2	011	010
3	010	011
4	110	100
5	111	101
6	101	110
7	100	111

Using Exclusive-Or ($\oplus$) operation –

This is very simple method to get Binary number from Gray code. These are following steps for *n*-bit binary numbers –

- The Most Significant Bit (MSB) of the binary code is always equal to the MSB of the given binary number.

- Other bits of the output binary code can be obtained by checking gray code bit at that index. If current gray code bit is 0, then copy previous binary code bit, else copy invert of previous binary code bit.

PART III

UNIVERSITY EXAM QUESTION PATTERN – 6 MARK (EACH QUESTION

CARRIES 2 MARKS) ANSWER ANY 10 QUESTION

1. Compare Asynchronous and Synchronous sequential logic.

S. No.	Synchronous sequential circuit	Asynchronous sequential circuit
1	Synchronous sequential circuits are digital circuits governed by clock signals.	Asynchronous sequential circuits are digital circuits that are not driven by clock. They can be called as self-timed circuits.
2	Output behavior depends on the input at discrete time.	Output depends on the sequence in which the input changes.
3	In synchronous sequential circuits inputs and outputs are considered at discrete time instants.	In asynchronous sequential circuits input and output signals are defined at every value of time.
4	In synchronous sequential circuits memory elements are used like clocked flip flop.	In asynchronous sequential circuits un-clocked memory elements are used like un-clocked flip flop or time delay elements.

5	The time variable is discrete.	The time variable is continuous.
6	Synchronous sequential circuits are easier to describe, analyze and design.	Asynchronous sequential circuits are more difficult to describe, analyze and design.
7	Due to the presence of clock pulse the operating speed of synchronous sequential circuits is low.	Because of absence of clock pulse, it can be operate faster than Synchronous sequential circuits
8	In these circuits change in state occurs in response to clock pulse.	In these circuits state change occurs whenever input variable change.
9	In synchronous circuits no timing problem in feedback path.	In asynchronous circuits timing problem involved in feedback path.
10	These are more expensive.	These are economical.
11	It has complex circuitry.	It has few components.
12	Any number of inputs can change simultaneously (during the absence of	Only one input is allowed to change at a time in the case of level inputs and only one pulse input is allowed to be present in the case of the pulse input.

13	clock).	
	Synchronous circuits are used in counters, shift registers, memory units.	On the other hand, Asynchronous circuits are used in low power and high-speed operations such as simple microprocessors, digital signal processing units, and in communication systems for email applications, internet access, and networking.

2. Draw the state diagram and characteristics equation of a D FF.

D	Q(next)
0	0
1	1

CHARACTERISTIC EQUATION = Q(next) = D

3. What is latch? What is the difference between latch and flip flop?

A latch is an electronic device that changes its output immediately on the basis of the applied input. One can use it to store either 0 or 1 at a specified time. A latch contains two inputs- SET and RESET, and it also has two outputs. They complement each other. One can use a latch for storing one bit of data. It is a memory device- just like the flip-flop. But it is not synchronous, and it does not work on the edges of the clock like the flip-flop.

The major difference between flip-flop and latch is that the flip-flop is an edge-triggered type of memory circuit while the latch is a level-triggered type. It means that the output of a latch changes

whenever the input changes. On the other hand, the latch only changes its state whenever the control signal goes from low to high and high to low.

4. Realize T Flip Flop using SR Flip Flop

5. How does the JK FF differ from an SR FF in its basic equation?

- In **J-K flip flop** when **both inputs are HIGH**, the output **toggles** i.e. it changes from high to low and low to high periodically when both the inputs are 1.
- For an **SR flip flop**, however, **when both the inputs are HIGH**, we encounter an **invalid state**, which is not present for a JK flip flop.
- This is explained with the help of the following function table:

FF triggers on positive transition
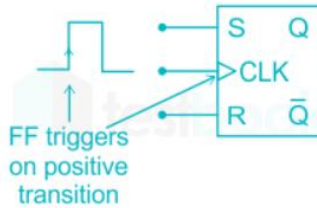
S	R	CLK	Q
0	0	↑	Q0 (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	Ambiguous (Invalid)

Similarly, for a JK flip flop, the block diagram along with the function table is as shown:
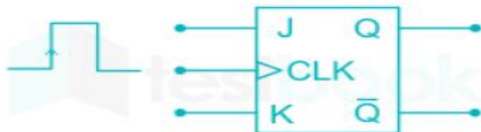
J	K	CLK	Q
0	0	↑	Q_0 (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	$\bar{Q}_0$ (toggles)

6. Define Synchronous counter.

synchronous counter A counter consisting of an interconnected series of flip-flops in which all the flip-flop outputs change state at the same instant, normally on application of a pulse at the counter input. These counters have advantages in speed over asynchronous ripple counters, in which the output must propagate along the chain of flip-flops after the application of a pulse at the count input. See also cascadable counter, shift counter.

7. Define Setup and Hold time.

Setup time is the amount of time required for the input to a Flip-Flop to be stable *before* a clock edge. Hold time is similar to setup time, but it deals with events after a clock edge occurs. **Hold time is the minimum amount of time required for the input to a Flip-Flop to be stable *after* a clock edge**.

8. What is the condition on JK FF to work as D FF?

$$J = D$$ $$K = \bar{D}$$

From Figure 6, it can be seen that the given JK flip-flop can be converted into a D-type flip-flop by driving its J and K input pins with the D input and its negation, respectively. Thus the additional hardware component required would be a NOT gate, resulting in the digital system shown in Figure 7.
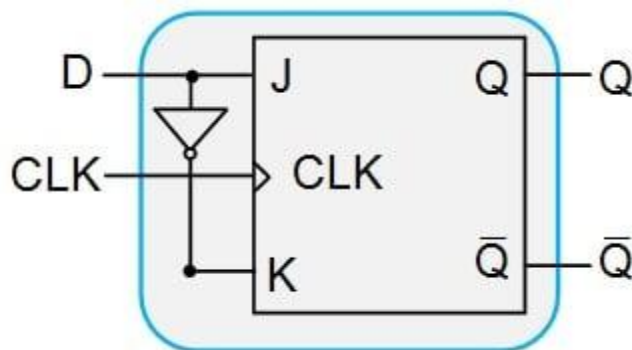
let us construct the JK-to-D verification table as shown in Figure 8.

JK to D Verification Table

Truth T

Input	Intermediate Inputs				Outputs	
D	Q	$\bar{Q}$	J = D	K = $\bar{D}$	Q	$\bar{Q}$
0	0	1	0	1	0	1
0	1	0	0	1	0	1
1	0	1	1	0	1	0
1	1	0	1	0	1	0

Truth Table of JK Flip-Flop

Input	P
D	
0	
0	
1	
1	

Figure 8: Comparison between the JK-to-D verification table and the truth table of a D flip-flop. Click to enlarge.

From the figure, it can be clearly seen that the entries in the first, second, and sixth columns of the JK-to-D verification table (shaded in beige) are the same as those in the D flip-flop's truth table. Thus, it can be concluded that the conversion process of JK flip-flop into D-type was successful.

9. What is race around condition? How do you eliminate it?

Toggling of the output more than once during the same clock pulse is called race around condition. It can be eliminated using an RC network (edge triggering) at the clock input or by using Master-slave JK flip flop.

10. Mention any two differences between the edge triggering and level triggering.

The **main difference** between edge and level triggering is that **in edge triggering, the output of the sequential circuit changes during the high voltage period or low voltage period while, in level triggering, the output of the sequential circuit changes during transits from the high voltage to low voltage or low voltage to high voltage.**

11. Draw the state diagram of MOD -10 counters.
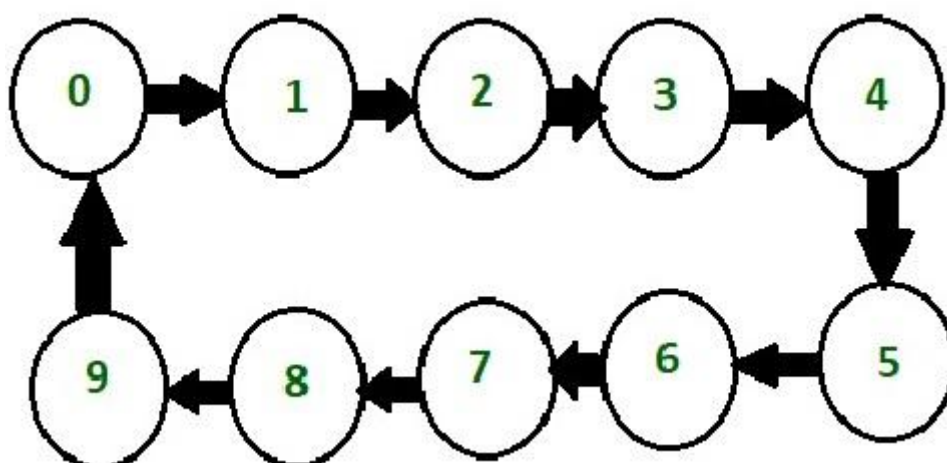
Circuit diagram mod-10 counter.

www.f-alpha.net

www.f-alpha.net
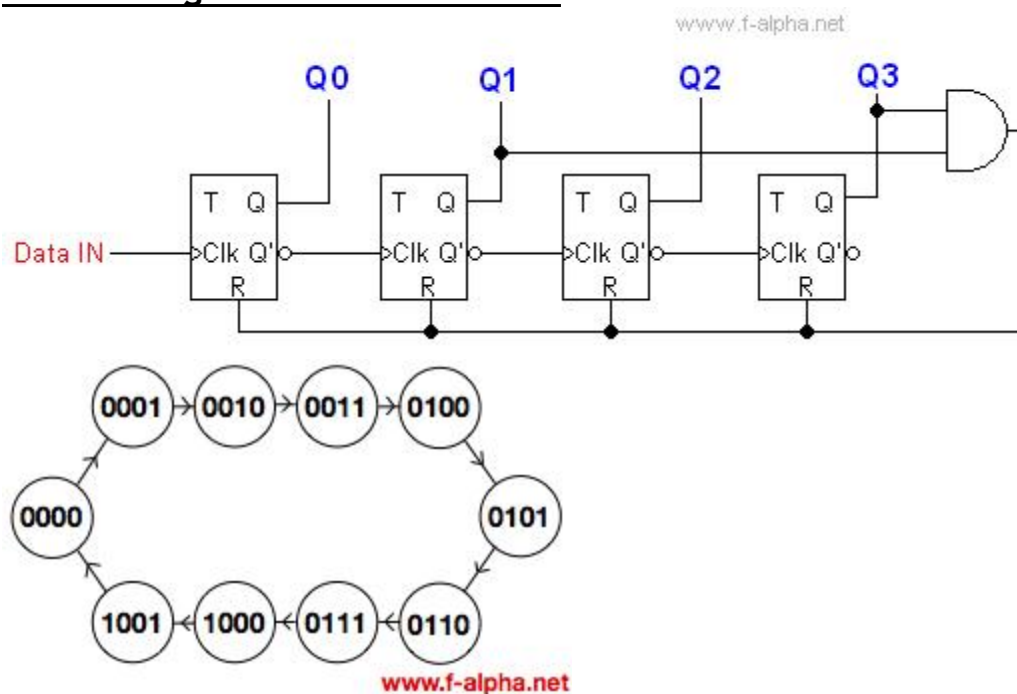
12. What is sequential circuit? Give some example.

