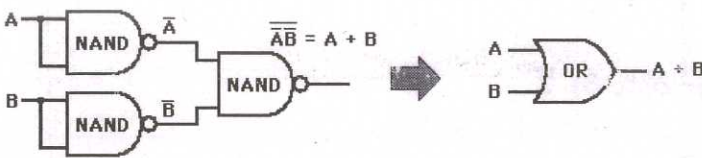
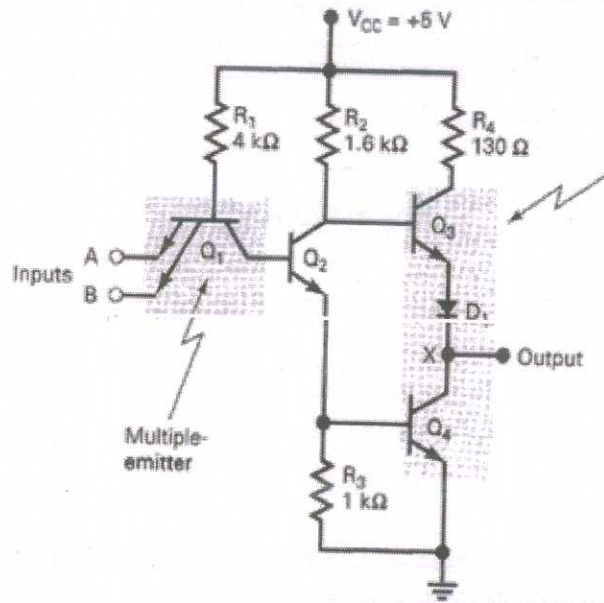


Scoring Indicators

Code : TED (15) 3042

Version:

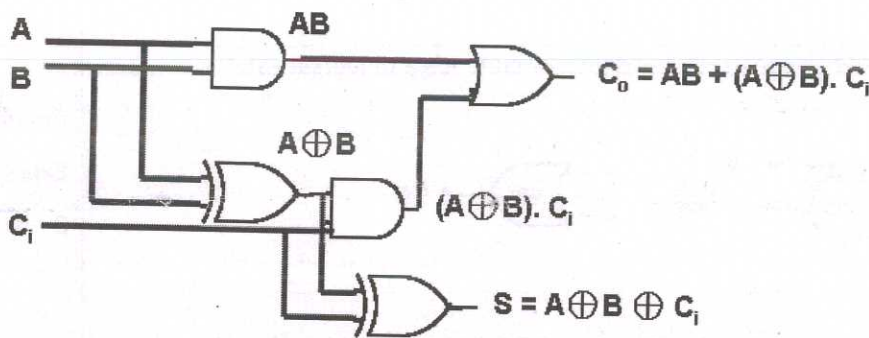
Qn. No.	Scoring Indicators	Split score	Total
I.	<u>PART A</u>		
1.	<ol style="list-style-type: none"> 1. Grey Code 2. The compliment of the product of two or more variables is equal to the sum of the individual variable or vice-versa. 3. Fan-out is the maximum number of logic gates which can be driven by logic gate output without affecting its performance. 4. MSFFs are used to overcome the race around problem in JKFF. 5. 5 flip flops. 'n' flip flops will have n bits and can count $2^n - 1$. 	2 marks each	10
II	<u>PART B</u>		
1.	<ol style="list-style-type: none"> (a) 1011010110 (b) 15B (c) 455 	Answer with calculation 2 marks each.	6
2.	<p>NAND gate can be used to construct any digital circuit. It's a universal gate.</p> 	Circuit-4 Expressions- 2 marks	6
3.	<p>TTL inverter:</p> <p>If A or B is low, the base-emitter junction of Q1 is forward biased and its base-collector junction is reverse biased. Then there is a current from Vcc through R1 to the base emitter junction of Q1 and into the LOW input, which provides a path to the ground for the current. Hence there is no current into the base of Q2 and making it into cur-off. The collector of Q2 is HIGH and turns Q3 into saturation.</p>	Circuit:4 Explain:2	6



When A and B are high, the two input base emitter junctions of Q1 are reverse biased and its base collector junction is forward biased. This permits current through R1 and the base collector junction of Q1 into the base of Q2, thus driving Q2 into saturation. As a result Q4 is turned ON by Q2, and producing LOW output which is near ground potential. At the same time, the collector of Q2 is sufficiently at LOW voltage level to keep Q3 OFF.

