**QP-Code : 20EC11T**

**I Semester Diploma Makeup Examination Nov/Dec - 2022**

**DIGITAL ELECTRONICS (20EC11T)**

# Scheme of Valuation

## SECTION – 1

1. (a) Perform following operations: 2X5=10M
   - (i) Steps -1 M, Correct Answer -1M
   - (ii) Steps -1 M, Correct Answer -1M
   - (iii) Steps -1 M, Correct Answer -1M
   - (iv) Steps -1 M, Correct Answer -1M
   - (v) Steps -1 M, Correct Answer -1M

   (b) (i) Conversion procedure/ Boolean expressions/ Example – 2M

   Logic Diagram -2M 2+2=4M

   (ii) AND Gate Explanation – 1M, Symbol- 0.5M, Boolean Expression-0.5, T.T- 1M

   NAND Gate Explanation – 1M, Symbol- 0.5M, Boolean Expression-0.5, T.T- 1M

   3X2=6M

2. (a) Perform following operations: 2X5=10M
   - i. Steps -1 M, Correct Answer -1M
   - ii. Steps -1 M, Correct Answer -1M
   - iii. Steps -1 M, Correct Answer -1M
   - iv. Steps -1 M, Correct Answer -1M
   - v. Steps -1 M, Correct Answer -1M

   (b) (i) Explanation: 4M, Example: 1M 4+1=5M

   (ii) Output of AND gate -1.5M

   Output of OR gate -1.5M

   Output of EX-OR gate -2M 1.5+1.5+2=5M

## SECTION – II

3. a) Each gate Symbol + Expression- 1M
   Truth table - 1M 2x5=10M

   b) (i) Equation steps 3 marks + Logic diagram 2 marks 3+2=5M

   (ii) Equation steps 3 marks+ Logic diagram 2 marks. 3+2=5M

4. a) Each logic gate carries 2 marks. (any 5 Gates) 2x5=10M

| | | | |
|---|---|---|---|
| b) (i) | Definition of SOP | 1.5M | |
| | Examples SOP | 1M | |
| | Definition of POS | 1.5M | |
| | Examples POS | 1M | 1.5+1+1.5+1=5M |
| (ii) | Expression POS | 1M | |
| | Missing terms | 2 M | |
| | Expression SOP | 2M | 1+2+2=5M |

## SECTION -III

**5 (a).** Block diagram - 5Marks

Explanation - 5 Marks 5+5=10M

**(b)** Truth table - 2 Marks
Explanation - 2 Marks

K-Map simplification - (1+1) = 2 Marks

Logic diagram - (2+2) = 4 marks 2+2+2+4=10M

**6(a).** Block diagram - 1 Mark

Explanation - 1Mark

Truth table - 2 Marks

K-Map simplification - (1+1+1) = 3Marks

Logic Diagram - 3 Marks 1+1+2+3+3=10M

**(b) (i)** Each comparison carries 1 mark 1*5= 5M

**(ii)** Mention any five Multiplexer ICs with their function 1*5=5M

## SECTION – IV

7. (a) (i) Each gate realization 2x3=6M
   NOT gate 2M
   AND gate 2M
   OR gate 2M

   (ii) Realization of 4:1Mux 4M

   (b) Definition (2M) + truth table (3M) + equation(2M) + logic diagram (3M) 2+3+2+3=10M

8. (a) truth table (3M) + equation(2M) + logic diagram (3M) + explanation(2M) 3+2+3+2=10M

   (b) (i) Any Five applications of Multiplexers. 1*5=5M

   (ii) Any FIVE applications of Encoders. 1*5=5M

## SECTION – V

9. (a) Logic diagram(2M) + Logic expressions (3M) + Truth table(2M) + Gate level Circuit (3M) 2+3+2+3=10M

(b) Interfacing -7M 7M

(c) Any 3 valid points 3M

10. (a) Any 5 valid points 1*5=5M

(b) At least 3 valid points for advantages (3M)
At least 2 valid points for disadvantages (2M) 3+2=5M