Computer circuits consist of combinational logic circuits and sequential logic circuits. Combinational circuits produce outputs immediately when their input changes. Sequential circuits require clocks to control their changes of state
Examples of sequential circuits are flip-flop, register, counter, clocks, etc.

13. Draw a NAND based logic diagram of Master Slave JK FF.

14. Convert Transparent flip flop into a JK flip flop.

15. Differentiate Asynchronous and Modulus counter

ANSWER ANY 5 QUESTION

1. i). Design and explain the working of an 4-bit Parallel counter

4 bit Synchronous UP Counter

The 4 bit up counter shown in below diagram is designed by using JK flip flop. External clock pulse is connected to all the flip flops in parallel.

For designing the counters JK flip flop is preferred .The significance of using JK flip flop is that it can toggle its state if both the inputs are high, depending on the clock pulse.

The inputs of first flip flop are connected to HIGH (logic 1), which makes the flip flop to toggle, for every clock pulse entered into it. So the synchronous counter will work with single clock signal and changes its state with each pulse.
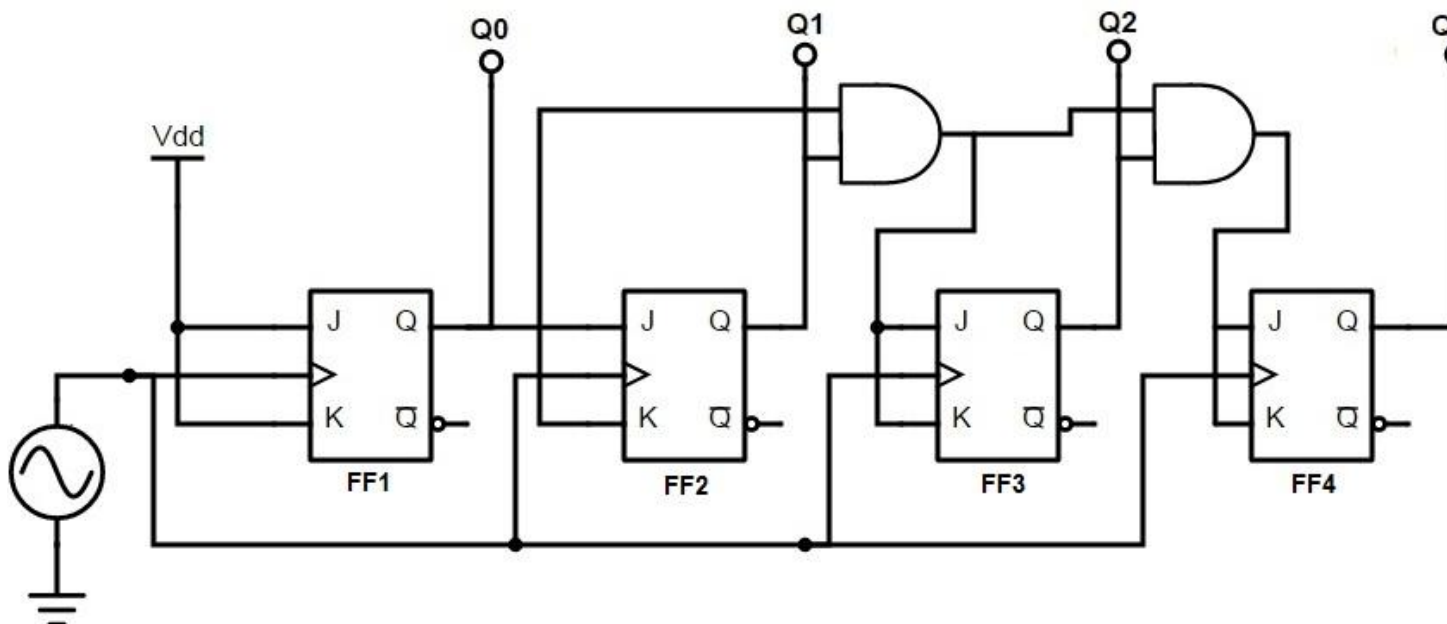
The output of first JK flip flop (Q) is connected to the input of second flip flop. The AND gates (which are connected externally) drives the inputs of other two flip flops . The inputs of these AND gates , are supplied from previous stage lip flop outputs.

If inputs of FF2 are connected directly to the Q1 output of FF1 , the counter would not function properly. This is because , the Q1 value is high at count of 210 , this means that the FF2 flip flop will toggle for the 3rd clock pulse. This results in wrong counting operation, gives the count as 710 instead of 410.

To prevent this problem AND gates are used at the input side of FF2 and FF3. The output of the AND gate will be high only when the Q0, Q1 outputs are high. So for the next clock pulse, the count will be 00012.

Similarly, the flip flop FF3 will toggle for the fourth clock pulse when Q0, Q1 and Q2 are high. The Q3 output will not toggle till the 8th clock pulse and will again remain high until 16th clock pulse. After the 16th clock pulse, the q outputs of all flip flops will return to 0.

Operation

In the up counter the 4 bit binary sequence starts from 0000 and increments up to 1111.Before understanding the working of the above up counter circuit know about JK Flip flop.

In the above circuit as the two inputs of the flip flop are wired together. So , there are only two possible conditions that can occur, that is, either the two inputs are high or low.

If the two inputs are high then JK flip-flop toggles and if both are low JK flip flop remembers i.e. it stays in the previous state.

Let us see the operation. Here clock pulse indicates edge triggered clock pulse .

1.) In the first clock pulse, the outputs of all the flip flops will be at 0000.

2.)In the second clock pulse, as inputs of J and k are connected to the logic high, output of JK flip flop(FF0) change its state .Thus the output of the first flip-flop(FF0) changes its state for every clock pulse .This can be observed in the above shown sequence .The LSB changes its state alternatively. Thus producing -0001

3.) In the third clock pulse next flip flop (FF1) will receive its J K inputs i.e (logic high) and it changes its state. At this state FF0 will change its state to 0. And thus input on the FF1 is 0.Hence output is -0010

4.) Similarly, in the fourth clock pulse FF1 will not change its state as its inputs are in low state, it remains in its previous state. Though it produces the output to FF2, it will not change its state due to the presence of AND gate. FF0 will again toggle its output to logic high state. Thus Output is 0011.

5.) In the fifth clock pulse, FF2 receives the inputs and changes its state. While, FF0 will have low logic on its output and FF1 will also be low state producing 0100.

This process continuous up to 1111.Working can be explained in the below table. The above mentioned working of synchronous counter can be clearly given in the below table.

Clock pulse	Q4	Q3	Q2	Q1	Outpu
1st Clock Pulse	Low	Low	Low	Low	0
2nd Clock Pulse	Low	Low	Low	High (starts toggling)	1
3rd Clock Pulse	Low	Low	High	Low	2
4th Clock Pulse	Low	Low (AND condition is not satisfied)	High (Remembers)	High	3
5th Clock Pulse	Low	High	Low (toggles)	Low	4
6th Clock Pulse	Low (AND condition not satisfied)	High (Remembers)	Low (remembers)	High	5
7th Clock Pulse	Low (AND condition satified)	High	High	Low	6
8th Clock Pulse	Low	High	High	High	7
9th Clock Pulse	High	Low	Low	Low	8
10th Clock Pulse	High(Remembers)	Low	Low	High	9
11th Clock Pulse	High	Low	High	Low	10
12th Clock Pulse	High	Low	High	High	11
13th Clock Pulse	High	High	Low	Low	12
14th Clock Pulse	High	High	Low	High	13
15th Clock Pulse	High	High	High	Low	14
16th Clock Pulse	High	High	High	High	15

The below table shows the outputs of 4 flip flops Q1, Q2, Q3, Q4.The first flip-flop toggles on every edge triggered pulse .While the second one triggers only if its inputs are high at a given clock pulse. The third flip-flop toggles if the two ouputs Q1 and Q2 are high. Similarly, Q4 will toggle if all the three Q1,Q2,Q3 are high.