4

A full adder is a combinational circuit that forms the arithmetic sum of input; it consists of three inputs and two outputs. A full adder is useful to add three bits at a time but a half adder cannot do so. In full adder sum output will be taken from X-OR Gate, carry output will be taken from OR Gate.



Circuit -2

Expression-2

Explanation-2

6

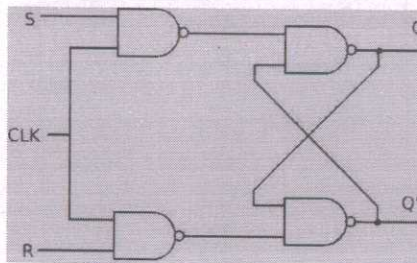
SI No.	Asynchronous Sequential Circuit	Synchronous Sequential Circuit
1	Memory elements are unclocked flip flops	Memory elements are clocked flip flops
2	Change in input signal can effect memory element at any instant of time	Change in input signal can effect memory element upon activation of clock signal
3	Difficult to design	Easier to design
4	Faster than synchronous	slower

Each parameter :2 marks each 2X3=6

6

SR flip flop:

An SR flipflop using NAND gates can be constructed as given below.



In the above circuit, we can see the a '0' at the clock input will make the output Q to 'No Change' condition. Because the output of the first level NAND gates will be 1 which will make the latch output to continue in its previous condition. The output $Q(t+1)$ stands for the future output at time $t+1$. And $Q(t)$ means present output at time 't'. The complimentary output Q' is not shown in the truth table. We know that Q' will be always compliment to Q.

Clk	S	R	$Q(t+1)$
0	x	x	$Q(t)$ No Change
1	0	0	$Q(t)$ No Change
1	0	1	0 Reset
1	1	0	1 Set
1	1	1	Prohibited

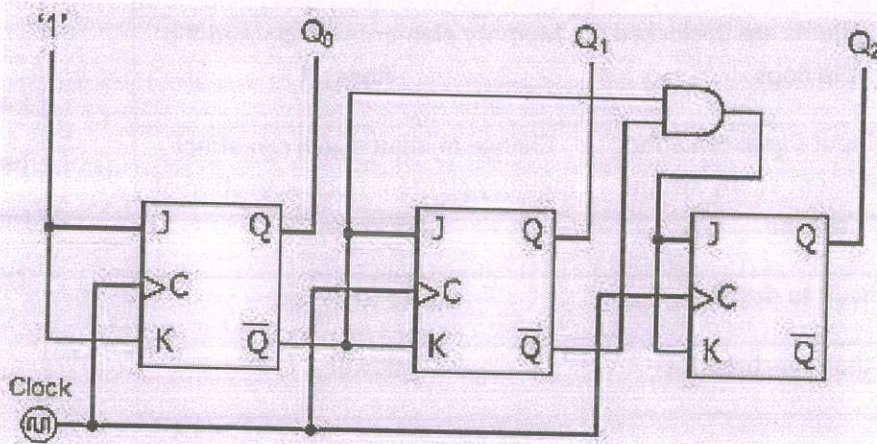
Explain:2
Circuit :2
Truth table :2

6

7

Mod-8 synchronous counter

The circuit diagram for a Mod-8 synchronous counter is as shown below.



- In synchronous counters all the flip-flops will have the same clock.
- A Mod-8 synchronous counter will have 8 states counting from 0 to 7.
- Three bits are required to represent 7.
- Thus the counting is from 000 to 111.
- Each J and K inputs are shorted together to make T flip-flop.

The truth table is as shown below.