(c) 1 mark for each class 1*5=5M

(d) 1 mark for each definition 1*5=5M

**********

**QP-Code : 20EC11T**

**I Semester Diploma Makeup Examination Nov/Dec - 2022**

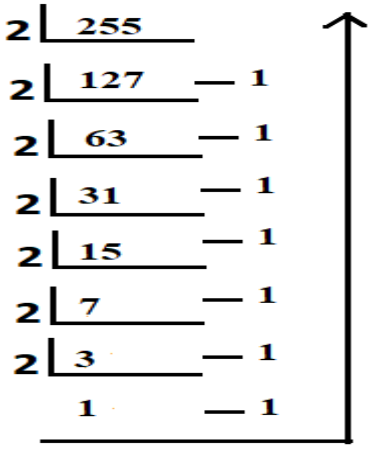
**DIGITAL ELECTRONICS (20EC11T)**

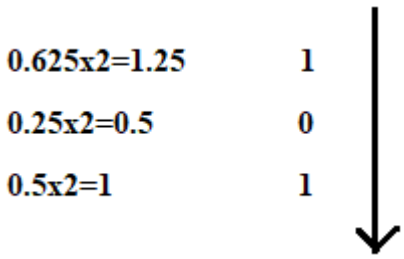
# Model Answers

## SECTION – 1

**1. (a) Perform following operations:** **2*5=10M**

(i) $255.625_{(10)}$ = _____$_{(2)}$




$255.625_{(10)} = 11111111.101_{(2)}$

(ii) $10100101_{(2)}$ = ……$_{(16)}$

Ans. 1010 0101 = $A5_{(16)}$

(i) $BDA_{(16)}$ = ……$_{(10)}$

$= Bx16^2+Dx16^1+16^0xA$

$=2816+208+10$

$=3034_{(10)}$

(ii) $101011_{(2)}$ = ……$_{(8)}$

101 011 $_{(2)} = 53_{(8)}$

(iii) 2's compliment of 1100110 = _____

1's compliment of 1100110=0011001

Add 1 to 1' compliment ➔ 0011001

+ 1

0011010

**(b) (i) Realize the Gray code to binary converter using logic gates.** **2+2=4M**

The conversion 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.

Example. Let the Gary code be $ABCD_{(2)}=1101_{(2)}$

Then the binary bits can be found as given below.

B1=A =1

B2=B1⊕B =1⊕1=0

B2=B2⊕C =0⊕0=0

B3=B2⊕D =0⊕1=1

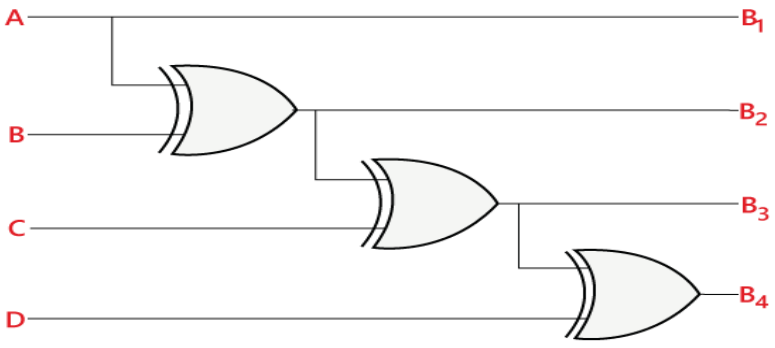
Thus Binary number for the given gray code is $=1001_{(2)}$

Expressions: B1=A ,

B2=B1⊕B

B2=B2⊕C

B3=B3⊕D




Logic Circuit for Gray to Binary Code Converter

**(ii) Explain the AND Gate and NAND gate with logic symbol, Truth table and Boolean expressions.** **1+0.5+0.5+1+1+0.5+0.5+1=6M**

AND Gate is a logic circuit with two or more inputs and one output that performs logical multiplication. Its output is HIGH (1) only when all its inputs are HIGH.


**Logic Symbol**

**Boolean Expression Q= A.B**

**Truth Table :**

| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

NAND Gate

NAND Gate is logic circuit with two or more inputs and one output. Its output is LOW (0) only when all its inputs are HIGH(1).