After reaching zero again the three flip flops toggles to logic low i.e 0000 and again count starts.
Timing diagram for up counter is shown below.
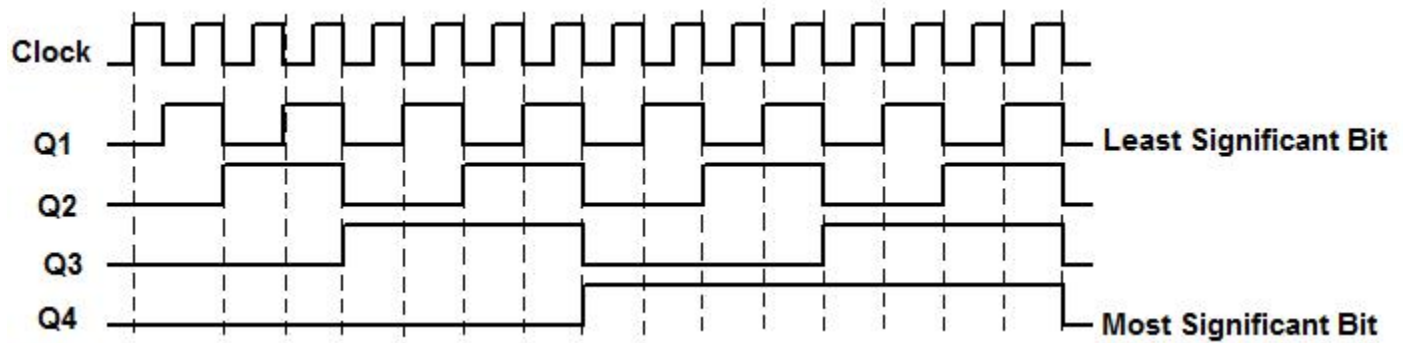
ii). Design and working of a BCD ripple counter with timing diagram.

A **Binary counter** is a **2-Mod counter** which counts up to 2-bit state values, i.e., 22 = 4 values. The flip flops having similar conditions for toggling like T and JK are used to construct the **Ripple counter**. Below is a circuit diagram of **a binary ripple counter**.

In the circuit design of the binary ripple counter, two JK flip flops are used. The high voltage signal is passed to the inputs of both flip flops. This high voltage input maintains the flip flops at a state 1. In JK flip flops, the negative triggered clock pulse use.
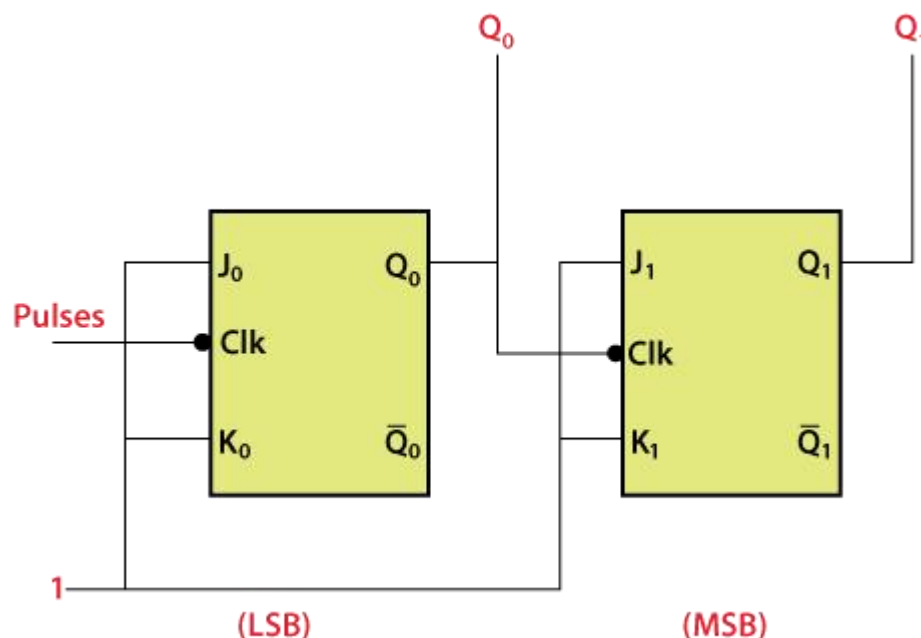
The outputs Q_0 and Q_1 are the LSB and MSB bits, respectively. The truth table of JK flip flop helps us to understand the functioning of the counter.

J_n	K_n	Q_{n+1}
0	0	Q_n
1	0	1
0	1	0
1	1	Q_n

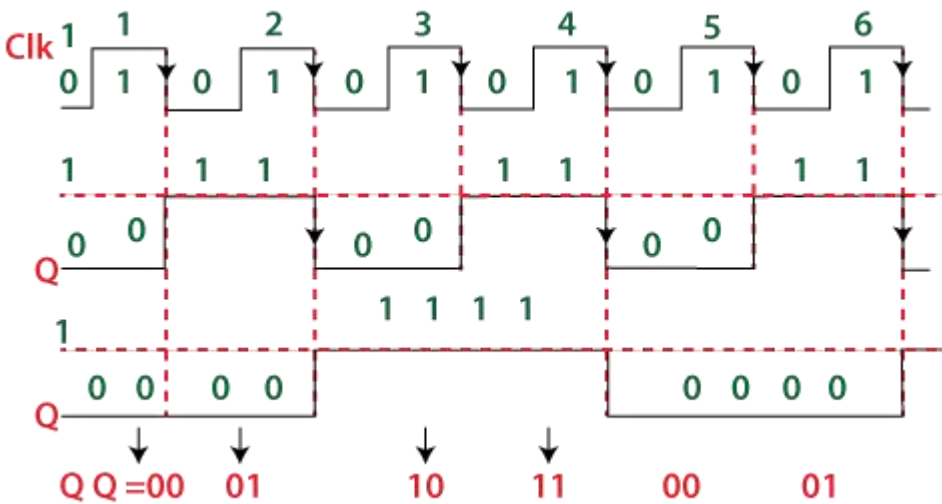When the high voltage to the inputs of the flip flops, the fourth condition is of the JK flip flop occurs. The flip flops will be at the state 1 when we apply high voltage to the input of the flip-flop. So, the states of the flip flops passes are toggled at the negative going end of the clock pulse. In simple words, the flip flop toggle when the clock pulse transition takes place from 1 to 0.

The state of the output Q_0 change when the negative clock edge passes to the flip flop. Initially, all the flip flops are set to 0. These flip flop changes their states when the passed clock goes from 1 to 0. The JK flip flop toggles when the inputs of the flip flops are one, and then the flip flop changes its state from 0 to 1. For all the clock pulse, the process remains the same.

Number of input pulses	Q_1	Q_0
0	–	-
1	0	0
2	0	1
3	1	0
4	1	1

The output of the first flip flop passes to the second flip flop as a clock pulse. From the above timing diagram, it is clear that the state of the second flip flop is changed when the output Q_0 goes transition from 1 to 0. The outputs Q_0 and Q_1 treat as LSB and MSB. The counter counts the values 00, 01, 10, 11. After counting these values, the counter resets itself and starts counting again from 00, 01, 10, and 1. The count values until the clock pulses are passed to J_0K_0 flip flop

2 i).Design and explain the working of an 4-bit Up/Down ripple counter
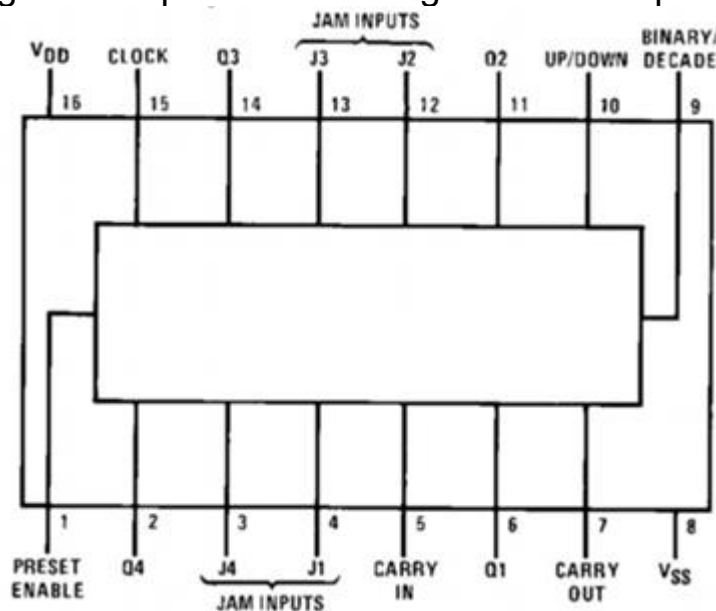
The block diagram shows the layout of the inputs and outputs of this component where Q4..Q1 are the four bits that comprise the binary encoded output that drive the BreadBoard's D/A convertor. The other two important signals are the CLOCK and the UP/DOWN inputs.

CD4029 Signal Descriptions

Internally the counter comprises a set of logic gates configured to implement the arithmetic addition operator (grab the data sheet for the full details). Normally the counter increments the 4 bit word (Q4,Q3,Q2,Q1) by one every time the clock input is toggled.

If the UP/DOWN input is asserted the counter counts down (subtracts one) upon each clock cycle instead.

These two modes of operation are what the Breadboard One project uses but we can run these two modes in isolation by modifying the circuit to simply disconnect the UP/DOWN input from the output of the SCHMITT trigger (as we will show below).

The CARRY OUT and CARRY IN signals are used when more than one counter are used "in cascade". Simply connecting the CARRY OUT of one counter to the CARRY IN of a second one, an 8 bit counter can be built where Q4..Q1 of the first are the low four bits and Q4..Q1 of the second are the high four bits. The CARRY signal is generated each time the counter reaches its limit and "rolls over" (to start the count again).

Signal	Type	Description
CLOCK	Mixed	Counter clock input.
UP/DOWN	Input	Increment or Decrement Count.
Q4..Q1	Output	4 bit binary encoded outputs.
J4..J1	Inputs	4 bit binary encoded "jam" inputs.
BINARY/ DECADE	Input	Binary or Decade counting mode.
CARRY IN	Input	Counter "carry in" input.
CARRY OUT	Output	Counter "carry out" output.
PRESET ENABLE	INPUT	Enable the jam inputs.

Timing Diagrams and Logic Analysis

A logic diagram is the easiest way describe the operation of a digital circuit like this.

