

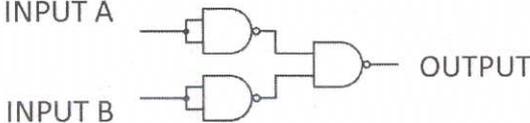
Scheme of Valuation
(Scoring Indicators)

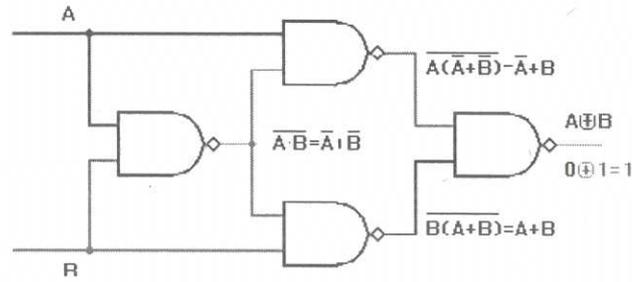
Revision: 2015

Code: TED

(15) 3133

Course: DIGITAL COMPUTER PRINCIPLES

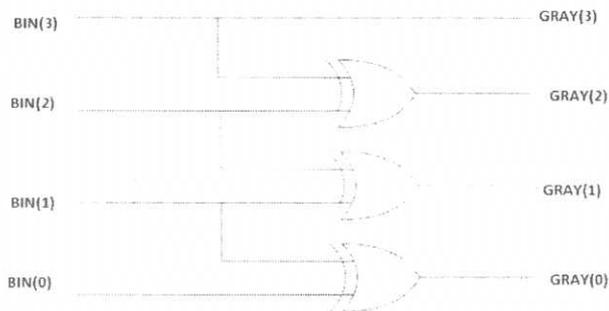
Q.No	Scoring Indicator	Split up score	Total score
I	PART A		
1	<p>2's complement of a binary number is obtained by adding 1 to the LSB of 1's complement of that number.</p> <p>∴ Binary number is $(10110010)_2$ 1's complement $\rightarrow (01001101)_2$</p> $\begin{array}{r} 1001110 \\ + 0000001 \\ \hline 01001110 \end{array}$ <p>complement of $(10110010)_2$ is $(01001110)_2$</p>	<p>1's complement -1</p> <p>2's complement -1</p>	2
2	A Karnaugh map (K-map) is a pictorial method used to minimize Boolean expressions without having to use Boolean algebra theorems and equation manipulations.	2	2
3	A flip-flop is a binary storage device capable of storing one bit of information. S-R flip-flop, D flip-flop, J-K flip-flop, T flip-flop	Definition - 1 Any 2 ex - 1	2
4	Hamming code	2	2
5	Programmable Array Logic. The PAL is a programmable logic device with a fixed OR array and a programmable AND array.	2 x 1	2
II	PART B		
1	<p style="text-align: center;">OR gate from NAND gates</p> <p>INPUT A  OUTPUT</p> <p style="text-align: center;">XOR gate from NAND gates</p>	3 x 2	6



2	$F = A + B [AC + (B+C') D]$ $= A + B [AC+ BD + C'D]$ $= A + ABC + BBD + BC'D$ $= A + ABC + BD + BC'D$ $= A + ABC + BD (1 + C')$ $= A + ABC + BD.1$ $= A (1 + BC) + BD$ $= A.1 + BD$ $= A + BD$	Steps – each 1 mark	6
3	$Y = A + B'C$ $A = A (B + B') = AB + AB'$ $AB = AB (C + C') = ABC + ABC'$ $AB' = AB' (C + C') = AB'C + AB'C'$ $B'C = B'C (A + A') = AB'C + A'B'C$ $A + BC = ABC + ABC' + AB'C + AB'C' + AB'C + A'B'C$ $= ABC + ABC' + AB'C' + AB'C + A'B'C$	6	6
4	The gray code is a binary code in which the adjacent numbers differ only in one position.	Figure -3	

4-bit binary				4-bit Gray			
B ₄	B ₃	B ₂	B ₁	G ₄	G ₃	G ₂	G ₁
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