Clock	Q3	Q2	Q1
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1
9	0	0	0

Explanation:3

Circuit:3

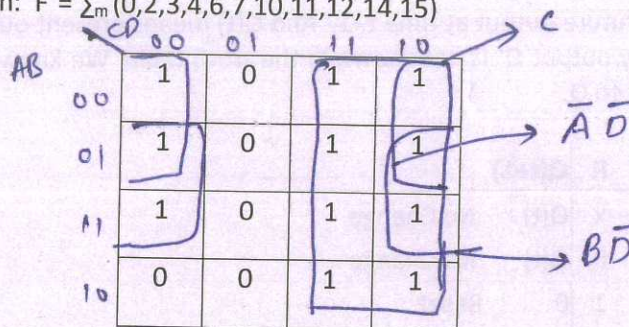
6

PART C

III

K Map Minimization: $F = \sum_m(0,2,3,4,6,7,10,11,12,14,15)$

(a)



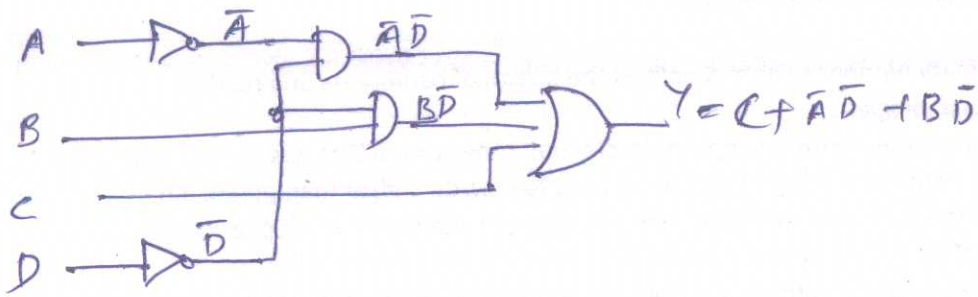
$$Y = C + \bar{A}\bar{D} + B\bar{D}$$

Mapping

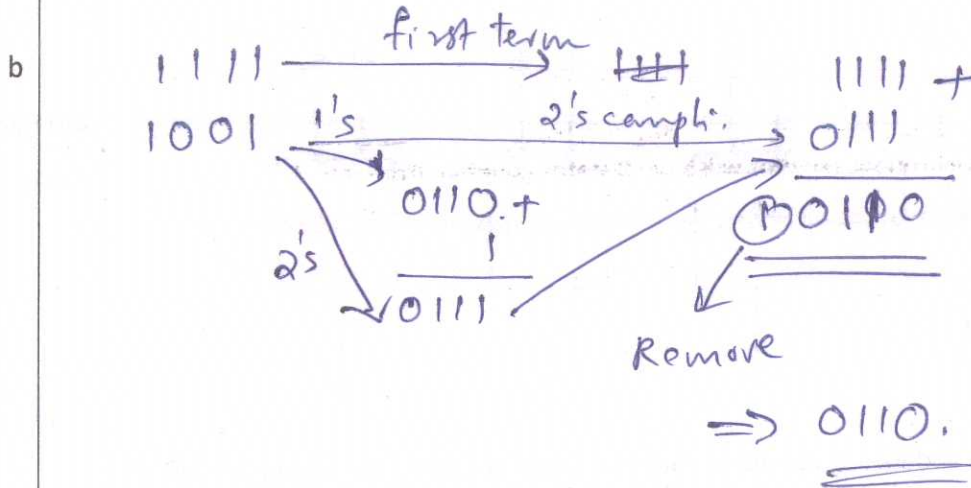
+Expression: 6

11

Circuit:4



Subtract 1001 from 1111 by using 2's complement method.



Circuit for the function through SOP form.

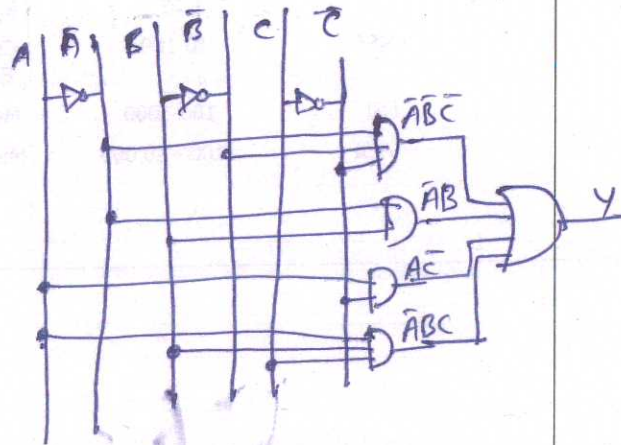
$$F_{(ABC)} = \sum_m(0,2,3,4,6,7)$$

IV
(a)