**Logic Symbol**

**Boolean Expression Q= $\overline{A.B}$**

**Truth Table :**

| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**2. (a) Perform following operations:** **2*5=10M**

(i) Subtract $00111_{(2)}$ from $10101_{(2)}$ using 2's compliment method.

a. Take the 2's compliment of the minuend i.e 00111

1' compliment – 11000

$$\begin{array}{r} 11000 \\ +\quad\ 1 \\ \hline 11001 \end{array}$$

Now add 2's compliment of minuend to the subtrahend

$$\begin{array}{r} 10101 \\ +\ 11001 \\ \hline 101110 \end{array}$$

Discard the carry.

Answer: $01110_{(2)}$

(ii) Multiply $1011_{(2)}$ with $0111_{(2)}$

$$\begin{array}{r} 1011 \times 0111 \\ \hline 1011 \\ 1011 \\ 1011 \\ 0000 \quad . \\ \hline 1001\ 101 \end{array}$$

(iii) Add $1234_{(16)}$ and $ABCDE_{(16)}$

$$\begin{array}{r} 1234_{(16)} \\ ABCDE_{(16)} \\ \hline ACF12_{(16)} \end{array}$$

(iv) Perform BCD addition on two BCD numbers $1001_{(2)}$ and $0011_{(2)}$

$$\begin{array}{r} 1001_{(2)} \\ 0011_{(2)} \\ \hline 1100_{(2)} \end{array}$$

Since the sum is greater than 9 , add 6 to the sum

$$\begin{array}{r} 1100 \\ 0110 \\ \hline 10010 \end{array}$$

The BCD sum is $0001\ 0010_{(2)}$

(v) The excess-3 code of $1010_{(2)}$ =_____.
Excees-3 code for a binary number is obtained by adding 3 to it.

$1010_{(2)}$
$0011_{(2)}$
$1101_{(2)}$

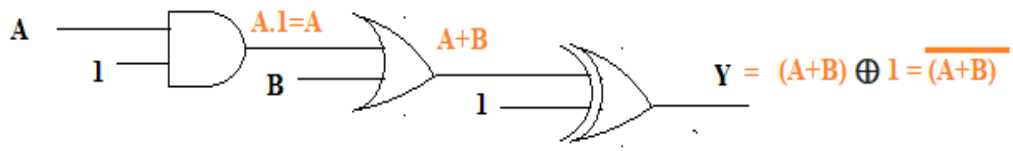
**(b) (i) Explain ASCII codes. Give an example.** **4+1=5M**

ASCII stands for American Standard Code for Information Interchange. Computers can only understand numbers, so an ASCII code is the numerical representation of a character such as 'a' or '@' . It is used for data communication in digital computers. The ASCII code is a 7-bit code capable of representing $2^7$ (128) number of different characters. The ASCII code starts from 00h to 7Fh. In this, the code from 00h 20h to 7Fh is used for graphic symbols.

For example: ASCII value of Alphabet A is $41_{(16)}$

ASCII value of Alphabet * is $2A_{(16)}$

**(ii) Apply Boolean algebra to find the expression for Y.** **1.5+1.5+2=5M**

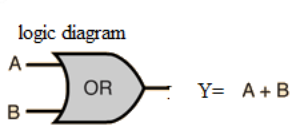



# SECTION – II

3. **a) Explain following gates with symbol, expression and truth table.**
   **AND, OR, NOR, NAND, EXOR.** **2*5=10M**

**OR Gate:**

The OR gate output attains the state 1 if either one or more inputs attain the state 1.
Boolean expression of OR gate can be given by,
Y = A + B




Truth Table

| A | B | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

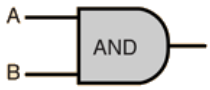
**AND Gate:**

In an AND gate, the output attains state 1 if and only if all the inputs are in state 1.

Boolean expression of AND Gate can be given by,

$Y = A \cdot B$

Logic Diagram




Y= AB

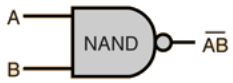
Truth Table

| A | B | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

**NAND Gate:**

It's a digital circuit with two or more inputs that creates an output that's the logical AND of all those inputs inverted.