(a) Conversion table



Truth Table
- 2

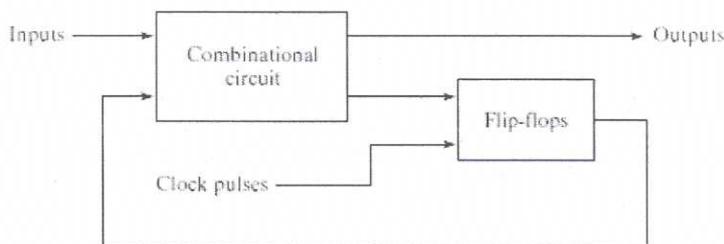
Explanation
- 1

6

5 A sequential circuit consists of a feedback path, and employs some memory elements.

Synchronous sequential circuit: outputs change only at specific time.

It has a common clock signal for controlling the changes of the state of the circuit.



(a) Block diagram



(b) Timing diagram of clock pulses

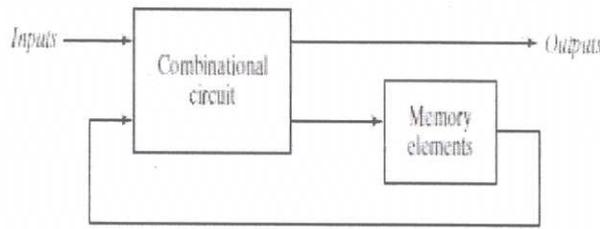
2 x 3

6

Asynchronous sequential circuit:

outputs change at any time.

That is the change is not dependent on common clock signal.



6 D flip-flop:

single input D (data)

D=HIGH a SET state

D=LOW a RESET state

Q follows D at the clock edge.

TO convert S-R flip-flop into a D flip-flop: add an inverter.

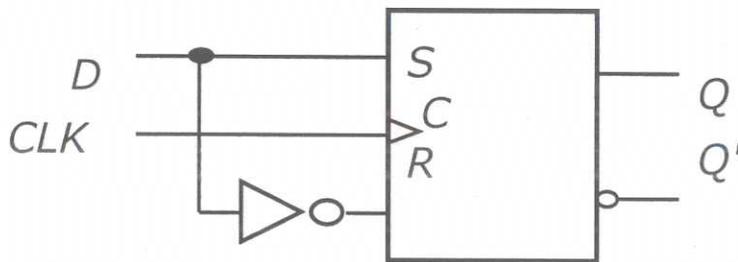


Fig - 3

A positive edge-triggered D flip-flop formed with an S-R flip-flop.

D	CLK	Q(t+1)	Comments
1	↑	1	Set
0	↑	0	Reset

↑ = clock transition LOW to HIGH

Explanation
- 3

7 CPU provides the address of the desired data

Decoding is used to select memory word specified by input address.

It is achieved with the decoding circuits.

A decoder with k inputs and 2k outputs requires 2k AND gates with k inputs per gate.

Working:

Memory chip have a pin called chip select(CS)

6 x 1

6

Control signals RD and WR are from the cpu are connected to OE(output enable) and WE(write enable) pins of memory chip

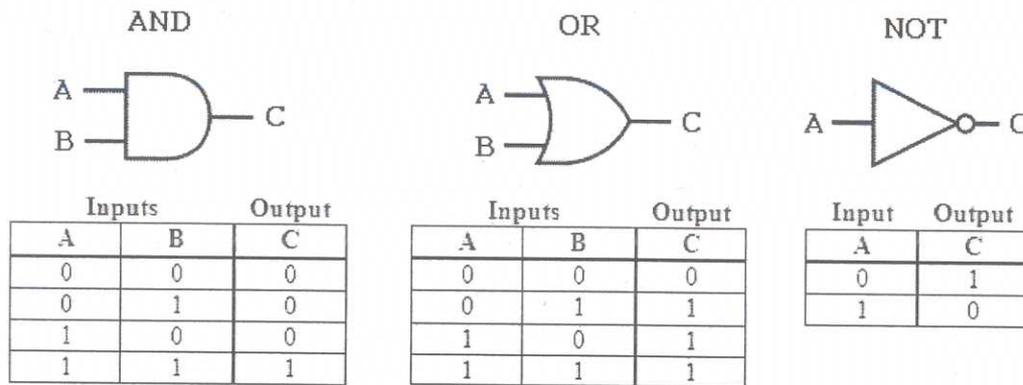
PART C

III a Logic gates are the basic building blocks of any digital system. It is an electronic circuit having one or more than one input and only one output. 3 basic logic gates are named as AND gate, OR gate, NOT gate etc.