Binary Mode

| CLOCK | CARRY IN | UP/DOWN | BINARY/DECADE | PRESET ENABLE | J1 | J2 | J3 | J4 | Q1 | Q2 | Q3 | Q4 | CARRY OUT |
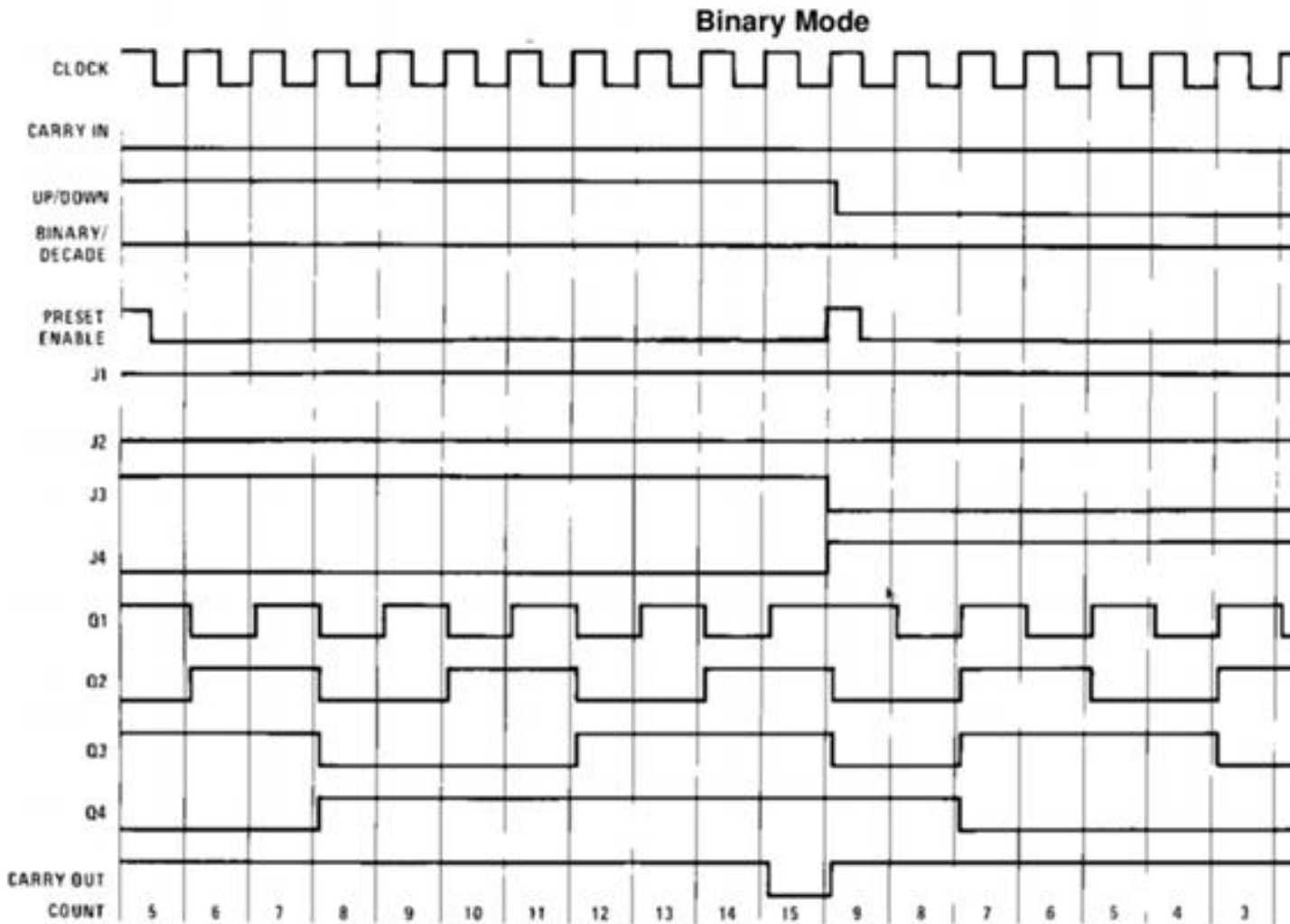
Binary Counter Logic Diagram

For each clock cycle (at the top of the diagram) the four bits cycle in a binary encoded sequence in this case starting at 5, counting up to 15 before being "jammed" to 9 and then counting down to zero and wrapping. Our use of the counter in Breadboard One is simpler in that we're using the UP/DOWN signal but not the jam or carry and to keep it even simpler, here we've asserted the counter as always UP and we observe the result on BS10 as:
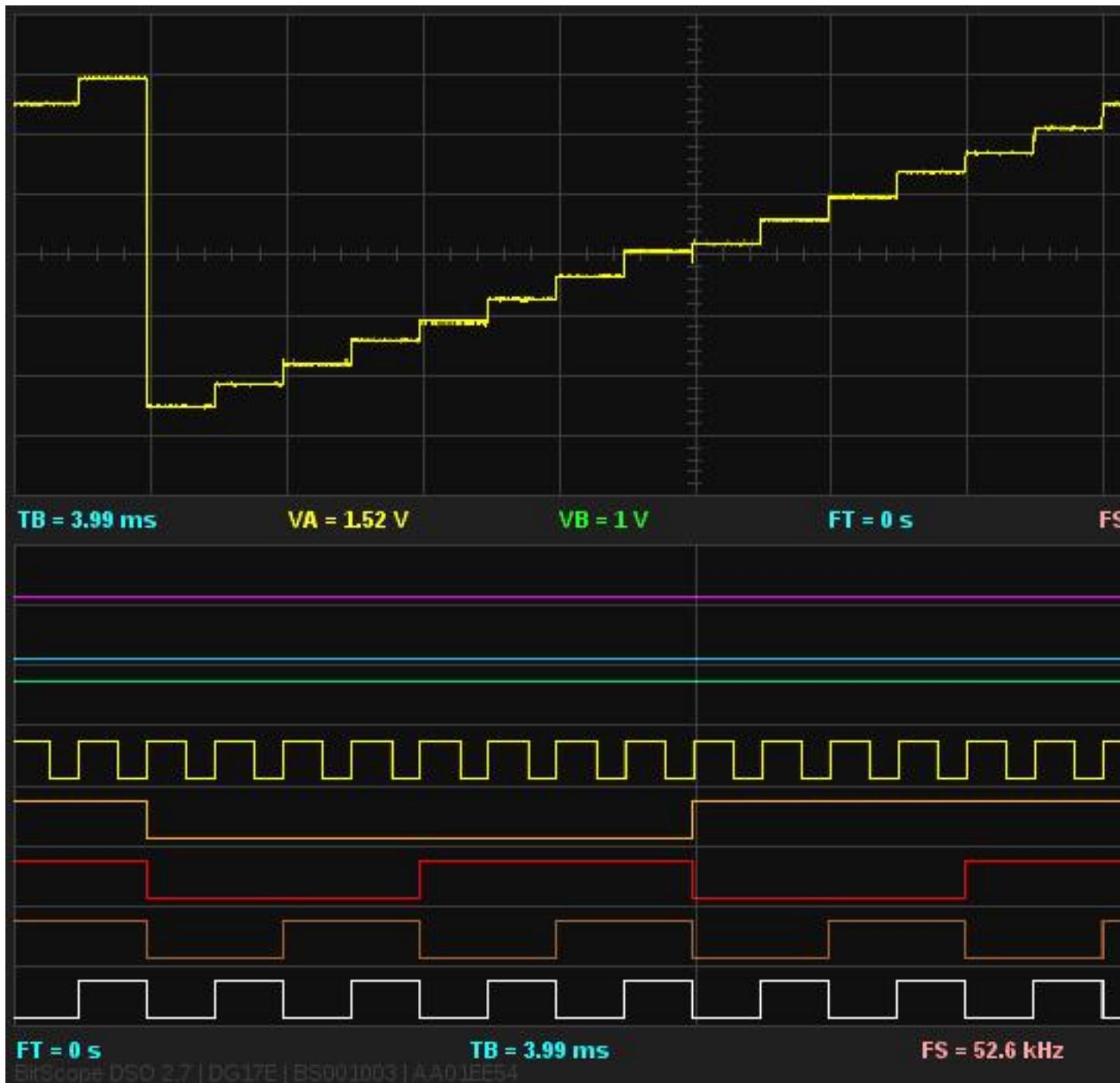
Binary Counter, Mixed Signal Analysis, UP Count

The top half of the display shows the binary encoded counter output as an analog signal produced by the D/A convertor. We'll explain the operation of this component in a future post. For now it's enough to understand that it shows an analog representation of the 4 bit counter output on Q1..Q4. These signals are shown on BitScope's logic channels 0..3 (white, brown, red and orange) and you can see their combined value aligns with the analog signal level for each value.

The clock input (driven by BitScope's waveform generator) is logic channel 5 (yellow) and the UP/DOWN signal is on channel 6 (green). Note that it remains high so the counter increments from 0 to 15 before wrapping and starting again.

Here's the same circuit with one modification, we've pulled the UP/DOWN signal on channel 6 (green) low:

TB = 3.99 ms   VA = 1.52 V   VB = 1 V   FT = 15.9 ms   FS

FT = 15.9 ms   TB = 3.99 ms   FS = 52.6 kHz

BitScope DSO 2.7 | DG17E | BS001003 | AA01EE54
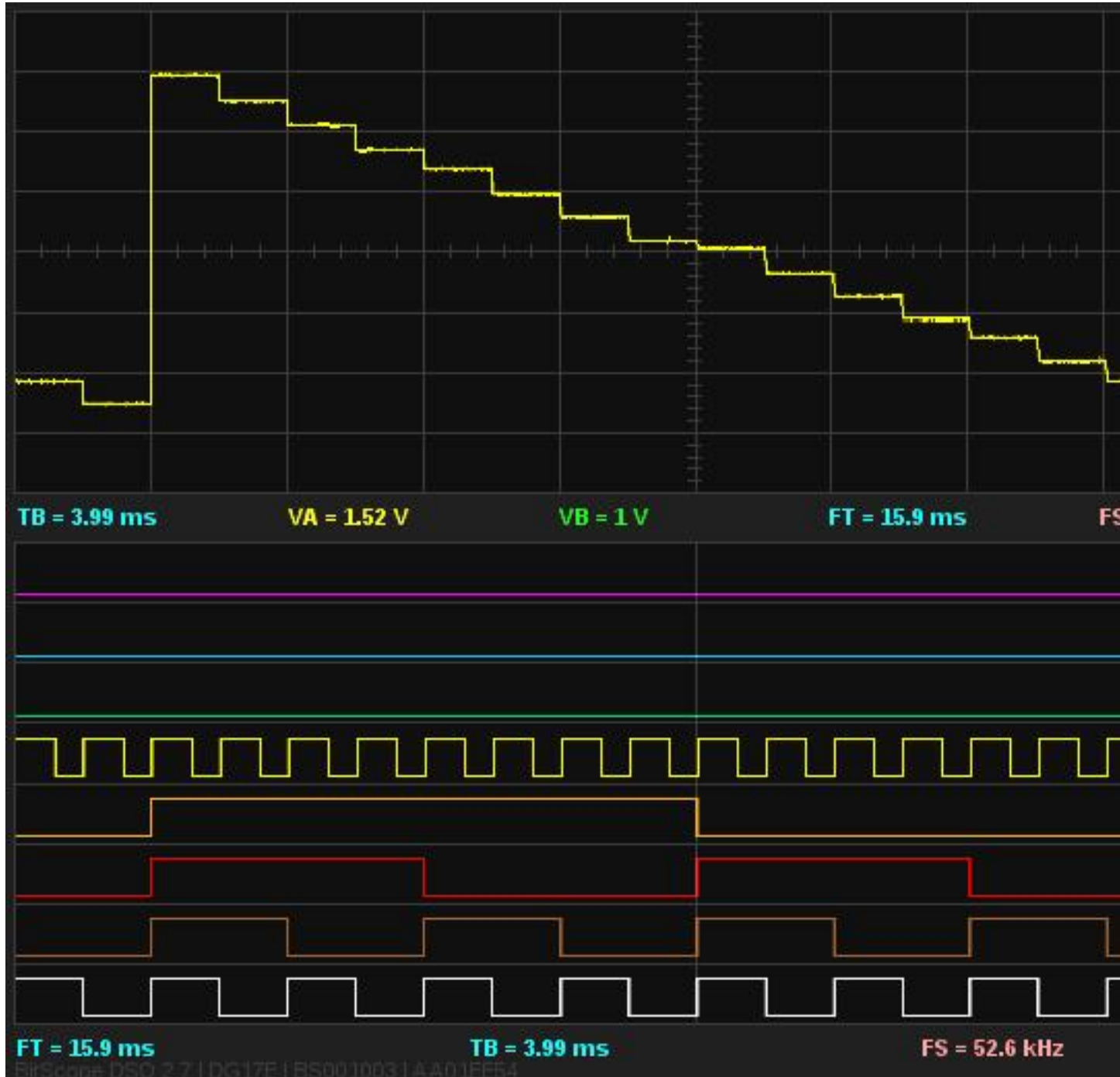
Binary Counter, Mixed Signal Analysis, DOWN Count

It's very easy with Breadboard One to re-arrange the circuit to try all sorts of variations on this theme and observe all the signals, digital and analog, using the BitScope but for now we have the full monty:

TB = 3.99 ms   VA = 1.44 V   VB = 1 V   FT = 15.9 ms   FS

FT = 15.9 ms   TB = 3.99 ms   FS = 52.6 kHz
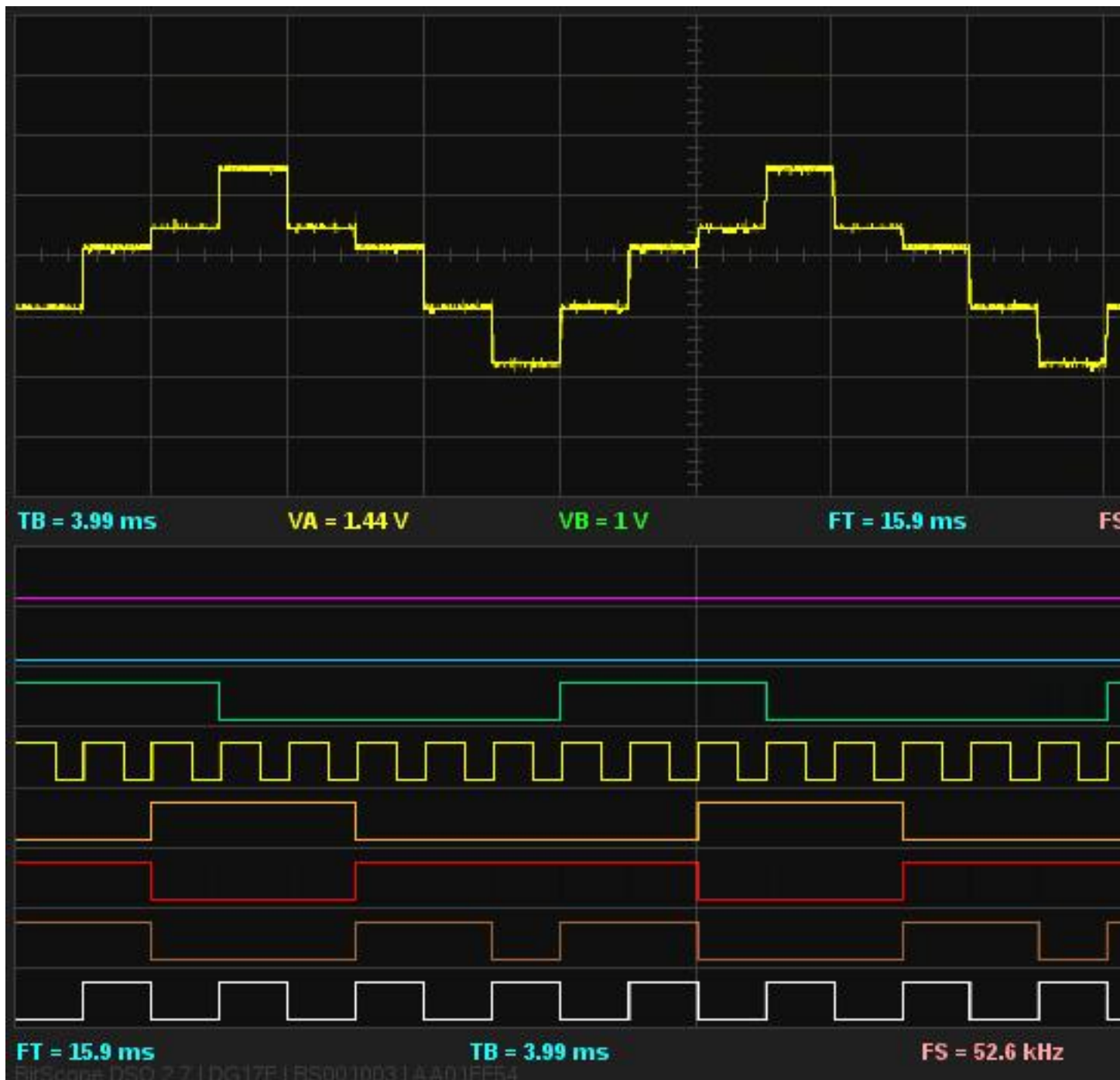
Binary Counter, Mixed Signal Analysis, Breadboard One

ii). Design and working of a synchronous MOD- 5 counter.

Synchronous Counters: It means that all flip-flops are clocked concurrently. The clock pulses drive the clock input of each flip-flop together hence there is no propagation delay.

Mod-5 Counter Synchronous Counter: This have five counter states. The counter design table for such counter shows the three flip-flop and their states also (0 to 5 states), as in table (a), the 6 inputs needed for the three flip-flops. The flip-flop inputs needed to step up the counter from the current to the next state have been worked out along with the assist of the excitation table illustrated in the table.

Input pulse Count	Counter States			Flip-Flop Inputs					
	A	B	C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	1	X	0	X	0	X
1	1	0	0	X	1	1	X	0	X
2	0	1	0	1	X	X	0	0	X
3	1	1	0	X	1	X	1	1	X
4	0	0	1	0	X	0	X	X	1
5(0)	0	0	0						

Table (a) counter Design Table for Mod-5 Counter

A flip-flop: The first state is 0. This change to 1 after the clock pulses. Thus JA must be 1 and KA may be 0 or 1 (i.e. X).

B flip-flop: The first state is 0 and this keeps unchanged after the clock pulse. Thus JB must be 0 and KB may be 0 or 1 (i.e. X)

C flip-flop: The state keeps unchanged. Thus Jc must be 0 and KC must be X. The flip-flop input values are entered in Karnaugh maps demonstrated in Table (b) [(i) (ii) (iii) (iv) (v) and (vi)] and a boolean expression is determined for the inputs to the 3-flip-flops and after that each expression is simplified. All the counter states have not been utilized; X's (don't) are entered to indicate un-utilized states. For each input the simplified expressions demonstrated under each map. At last, these minimal expressions for the flip-flop inputs are utilized to illustrate a logic diagram for the counter that is demonstrated in fig. (b).

(i) Map for J_A

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	1	0	X	1
A	X	X	X	X

$J_A = \overline{C}$

(ii) Map for K_A

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	X	X	X	X
A	1	X	X	1

$K_A = 1$

(iii) Map for J_B

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	0	0	X	X
A	1	X	X	X

$J_B = A$

(iv) Map for K_B

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	X	X	X	0
A	X	X	X	1

$K_B = A$

(v) Map for J_C

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	0	X	X	0
A	0	X	X	1

$J_C = AB$

(vi) Map for K_C

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	$B\,C$	$B\,\overline{C}$
$\overline{A}$	X	1	X	X
A	X	1	X	X

$K_C = 1$

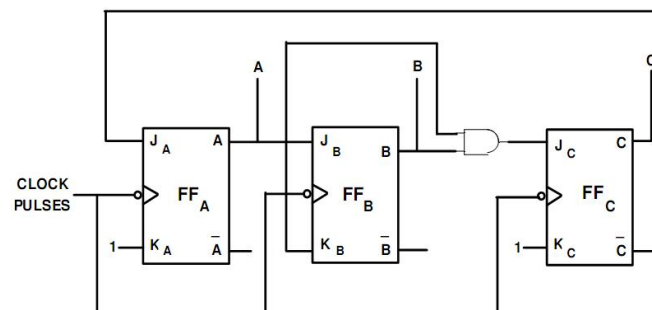**Table (b) Karnaugh Maps for MOD-5 Synchronous Counter**

Fig. (b) Logic Diagram of MOD-5 Synchronous Counter

3. i).Design a synchronous counter with states 0, 1, 2, 3, 0, 1,.... Using JK flip flop.


ii).Construct a JK FF using a D FF, a 2:1 Multiplexer and an inverter.