No	A	B	C	Y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

$\rightarrow \bar{A}\bar{B}\bar{C}$
 $\rightarrow \bar{A}B\bar{C}$
 $\rightarrow \bar{A}BC$
 $\rightarrow A\bar{B}\bar{C}$
 $\rightarrow ABC$

$$\begin{aligned}
 Y &= \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}BC \\
 &\quad + A\bar{B}\bar{C} + ABC \\
 &\quad + ABC \\
 Y &= \bar{A}\bar{B}\bar{C} + \bar{A}B + A\bar{C} + ABC
 \end{aligned}$$



Truth Table: 3

SOP: 3

Circuit : 3

Need for Binary number system:

b

- Binary number system simplifies the design of computers and related technologies.
- Binary is extremely simple to implement. Any system that has an "on" and "off" or "high" and "low" state can be used to encode and/or manipulate data.
- Binary is the lowest "base" possible (base 2) and hence any higher counting system can be easily encoded (e.g. decimal, octal, hexadecimal, etc.)
- Binary data is extremely robust in transmission because of less noise.

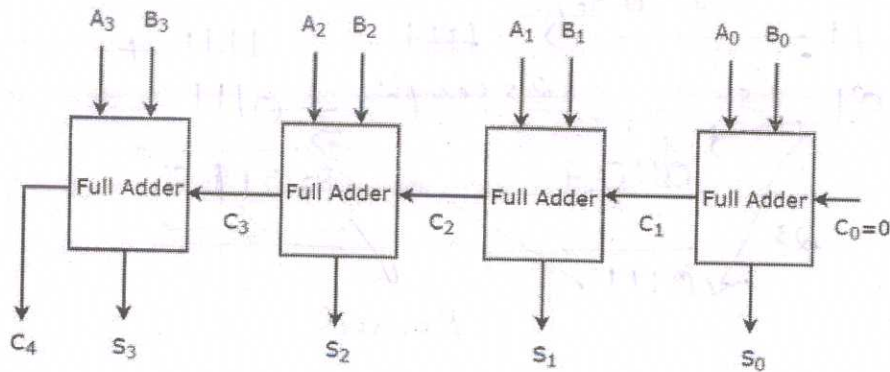
Explain : 6

6

4 bit parallel binary adder:

V

a



Circuit: 4

Explain:5

9

Here, the 4 Full adders are cascaded. Each Full adder is getting the respective bits of two parallel inputs A & B. The carry output of one Full adder will be the carry input of subsequent higher order Full adder. This 4-bit binary adder produces the resultant sum having at most 5 bits. So, carry out of last stage Full adder will be the MSB.

In this way, we can implement any higher order binary adder just by cascading the required number of Full adders. This binary adder is also called as ripple carry (binary) adder because the carry propagates (ripples) from one stage to the next stage.

b

Level of Integration	Number of Gates	Typical Applications
SSI	1-10	Basic Gates
MSI	10-100	Counters, Decoder, Encoder
LSI	100-1000	Memories, ADC/DAC
VLSI	1000-10,000	MPU's

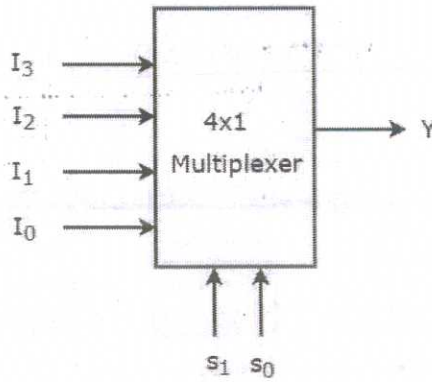
1.5 for each scale

1.5X4 =6

6

VI
a

4x1 Multiplexer has four data inputs I_3, I_2, I_1 & I_0 , two selection lines s_1 & s_0 and one output Y . The block diagram of 4x1 Multiplexer is shown in the following figure.



One of these 4 inputs will be connected to the output based on the combination of inputs present at these two selection lines.