The Logic AND Gate is a type of digital logic circuit whose output goes HIGH to a logic level 1 only when all of its inputs are HIGH

The Logic OR Gate is a type of digital logic circuit whose output goes HIGH to a logic level 1 only when one or more of its inputs are HIGH

A NOT gate, often called an inverter performs logical negation on its input.



3 x 3

9

III b

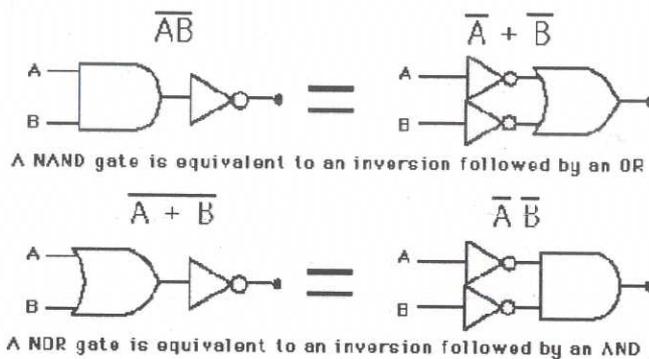


Figure -4

Theorems - 2

6

A	B	\bar{A}	\bar{B}	A+B	A·B	$\overline{A+B}$	$\bar{A} \cdot \bar{B}$	$\overline{A \cdot B}$	$\bar{A} + \bar{B}$
0	0	1	1	0	0	1	1	1	1
0	1	1	0	1	0	0	0	1	1
1	0	0	1	1	0	0	0	1	1
1	1	0	0	1	1	0	0	0	0

- Theorem 1: The complement of the product of two or more logical variables is equal to the sum of the complements of the variables
- Theorem 2: The complement of the sum of two or more variables is equal to the product of the complements of the variables.

IV
a

i) $(4F7.A8)_{16} = 0100\ 1111\ 0111.10101000_2$ (each hex digit to binary)
 $= 010\ 011\ 110\ 111.101\ 010\ 000$ (grouping of 3 bits)
 $= (2367.520)_8$

ii) $10110.0101_2 = 0001\ 0110.0101$ (grouping of 4 bits)
 $= 16.5_{16}$ (each 4-bit group to hex digit)

iii) 163.875

$ \begin{array}{r} 2 \overline{)163} \\ \underline{2 \ 81} \ -1 \\ 2 \overline{)40} \ -1 \\ \underline{2 \ 20} \ -0 \\ 2 \overline{)10} \ -0 \\ \underline{2 \ 5} \ -0 \\ 2 \overline{)2} \ -1 \\ \underline{2 \ 1} \ -0 \\ \hline 0 \ -1 \end{array} $ <p style="text-align: center;">↑</p> <p>Successive division - integer part</p>	$ \begin{array}{r} 875 \\ \times 2 \\ \hline 1.750 \\ \downarrow \\ 0.75 \\ \times 2 \\ \hline 1.50 \\ \downarrow \\ 1.00 \end{array} $ <p style="text-align: center;">↓</p> <p>Successive multiplication fraction part</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

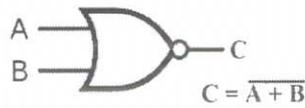
$$163.875 = (10100011.111)_2$$

3 x 3

9

IV
b

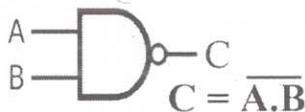
NOR GATE



TRUTH TABLE		
INPUT		OUTPUT
A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

This is a NOT-OR gate which is equal to an OR gate followed by a NOT gate. The outputs of all NOR gates are low if any of the inputs are high. The symbol is an OR gate with a small circle on the output. The small circle represents inversion.