Logic Diagram



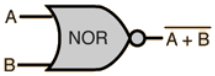

Truth Table

| A | B | $Y=\overline{AB}$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**NOR Gate:**

It's a digital circuit with two or more inputs that creates an output that's the logical OR of all those inputs inverted.

Logic Diagram

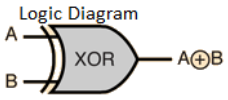



Truth Table

| A | B | $\overline{A+B}$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

**EXOR Gate:**

The Exclusive-OR gate is known as the XOR gate. If one of the input is 1,the Output is 1 or else output is zero.




Truth Table

| A | B | Y=A⊕B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**b) Simplify the following logic expression using Boolean Algebra. Draw the logic diagram.** **3+2, 3+2=10M**

i) Y=(A+B)(A+ $\bar{\bar{B}}$)($\bar{A}$+B)

Solution.

Y= (A+B)(A+B) ($\bar{A}$ + $B$).

Y=(A+B)($\bar{A}$ + $B$).

Y=(A+2AB+B)($\bar{A}$ + $B$).

Y=A. $\bar{A}$+AB+B$\bar{A}$ +B.B

Y=0+AB+$\bar{A}$ B+B

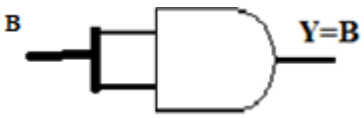
Y=AB+B($\bar{A}$ + 1).

Y=AB+B

Y=B(A+1)

Y=B

logic diagram:




ii) Y=ABCD+AB$\overline{CD}$+$\overline{AB}$ CD.

Y= ABCD+AB($\bar{c}$+$\bar{D}$) + ($\overline{A}$+$\bar{B}$)$CD$

$Y$=$ABCD$+$AB$ $\bar{c}$+$AB\bar{D}$+$\bar{A}CD$+$\bar{B}CD$

$Y$=$AB$ ($\bar{c}$+$CD$) + $AB\bar{D}$+$\bar{A}CD$+$\bar{B}CD$

Y= AB($\bar{c}$+D) + $AB\bar{D}$+$\bar{A}CD$+$\bar{B}CD$

Y= $AB$ $\bar{c}$+$ABD$+ $AB\bar{D}$ + $\bar{A}CD$+$\bar{B}CD$

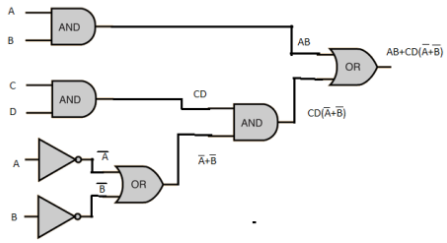
$Y$= $AB$ $\bar{c}$+$AB$ ($D$+ $\overline{D}$) + $\bar{A}CD$+$\bar{B}CD$

$Y$= $AB$ $\bar{c}$+$AB$+$\bar{A}CD$+$\bar{B}CD$

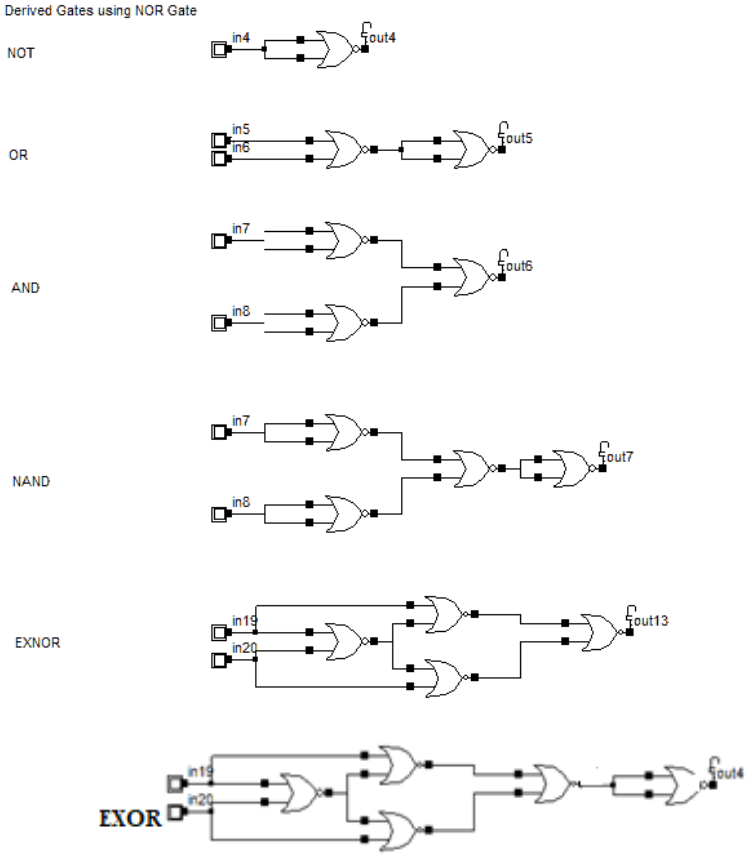
$Y$= $AB$ ($\bar{c}$+1)+$\bar{A}CD$+$\bar{B}CD$