Selection Lines		Output
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

From Truth table, we can directly write the Boolean function for output, Y as

$$Y = S_1' S_0' I_0 + S_1' S_0 I_1 + S_1 S_0' I_2 + S_1 S_0 I_3$$

Explanation:3
Diagram :2
Truth table:2
Expression :2

9

b

Parameter	TTL	CMOS
Fan-in	12-14	>10
Fan-out	10	25
Basic Gate	NAND	NOR/OR
Power Dissipation in mW/gate	10	1
Noise Immunity	Very good	Very good
Propagation delay in ns	10	70

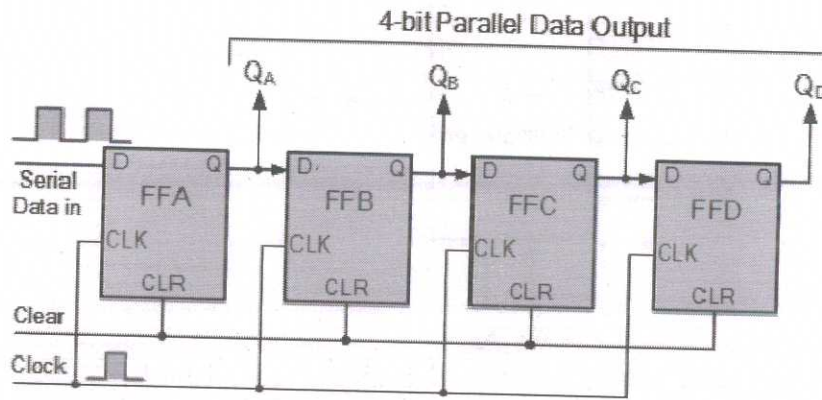
Any 3
parameter 2
marks each.
 $3 \times 2 = 6$

6

Shift Register Types:

VII
a.

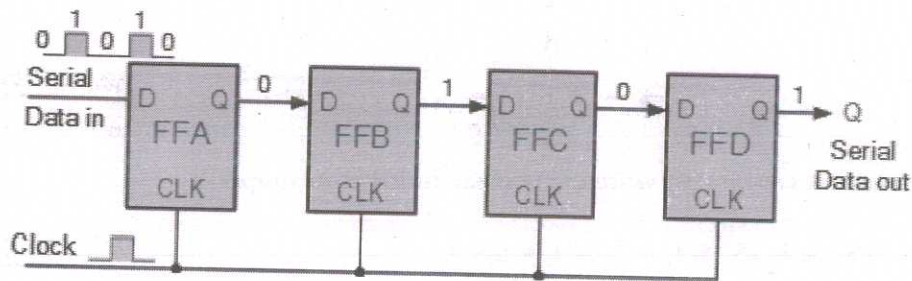
4-bit Serial-in to Parallel-out Shift Register



We can see that, the data given at the first flip flop input will shift through the subsequent flip flops with respect to the clock.

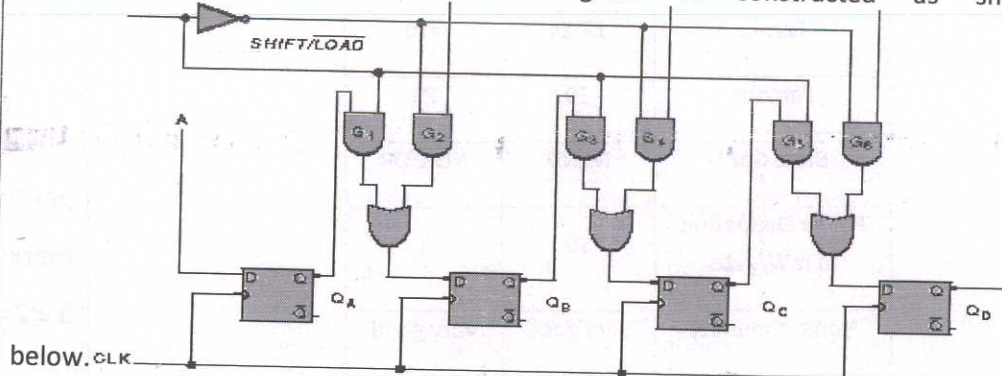
Serial-in to Serial-out (SISO) Shift Register

This shift register is very similar to the SIPO above, except were before the data was read directly in a parallel form from the outputs Q_A to Q_D .



Parallel-in Serial-out (PISO) Shift Register

The Parallel-in to Serial-out shift register is constructed as shown



- The loading and shifting operation is achieved by using a 2X1 multiplexer circuit using two AND gates and one OR gate.
- Thus while Loading, the data will load to each flip flop's input in parallel fashion.