NAND GATE



Truth Table		
INPUT		OUTPUT
A	B	A NAND B
0	0	1
0	1	1
1	0	1
1	1	0

This is a NOT-AND gate which is equal to an AND gate followed by a NOT gate. The outputs of all NAND gates are high if any of the inputs are low. The symbol is an AND gate with a small circle on the output. The small circle represents inversion.

6

3 x 2

V
a

A maxterm is a sum (OR) of all the variables in the function, in direct or complemented form. A maxterm has the property that it is equal to 0 on exactly one row of the truth table.

A (B' + A) B – POS

$$A = A + BB' = (A + B).(A + B')$$

$$B = B + AA' = (A + B).(A' + B)$$

$$\begin{aligned} A (B' + A) B &= (A + B).(A + B')(A + B')(A + B).(A' + B) \\ &= (A + B).(A + B').(A' + B) \end{aligned}$$

Definition –
2

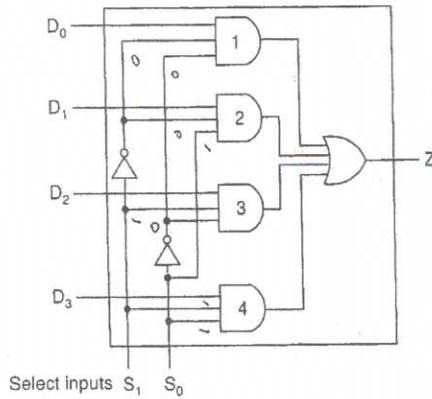
Expansion
–6

8

V
b

A multiplexer is a logic circuit that accepts several data inputs but only one output. In general a MUX has n input signals, m control signals and 1 output signal. And using these m control signal it can select atmost 2^m input signals. That is $n \leq 2^m$. And hence the MUX can also be named as a data selector.

A 4x1 multiplexer have 4 inputs.
It requires 2 data select inputs
The logic levels applied to s_0 and s_1 determine which AND gate is selected



Select inputs		Output
S_1	S_0	Z
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

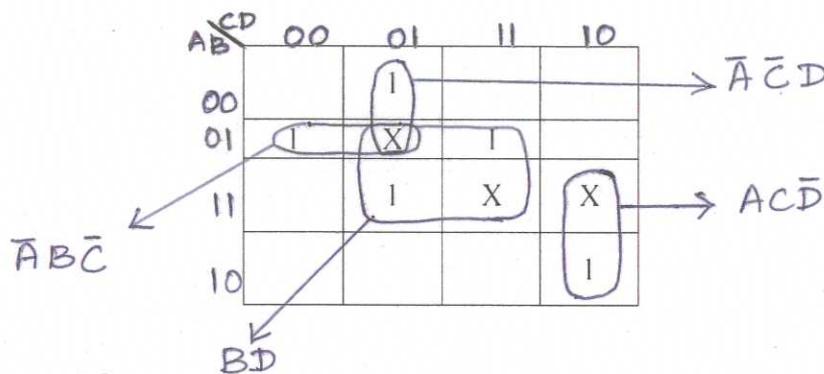
Figure - 4

7

Explanation
- 3

VI
a

$F(A, B, C, D) = \sum m(1, 4, 7, 10, 13) + d(5, 14, 15)$



$$F(A, B, C, D) = A'B C' + A'C'D + ACD' + BD$$

Figure - 6

8

Terms - 2

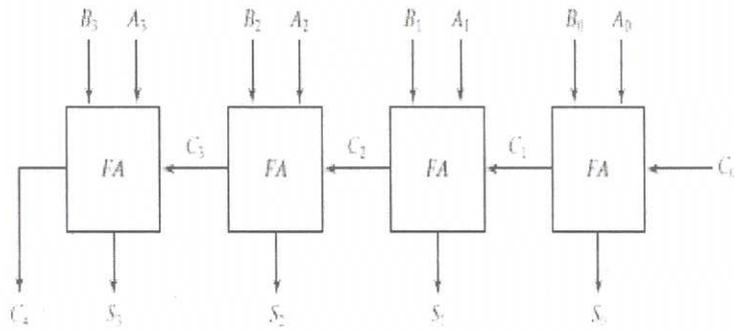
VI
b

A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers. It can be constructed with full adders connected in cascade. The output carry from each full adder connected to the input carry of the next full adder in the chain.