4. i).Design and explain the working of an 4-bit Up/Down Parallel counter.
ii).Design and working of a synchronous MOD- 6 counter using JK FF

5. Design a sequence detector which detects the sequence 01110 using D flip flop

6. (i).Explain the operation of universal shift register with neat block diagram.

4-bit Universal shift register diagram is shown below.
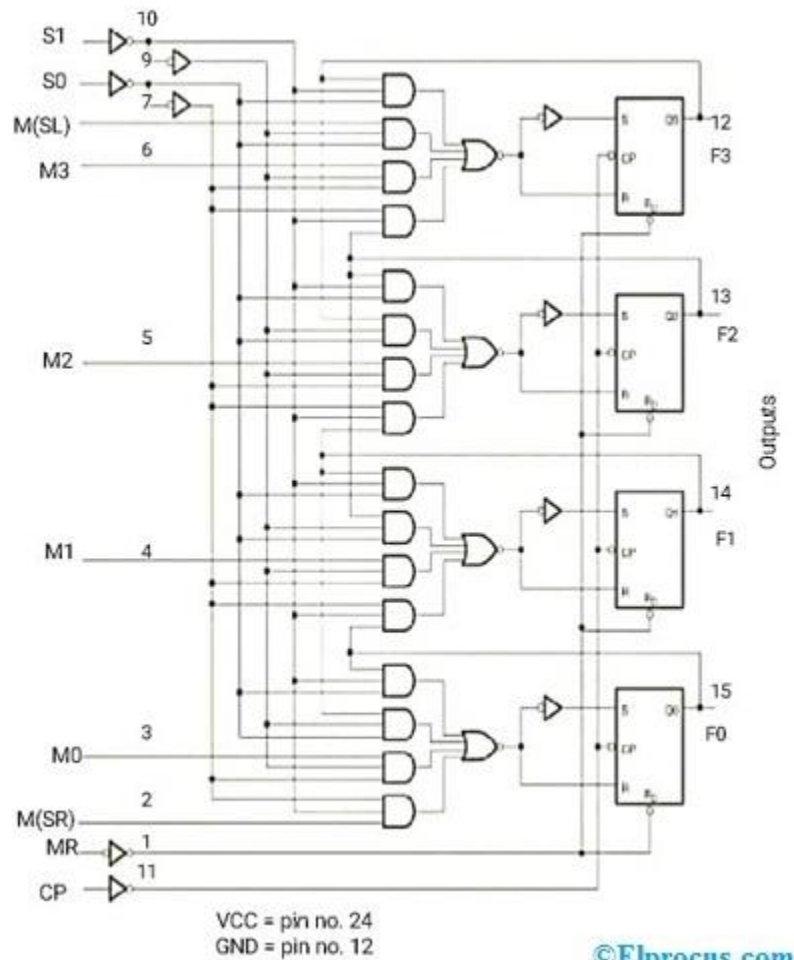
Universal Shift Register Diagram

- Serial input for shift-right control enables the data transfer towards the right and all the serial input and output lines are connected to the shift-right mode. The input is given to the AND gate-1 of the flip-flop -1 as shown in the figure via serial input pin.
- Serial input for shift-left enables the data transfer towards the left and all the serial input and output lines are connected to shift-left mode.
- In parallel data transfer, all the parallel inputs and outputs lines are associated with the parallel load.
- Clear pin clears the register and set to 0.
- CLK pin provides clock pulses to synchronize all the operations.

- In the control state, the information or data in the register would not change even though the clock pulse is applied.
- If the register operates with a parallel load and shifts the data towards the right and left, then it acts as a universal shift register.

Design of Universal Shift Register

The design of a 4-bit universal shift register using multiplexers and flip-flops is shown below.
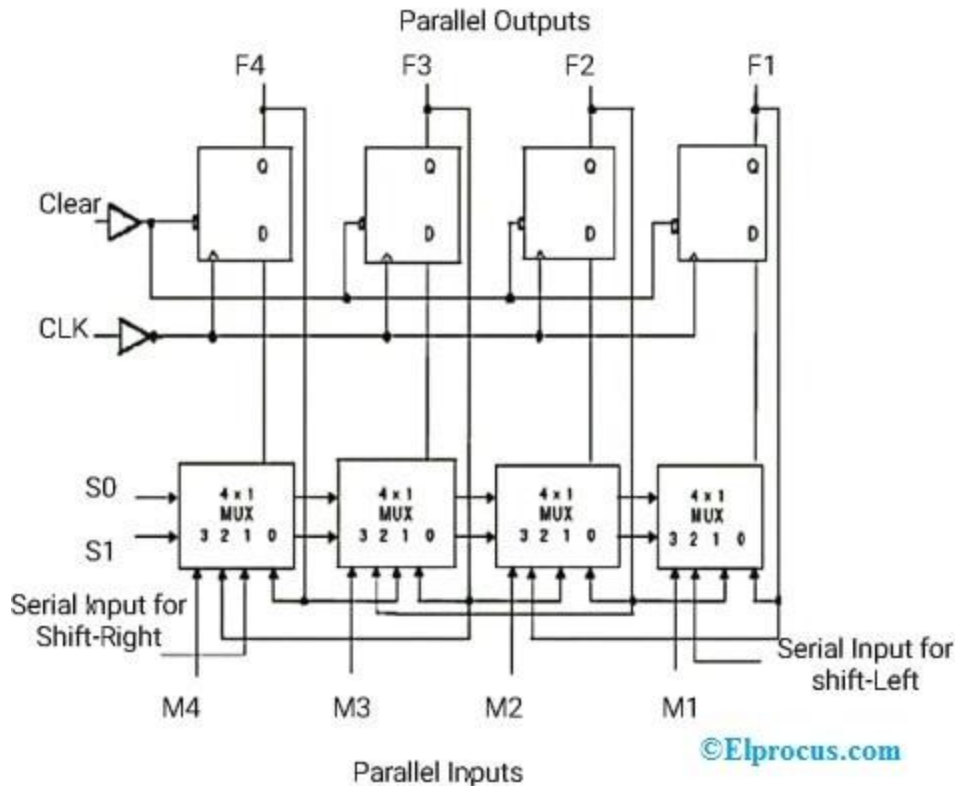
Universal Shift Register

Design

- S0 and S1 are the selected pins that are used to select the mode of operation of this register. It may be shift left operation or shift right operation or parallel mode.
- Pin-0 of first 4×1 Mux is fed to the output pin of the first flip-flop. Observe the connections as shown in the figure.
- Pin-1 of the first 4X1 MUX is connected to serial input for shift right. In this mode, the register shifts the data towards the right.
- Similarly, pin-2 of 4X1 MUX is connected to the serial input for shift-left. In this mode, the universal shift register shifts the data towards the left.
- M1 is the parallel input data given to the pin-3 of the first 4×1 MUX to provide parallel mode operation and stores the data into the register.

- Similarly, remaining individual parallel input data bits are given to the pin-3 of related 4X1MUX to provide parallel loading.
- F1, F2, F3, and F4 are the parallel outputs of Flip-flops, which are associated with the 4×1 MUX.

Universal Shift Register Working

- From the above figure, selected pins the mode of operation of the universal shift register. Serial input shifts the data towards the right and left and stores the data within the register.
- Clear pin and CLK pin are connected to the flip-flop.
- M0, M1, M2, M3 are the parallel inputs while F0, F1, F2, F3 are the parallel outputs of flip-flops
- When the input pin is active HIGH, then the universal shift register loads / retrieve the data in parallel. In this case, the input pin is directly connected to 4×1 MUX
- When the input pin (mode) is active LOW, then the universal shift register shifts the data. In this case, the input pin is connected to 4×1 MUX via NOT gate.
- When the input pin (mode) is connected to GND (Ground), then the universal shift register acts as a Bi-directional shift register.
- To perform the shift-right operation, the input pin is fed to the 1st AND gate of the 1st flip-flop via serial input for shit-right.
- To perform the shift-left operation, the input pin is fed to the 8th AND gate of the last flip-flop via input M.
- If the selected pins S0= 0 and S1 = 0, then this register doesn't operate in any mode. That means it will be in a Locked state or no change state even though the clock pulses are applied.
- If the selected pins S0 = 0 and S1 = 1, then this register transfers or shifts the data to left and stores the data.
- If the selected pins S0 = 1 and S1 = 0, then this register shifts the data to right and hence performs the shift-right operation.
- If the selected pins S0 = 1 and S1 = 1, then this register loads the data in parallel. Hence it performs the parallel loading operation and stores the data.

S0	S1	Mode of Op
0	0	Locked state (N
0	1	Shift-L

|			Shift-Ri
1	0		
1	1		Parallel Lo

From the above table, we can observe that this register operates in all modes with serial/parallel inputs using 4×1 multiplexers and flip-flops.

(ii). Explain the working Master/Slave JK FF

The master-slave flip flop is constructed by combining two JK flip flops. These flip flops are connected in a series configuration. In these two flip flops, the 1st flip flop work as "master", called the master flip flop, and the 2nd work as a "slave", called slave flip flop. The master-slave flip flop is designed in such a way that the output of the "master" flip flop is passed to both the inputs of the "slave" flip flop. The output of the "slave" flip flop is passed to inputs of the master flip flop.

In "master-slave flip flop", apart from these two flip flops, an inverter or NOT gate is also used. For passing the inverted clock pulse to the "slave" flip flop, the inverter is connected to the clock's pulse. In simple words, when CP set to false for "master", then CP is set to true for "slave", and when CP set to true for "master", then CP is set to false for "slave".