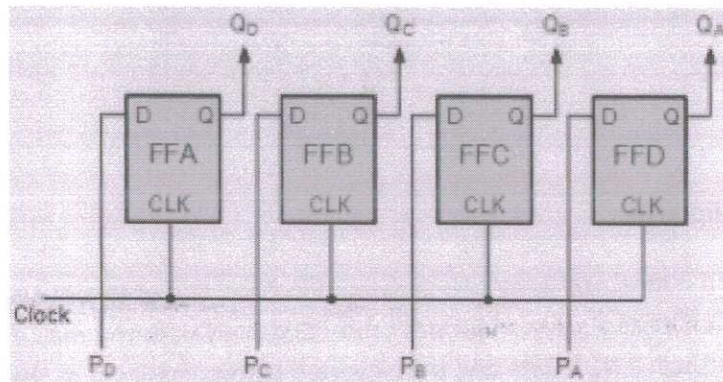
Each 4 types
diagram:2
Explanation:1
 $4 \times (2+1) = 12$

1.

- While Shifting, the data will shift from left to right serially just like in a SISO.

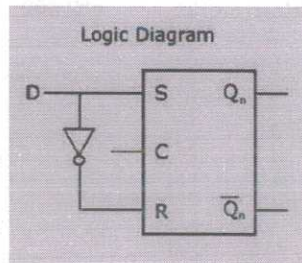
Parallel-in Parallel-out (PIPO) Shift Register

The data is presented in a parallel format to the parallel input pins P_A to P_D and then transferred together directly to their respective output pins Q_A to Q_D by the same clock pulse.



D Flip flop using SR.

b



Truth table.

Clk	D	Q(t+1)
0	x	Q(t) No Change
1	0	0
1	1	1

Diagram:2

Truth table:1

3

Ring Counter

VIII
a

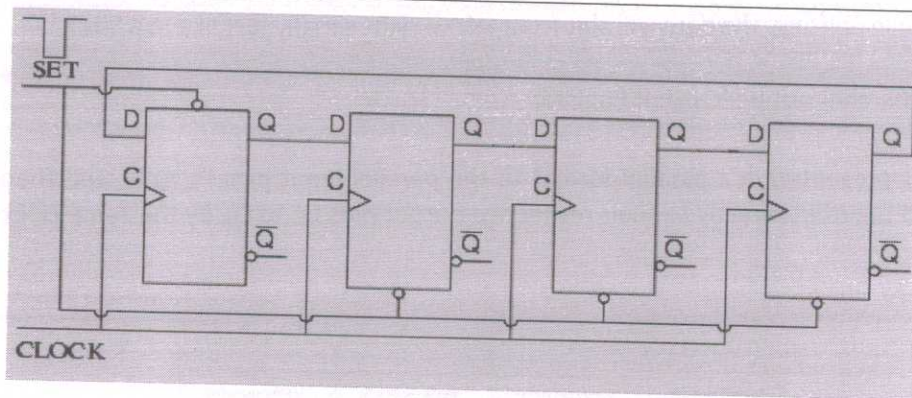
A ring counter is just like a shift register using D flip flops. The first LSB flip flop is made 1 by activating its Preset input for a small time. All other flip flops are kept 0 in the initial condition. The output of the last MSB flip flop is feedback to the D input of the first flip flop. This 1 will move in a ring manner from left flip flop to the right and coming back to the left.

So on each successive clock pulse, the counter circulates the same data bit between the four flip-flops over and over again around the "ring" every fourth clock cycle.

Diagram:4

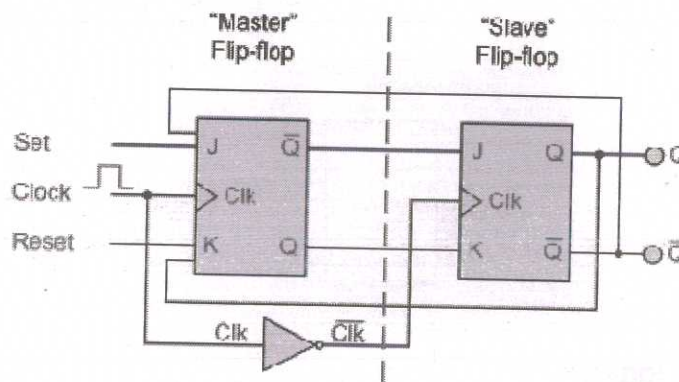
Explanation:4

8



Master Slave JK Flip flop

b Master-slave flip flop is designed using two separate flip flops. Out of these, one acts as the master and the other as a slave. The clock pulse [Clk] is given to the master J-K flip flop and it is sent through a NOT Gate and thus inverted before passing it to the slave J-K flip flop.