Fig - 4

7

Addition of n-bit numbers requires a chain of n full adders or a chain of one-half adder and n - 1 full adders.

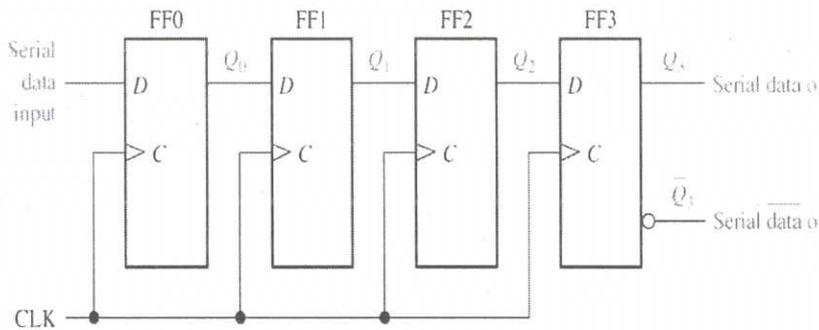


Explanation - 3

VII
 Serial In / Serial Out Shift Registers (SISO)
 Serial In / Parallel Out Shift Registers (SIPO)
 Parallel In / Serial Out Shift Registers (PISO)
 Parallel In / Parallel Out Shift Registers (PIPO)

Serial In Serial Out Shift Registers

Serial data is transferred into a register
 New bit is clocked into the first flip flop at each clock
 Previous bit is transferred to the next ff
 The bit on the last ff is shifted out



Listing - 2

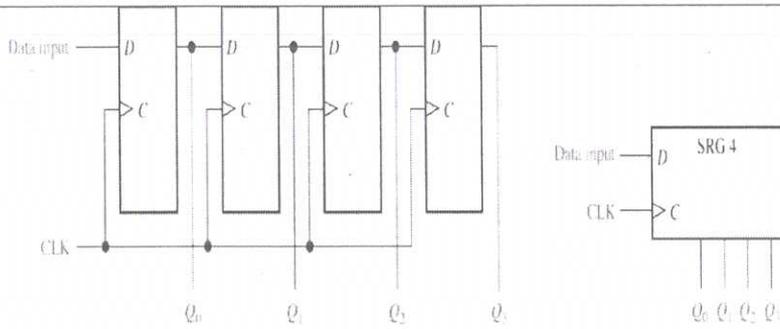
Fig - 4 x 2 (8)

15

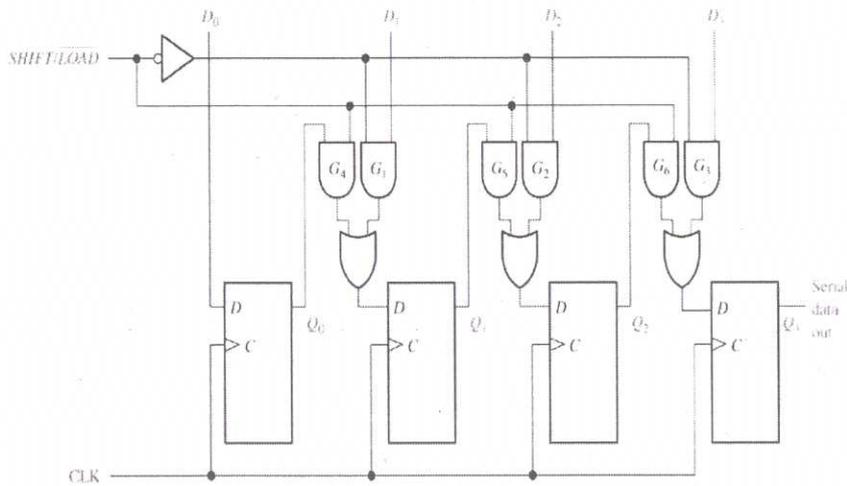
Serial In, Parallel Out Shift register (SIPO)