Master-slave JK Flip-Flop
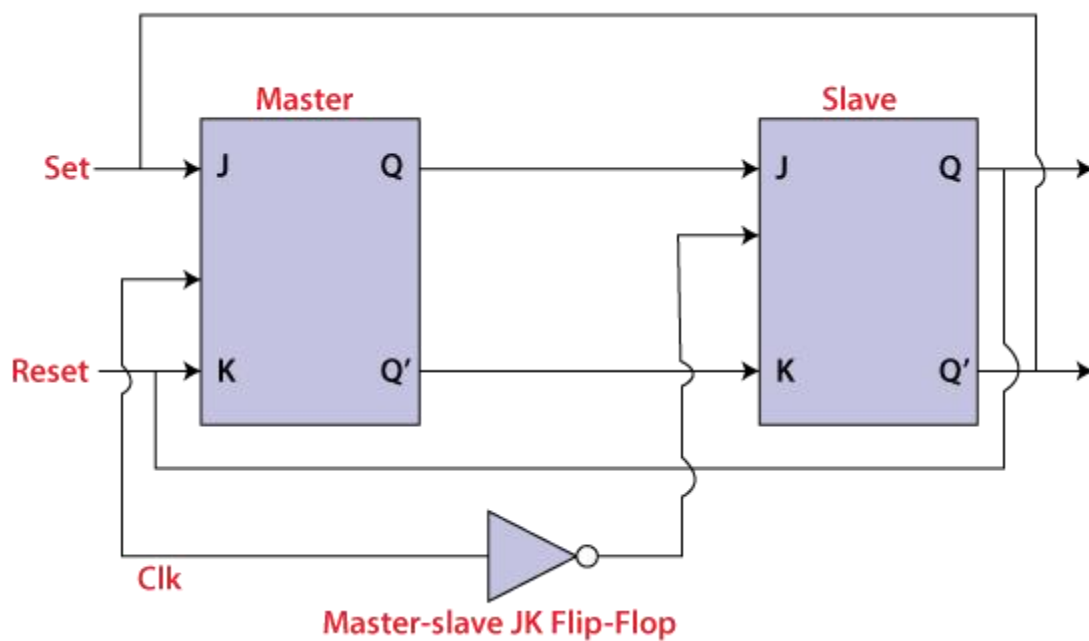
Working:

- When the clock pulse is true, the slave flip flop will be in the isolated state, and the system's state may be affected by the J and K inputs. The "slave" remains isolated until the CP is 1. When the CP set to 0, the master flip-flop passes the information to the slave flip flop to obtain the output.

- The master flip flop responds first from the slave because the master flip flop is the positive level trigger, and the slave flip flop is the negative level trigger.

- The output Q'=1 of the master flip flop is passed to the slave flip flop as an input K when the input J set to 0 and K set to 1. The clock forces the slave flip flop to work as reset, and then the slave copies the master flip flop.

- When J=1, and K=0, the output Q=1 is passed to the J input of the slave. The clock's negative transition sets the slave and copies the master.

- The master flip flop toggles on the clock's positive transition when the inputs J and K set to 1. At that time, the slave flip flop toggles on the clock's negative transition.

- The flip flop will be disabled, and Q remains unchanged when both the inputs of the JK flip flop set to 0.

Timing Diagram of a Master Flip Flop:
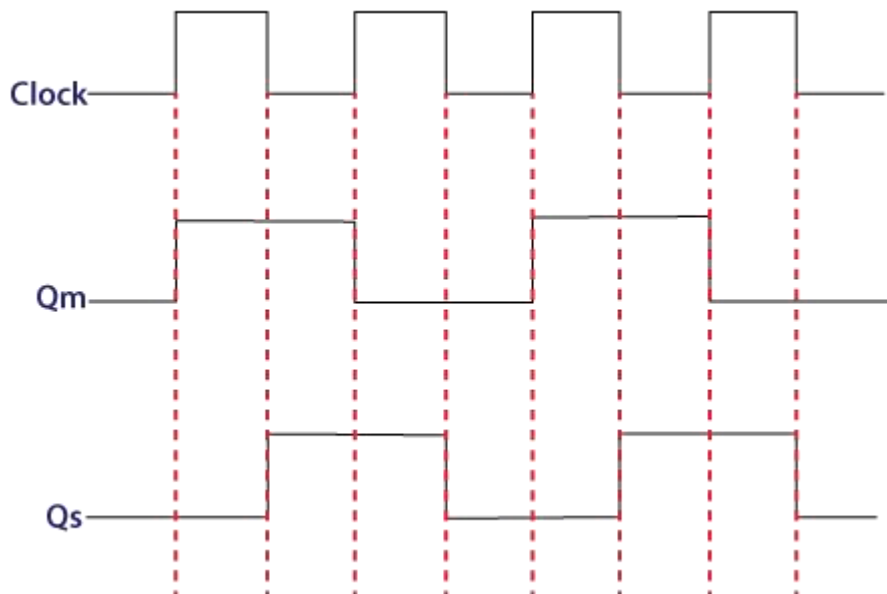
- When the clock pulse set to 1, the output of the master flip flop will be one until the clock input remains 0.

- When the clock pulse becomes high again, then the master's output is 0, which will be set to 1 when the clock becomes one again.

- The master flip flop is operational when the clock pulse is 1. The slave's output remains 0 until the clock is not set to 0 because the slave flip flop is not operational.

- The slave flip flop is operational when the clock pulse is 0. The output of the master remains one until the clock is not set to 0 again.

- Toggling occurs during the entire process because the output changes once in the cycle.

7. i). Draw the logic diagram for a 5- bit serial load shift register using D FF & explain.

A 5-bit serial-load shift-right register which has D flip-flops having Label inputs data, CLK, and CLR. Label outputs A, B, C, D and E.
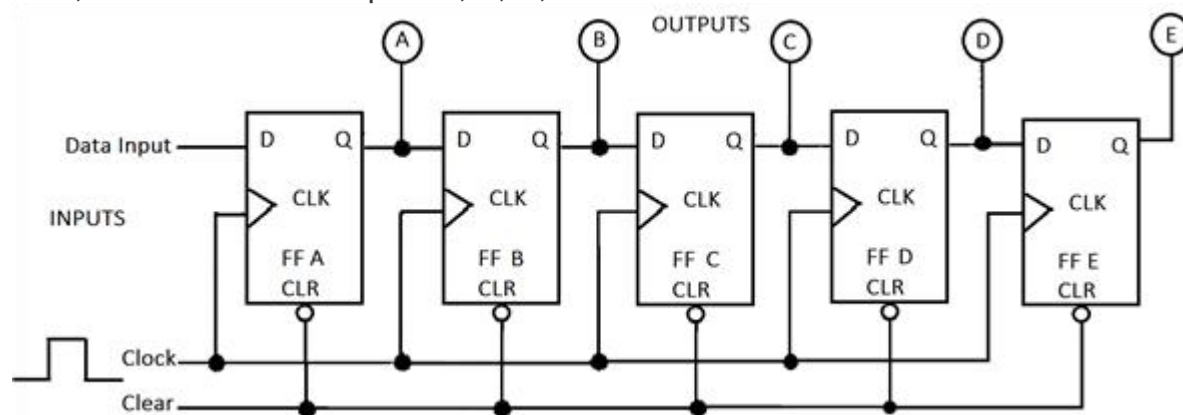
ii). Write notes on state minimization

State minimization is the transformation of a given machine into an equivalent machine with no redundant states.

Two states, si and sj of machine M are distinguishable if and only if there exists a

nite input sequence which when applied to M causes di
erent output sequences depending on whether M started in si or sj . Such a sequence is called a
distinguishing sequence for (si; sj). If there exists a distinguishing sequence of length k for (si; sj), they
are said to be k- distinguishable.

| PS | NS, z | |
	$x = 0$	$x = 1$
A	E, 0	D, 1
B	F, 0	D, 0
C	E, 0	B, 1
D	F, 0	B, 0
E	C, 0	F, 1
F	B, 0	C, 0

Outline of state minimization procedure: All states equivalent to each other form an equivalence class.
All the states in an equivalence class may be combined into one state in the reduced (quotient) machine.
These equivalence classes form a partition of the set of states. Start with all states in a partition of a
single block. Iteratively refine this partition by separating the 1- distinguishable states, 2-distinguishable
states and so on. In general, when obtaining Pk+1 from Pk, place in the same block of Pk+1 the states
that are (k+1)- equivalent, and in different blocks states that are (k + 1)-distinguishable.

PART IV

UNIVERSITY EXAM QUESTION PATTERN – 6 MARK (EACH QUESTION

1. How the memories are classified.

Main memory can be generally classified into random-access memory
(RAM) and read-only memory (ROM). It is a volatile memory. ... Secondary
Memory– Secondary memory is also frequently known as auxiliary memory.
The control unit can't directly communicate with the secondary memory.

2. What is an EPROM?

An EPROM, or erasable programmable read-only memory, is a type of
programmable read-only memory chip that retains its data when its power
supply is switched off. Computer memory that can retrieve stored data after
a power supply has been turned off and back on is called non-volatile.

3. Compare and contrast static RAM and dynamic RAM

. What is the difference between static RAM and DRAM? Static RAM is fast and expensive, and dynamic RAM is less expensive and slower. Therefore static RAM is used to create the CPU's speed-sensitive cache, while dynamic RAM forms the larger system RAM space.

4.What is PLD? List their types.

There are three kinds of PLDs based on the type of arrays, which has programmable feature.

- Programmable Read Only Memory.
- Programmable Array Logic.
- Programmable Logic Array.

5.Explain write operation with an example.

The stage configures a connection to an external data source and insert records into a table. The classicfedwrite operation determines how the records of a data set are inserted into the table.

6.. Distinguish between PAL and PLA.

The distinction between PLA and PAL is that, PAL have programmable AND array and fixed or array. On the other hand, PLA have programmable AND array and programming OR array. ... PLA stands for Programmable Logic Array. While PAL stands for Programmable Array Logic.

7. Which memory is called volatile? Why?

RAM (Random Access Memory) is called volatile memory, because in RAM memory gets erased on turning off the power. Computer has two types of memory, RAM & ROM (Read only Memory). The operations performed by the CPU, use temporary & fast memory which is RAM

8.. Write the advantages of EPROM over a PROM.