When Clk=1, the master J-K flip flop gets disabled. The Clk input of the master input will be the opposite of the slave input. So the master flip flop output will be recognized by the slave flip flop only when the Clk value becomes 0. Thus, the input will make an output only after both positive and negative clock cycle. Thus there will not be a continuous toggling. Thus race around condition is avoided.

Diagram:4
Explanation:3

7

IX

a

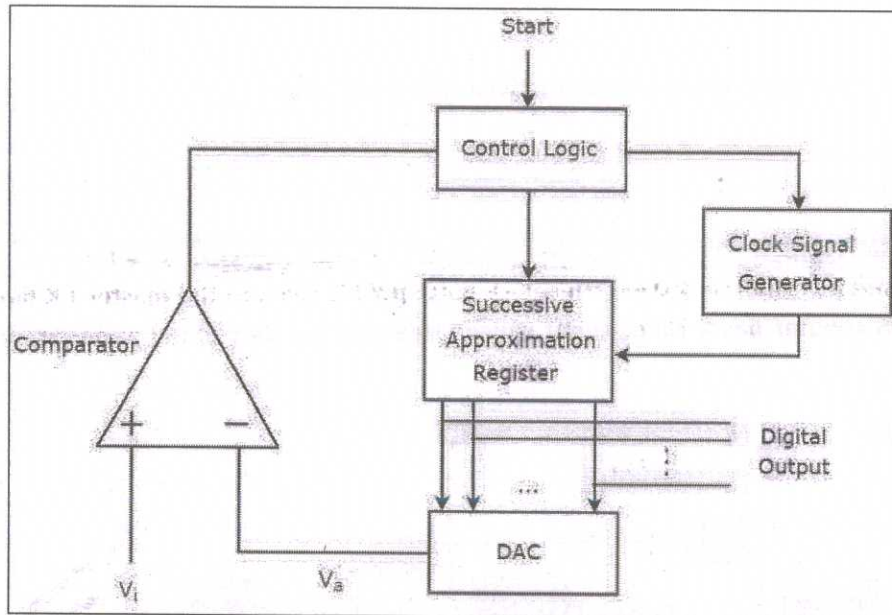
Successive Approximation type ADC

The successive approximation ADC mainly consists of 5 blocks: Clock signal generator, Successive Approximation Register (SAR), DAC, comparator and Control logic.

Diagram:5

Explanation:4

9



The working of a successive approximation ADC is as follows.

- The control logic resets all the bits of SAR and enables the clock signal generator in order to send the clock pulses to SAR.
- The binary (digital) data present in SAR will be updated for every clock pulse based on the output of comparator. The output of SAR is applied as an input of DAC.
- DAC converts the received digital input, which is the output of SAR, into an analog output. The comparator compares this analog value V_a with the external analog input value V_i .
- The output of a comparator will be '1' as long as V_i is greater than V_a . Similarly, the output of comparator will be '0', when V_i is less than or equal to V_a .
- The operations mentioned in above steps will be continued until the digital output is a valid one.
- The digital output will be a valid one, when it is almost equivalent to the corresponding external analog input value V_i .

b

Resolution:

Accuracy is the ratio between the maximum signal measured to the smallest part that can be resolved.

Each

Definition: 2
marks each.

6

Accuracy:

Accuracy can be defined as the amount of uncertainty in a measurement with respect to an absolute standard.

Or the accuracy of the A/D converter determines how close the actual digital output is to the theoretically expected digital output for a given analog input.