Data bits entered serially (right-most bit first)
 Difference from SISO is the way data bits are taken out of the register - in parallel.
 Output of each stage is available

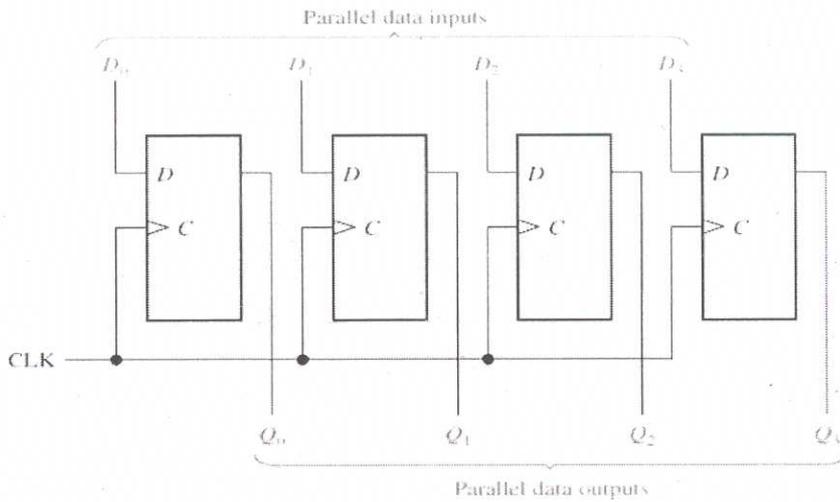
Explanation - 5



4-bit parallel in/serial out shift register (PISO)



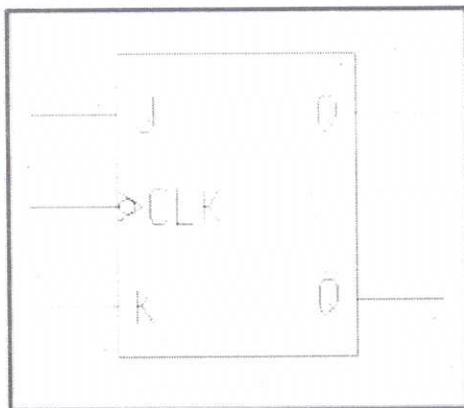
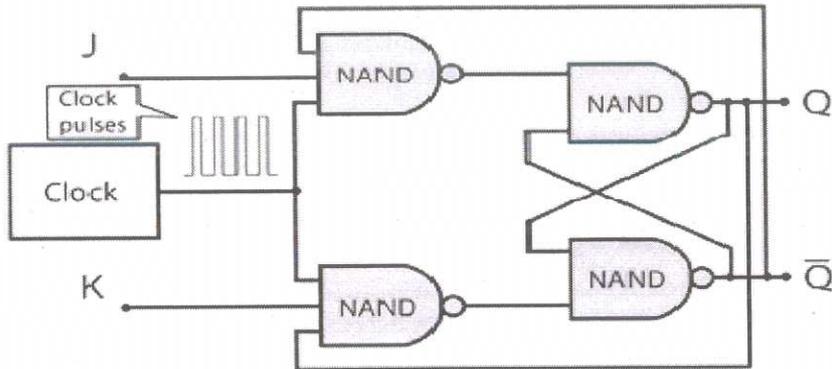
Parallel In, Parallel Out Shift Register (PIPO)



Immediately following simultaneous entry of all data bits, it appear on parallel output.