To write associate EPROM, its storage cells should stay within the same initial state. EPROM provides reduced storage permanency as compared

to PROM as a result of the EPROM is receptive to radiation and electrical noise. in the construction of EPROM, MOS transistors are used.

9.Compare the features of PROM, PAL and PLA

PROM: A PROM has a fixed AND array but a programmable OR array. When AND array is fixed that means we can only get certain min-terms. ... PAL: Here OR array is fixed and AND array is programmable. So here the min terms can be made as we wish, The hardware is less sophisticated than PLA.

10. What is access time and cycle time of a memory?

Cycle time is the time, usually measured in nanosecond s, between the start of one random access memory (RAM) access to the time when the next access can be started. Access time is sometimes used as a synonym (although IBM deprecates it). ... In manufacturing, cycle-time is the total time it takes to produce an order.

11. What is PLA? How does it differ from PAL and GAL?

The main difference among these two is that PAL can be designed with a collection of AND gates and fixed collection of OR gates whereas PLA can be designed with a programmable array of AND although a fixed collection of OR gate.

12. What is memory decoding?

Explanation: The Memory IC used in a digital system is selected or enabled only for the range of addresses assigned to it and this process is called memory decoding. It decodes the memory to be selected for a specific address.

14. Write the advantages of E2PROM over an EPROM

EEPROMs are designed to be programmed and erased within the hardware it was designed into. An EPROM requires a UV light to erase (if not a one time programmable) and an external programmer for programming. ... An EEPROM makes it easy to develop with in that you don't have to continually erase and program chips each time.

16. Draw the static and dynamic RAM cells

Dynamic RAM is the most common type of memory in use today. Inside a DRAM chip, each memory cell holds one bit of information and is made up of two parts: a transistor and a capacitor. These are, of course, extremely small transistors and capacitors so that millions of them can fit on a single memory chip. The capacitor holds the bit of information — a 0 or a 1 (see How Bits and Bytes Work for information on bits). The transistor acts as a switch that lets the control circuitry on the memory chip read the capacitor or change its state. Dynamic RAM is the most common type of memory in use today. Inside a DRAM chip, each memory cell holds one bit of information and is made up of two parts: a transistor and a capacitor. These are, of course, extremely small transistors and capacitors so that millions of them can fit on a single memory chip. The capacitor holds the bit of information — a 0 or a 1 (see How Bits and Bytes Work for information on bits). The transistor acts as a switch that lets the control circuitry on the memory chip read the capacitor or change its state.

18. What is meant by memory expansion? Mention its limit

EMS – Expanded Memory Specification

Expanded Memory Specification (EMS), or expanded memory, is a technique for utilizing more than 1MB of main memory in DOS -based computers. The limit of 1MB is built into the DOS operating system.

41. Differentiate synchronous and asynchronous sequential circuits?

Sr. No.	Key	Synchronous Sequential Circuits	Asynchronous Sequential Circuits
1	Definition	Synchronous sequential circuits are digital sequential circuits in which the feedback to the input for next output generation is governed by clock signals.	On other hand Asynchronous sequential circuits are digital sequential circuits in which the feedback to the input for next output generation is not governed by

			clock signals.
2	Memory Unit	In Synchronous sequential circuits, the memory unit which is being get used for governance is clocked flip flop.	On other hand unclocked flip flop or time delay is used as memory element in case of Asynchronous sequential circuits.
3	State	The states of Synchronous sequential circuits are always predictable and thus reliable.	On other hand there are chances for the Asynchronous circuits to enter into a wrong state because of the time difference between the arrivals of inputs. This is called as race condition.
4	Complexity	It is easy to design Synchronous sequential circuits	However on other hand the presence of feedback among logic gates causes instability issues making the design of Asynchronous sequential circuits difficult.
5	Performance	Due to the propagation delay of clock signal in reaching all	Since there is no clock signal delay, these are fast compared to the Synchronous

		elements of the circuit the Synchronous sequential circuits are slower in its operation speed	Sequential Circuits
6	Example	Synchronous circuits are used in counters, shift registers, memory units.	On other hand Asynchronous circuits are used in low power and high speed operations such as simple microprocessors, digital signal processing units and in communication systems for email applications, internet access and networking.

42. What are the two types of Asynchronous sequential circuits?

- Fundamental Mode. Only One input can be change at a time after stable state. This mode is widely used for design.
- Pulse mode: - More than one input can be change at a time after stable state.

43. Define flow table and primitive flow table

A primitive flow table is a flow table with only one stable total state in each row. The total state consists of the internal state combined with the input. The resulting primitive table for the gated latch is shown below: First, we fill in one square in each row belonging to the stable state in that row.

44. Define state table and state assignment.

State Tables and State Diagrams. State Tables and State Diagrams. The relationship that exists among the inputs, outputs, present states and next states can be specified by either the state table or the state diagram. State Table.

45. Differentiate stable and unstable state

A system is said to be in stable equilibrium if, when displaced from equilibrium, it experiences a net force or torque in a direction opposite to the direction of the displacement. ... A system in unstable equilibrium accelerates away from its equilibrium position if displaced even slightly.

46. Draw block diagram for Moore and Mealy model.

The state diagram of the above Mealy Machine is −
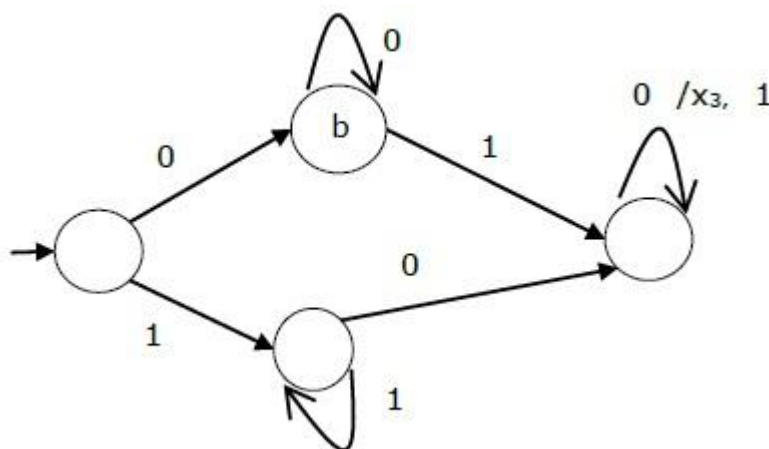
Moore Machine
Moore machine is an FSM whose outputs depend on only the present state.

A Moore machine can be described by a 6 tuple $(Q, \Sigma, O, \delta, X, q_0)$ where −

- Q is a finite set of states.
- Σ is a finite set of symbols called the input alphabet.
- O is a finite set of symbols called the output alphabet.
- δ is the input transition function where $\delta: Q \times \Sigma \to Q$
- X is the output transition function where $X: Q \to O$
- q_0 is the initial state from where any input is processed ($q_0 \in Q$).

The state table of a Moore Machine is shown below −

Present state	Next State		Output
	Input = 0	Input = 1	
→ a	b	c	x_2
b	b	d	x_1
c	c	d	x_2
d	d	d	x_3

The state diagram of the above Moore Machine is −
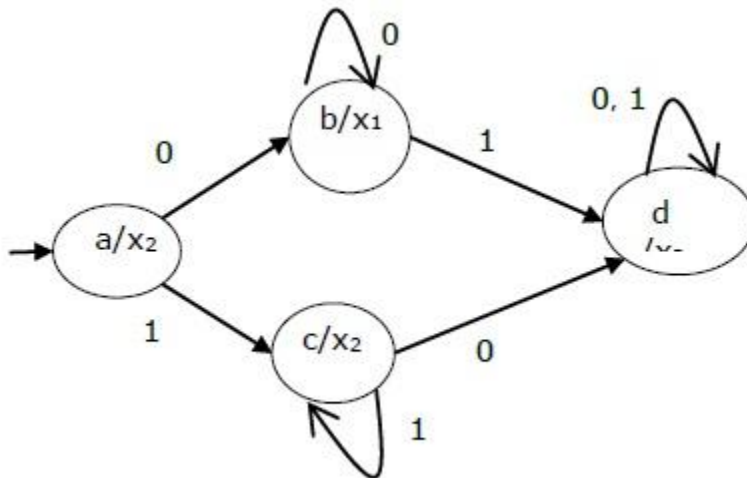
Mealy Machine vs. Moore Machine
The following table highlights the points that differentiate a Mealy Machine from a Moore Machine.

Mealy Machine	Moore Machine
Output depends both upon the present state and the present input	Output depends only upon the present state.
Generally, it has fewer states than Moore Machine.	Generally, it has more states than Mealy Machine.
The value of the output function is a function of the transitions and the changes, when the input logic on the present state is done.	The value of the output function is a function of the current state and the changes at the clock edges, whenever state changes occur.
Mealy machines react faster to inputs. They generally react in the same clock cycle.	In Moore machines, more logic is required to decode the outputs resulting in more circuit delays. They generally react one clock cycle later.

47. Define the terms race and critical race.

Critical race theory is an intellectual movement and a framework of legal analysis according to which (1) race is a culturally invented category used to oppress people of colour and (2) the law and legal institutions in the United States are inherently racist insofar as they function to create and maintain social, …

48. What is a state diagram? Give an example

A state diagram shows the behavior of classes in response to external stimuli. Specifically a state diagram describes the behavior of a single object in response to a series of events in a system. Sometimes it's also known as a Harel state chart or a state machine diagram.

49. What are Hazards? How it can be avoided?

When we refer to hazards in relation to occupational safety and health the most commonly used definition is 'A Hazard is a potential source of harm or adverse health effect on a person or persons'. The terms Hazard and Risk are often used interchangeably but this simple example explains the difference between the two.

50. Compare the ASM chart with a conventional flow chart.

ASM is an algorithm consists of a few steps, which is used to simplify a sequential digital system. An ASM chart is resembles a conventional flow chart but the difference is, a conventional flow chart does not have timing relationships but the ASM takes timing relationship into account.