Settling time :

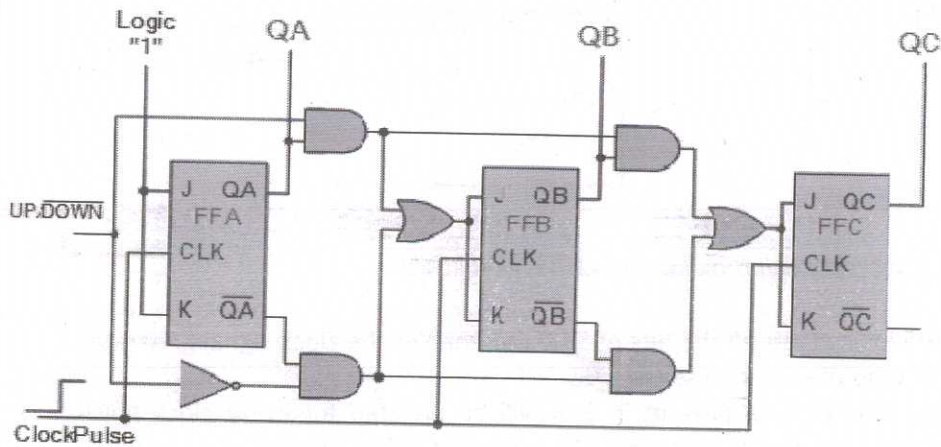
Settling time is the time taken to reach the final converted value.

3X2=6

X
a

3-bit Up/Down Counter

An Up/Down counter has a Mode input called Up/Down for Up counting and Down counting.



The circuit above is of a simple 3-bit Up/Down synchronous counter using JK flip-flops configured to operate as toggle or T-type flip-flops giving a maximum count of zero (000) to seven (111) and back to zero again. Then the 3-Bit counter advances upward in sequence (0,1,2,3,4,5,6,7) or downwards in reverse sequence (7,6,5,4,3,2,1,0).

Diagram:5

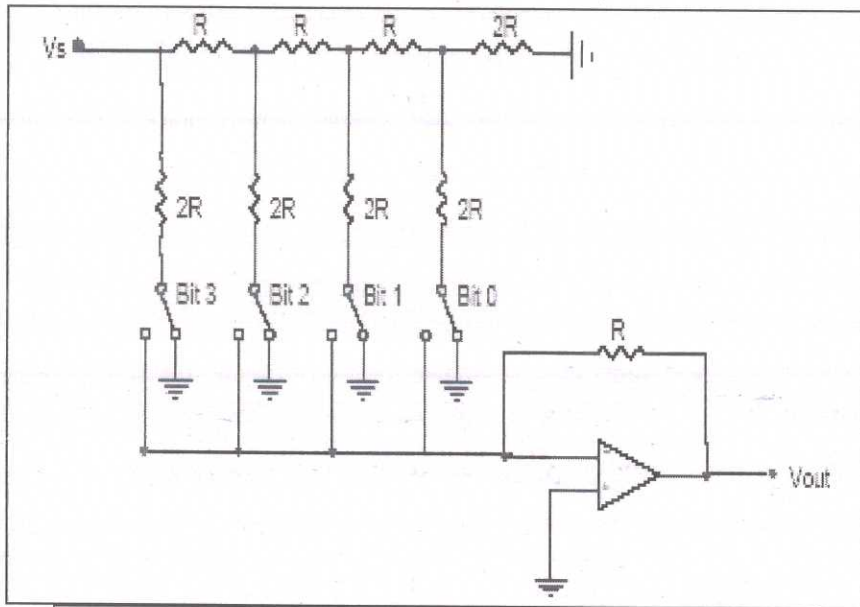
Explanation:3

8

b

R-2R ladder type DAC

The circuit diagram for a R-2R ladder type DAC is as shown below.



$$V_{out} = -R \left(b_3 \frac{V_{ref}}{2R} + b_2 \frac{V_{ref}}{4R} + b_1 \frac{V_{ref}}{8R} + b_0 \frac{V_{ref}}{16R} \right)$$

Each bit corresponds to a switch:

- If the bit is high, the corresponding switch is connected to the inverting input of the op-amp.
- If the bit is low, the corresponding switch is connected to the ground.

For general n-Bit R-2R Ladder or Binary Weighted Resistor DAC

$$V_{out} = -V_{ref} \sum_{i=1}^n b_{n-i} \frac{1}{2^i}$$

Circuit:4

Explanation:3

7