VIII J-K flip-flop:

- a Q and Q' are fed back to the pulse-steering NAND gates.
 No invalid state.
 J=HIGH (and K=LOW) - SET state
 K=HIGH (and J=LOW) - RESET state
 both inputs LOW - no change
 both inputs HIGH - toggle



J	K	CLK	Q(t+1)	Comments
0	0	↑	Q(t)	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	Q(t)'	Toggle

$$Q(t+1) = J \cdot Q' + K' \cdot Q$$

Diagram - 5

Explanation - 3

VIII Ring Counter

b

General behaviour of ring counter
 N bit ring counter needs N FF
 Positive feedback is given
 It has N states.
 It repeats the sequence after N clocks
 The initial status is circulated.
 If all output is zero initially. Then the outputs remains zeros forever.
 And if all are ones initially, then the output remains 1 forever.

4 bit Ring Counter

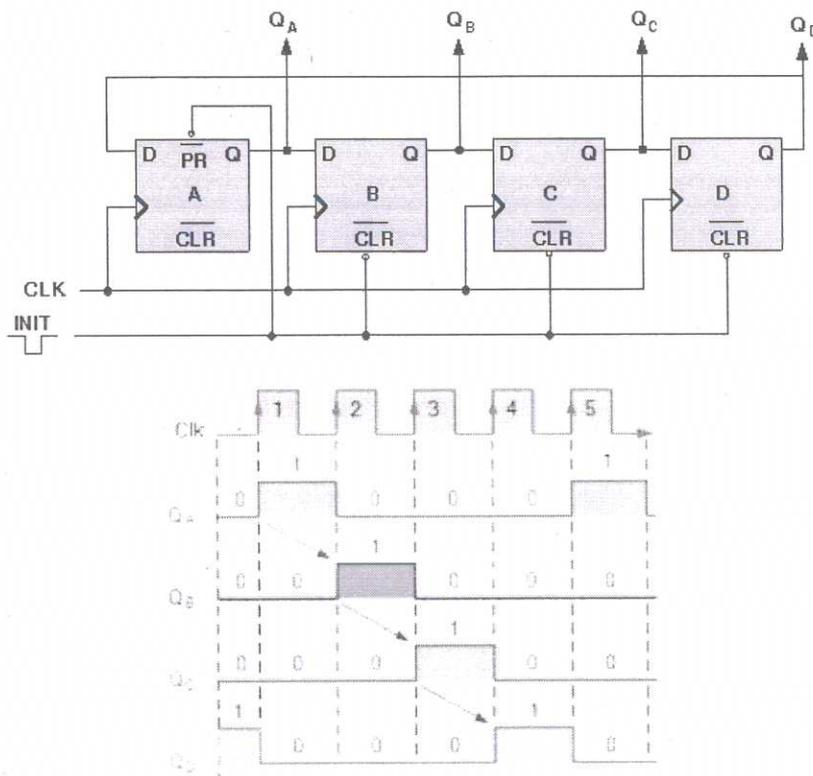


Fig-3

7

Explanation
 - 4

IX

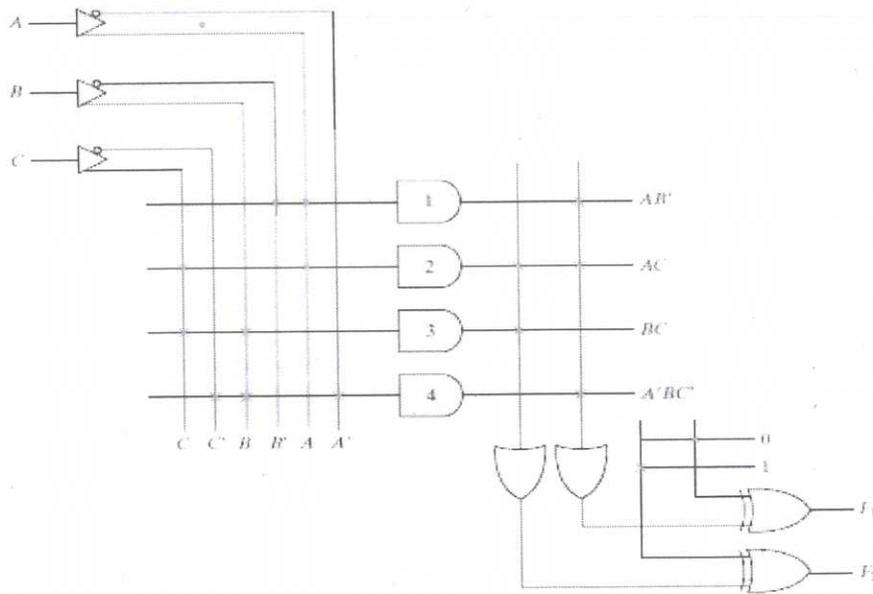
a

Programmable Logic Array (PLA) is a type of PLD.
 It contains a set of AND gate array and OR gate array
 The AND gate array is linked to OR gate array
 For n input variables, It needs n inverter gates 2n AND gates and m OR gates
 The AND matrix can be used to implement the product term and OR matrix for the implementation of sum of product terms

Fig - 6

8

Explanation
 - 2



IX
b

1. Accuracy
It indicates how close the measured value is to the true value.
Ways of specifying accuracy are Full scale error and Linearity error
2. Resolution
It is the smallest change that can occur in the analog output as a result of a change in the digital input. It is the measure of how finely its output may change between discrete, binary steps
3. Offset voltage/Offset error
There will be a very small output voltage when the inputs are all zero. This voltage is called offset error
4. Monotonicity
A DAC is monotonic if its output value increases as the binary inputs are incremented from one value to the next
5. Settling time
It represents the time it takes for the output to settle within a specified final value following a code change at the input.

Listing - 3

Explanation
- 4

7

X
a

Digital to analog converter is used to convert digital quantity into analog quantity. It produces an output current of voltage proportional to digital quantity applied to its input. R-2R DAC is the most popular DAC. It contains a network of two resistors of R and 2R. The binary inputs D4, D3, D2, D1 is applied through the resistors of value 2R.

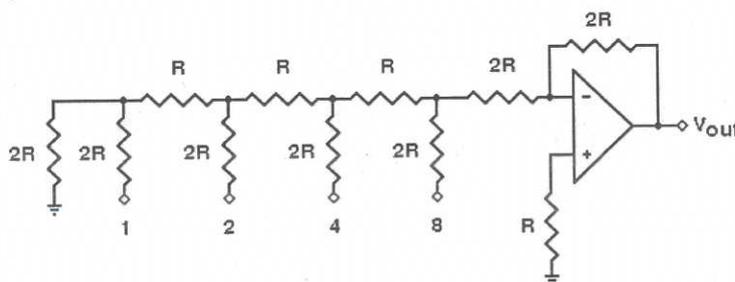


Figure - 4

Explanation
- 4

8

X
b It consists of a comparator, an AND gate, a counter and associated display, and a DAC. Comparator compares the analog input data with the analog converted digital data.

Working: Comparator compares the analog input data V_a with the converted digital data V_d fed back to the comparator input from the output

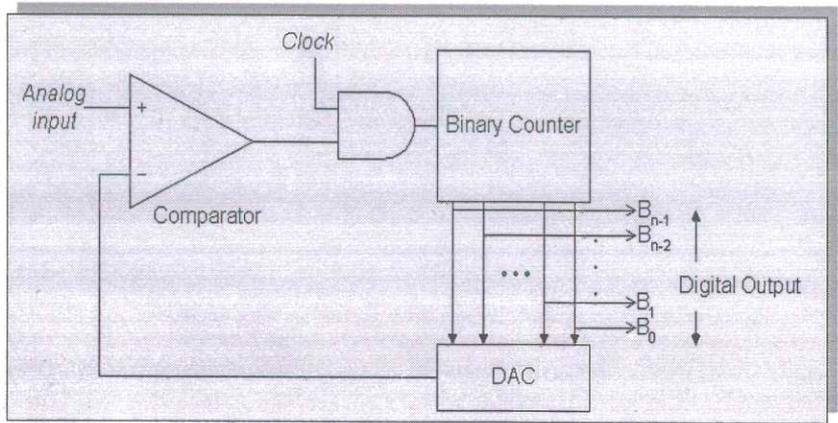


Figure – 4

7

Explanation
- 3