$Y$= $AB$ +$CD$ ($\bar{A}$+$\bar{B}$)

Logic Diagram




**4.(a) Show the realization of all logic gates using NOR gate.** **2*5= 10M**

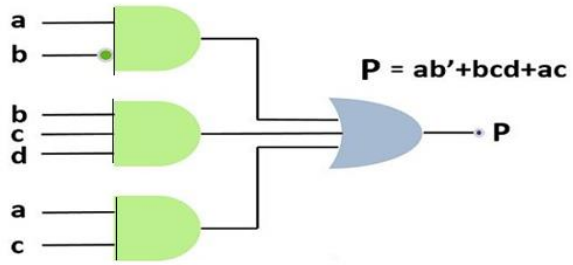



- **NOT using NOR**: It's simple. Just connect both the inputs together.
- **OR using NOR**: Connect a NOT using NOR at the output of the NOR to invert it and get OR logic.
- **AND using NOR**: Connect two NOT using NORs at the inputs of a NOR to get AND logic.
- **NAND using NOR**: Just connect another NOT using NOR to the output of an AND using NOR.
- **EXNOR using NOR:** This one's a bit tricky. You share the two inputs with three gates. The output of the first NOR is the second input to the other two. Finally, another NOR takes the outputs of these two NOR gates to give the final output.
- **EXOR using NOR:** Just connect another NOT gate using NOR to the output of EXNOR using NOR gate.

**(b) (i) What is SOP and POS with examples.** **1.5+1+1.5+1=5M**

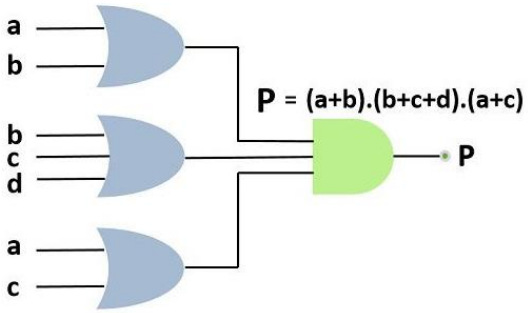
When we add two or multiple product terms by a Boolean addition, the output expression is a **sum-of-products (SOP)**.

Example, the expression a'bc' + a'bd' + a'bc'd shows a SOP expression. It can also have a single variable term within the expression like a + bc +a'b.




**POS (Product of Sums)** is the representation of the Boolean function in which the variables are first summed, and then the Boolean product is applied to the sum terms.

Example, (a'+b).(a+b'+c) is POS expression where we can see that the variables are added then each bigger term is the product of the other.




**ii) Illustrate POS to SOP conversion F (A, B, C) = (2, 3, 5).** **1+2+2=5M**

POS Expression is F(A+$\bar{B}$+C)(A+$\bar{B}+\bar{c}$) $\overline{(A}+B+\bar{c})$

*Missing terms are*

*F(A,B,C)=∑m(0,1,4,6,7)*

*SOP Expression is*

*F= `$\bar{A}\bar{B}\bar{c}$+ $\bar{A}\bar{B}$ c +A$\bar{B}\bar{c}$ + AB$\bar{c}$ +ABC*

# SECTION -III

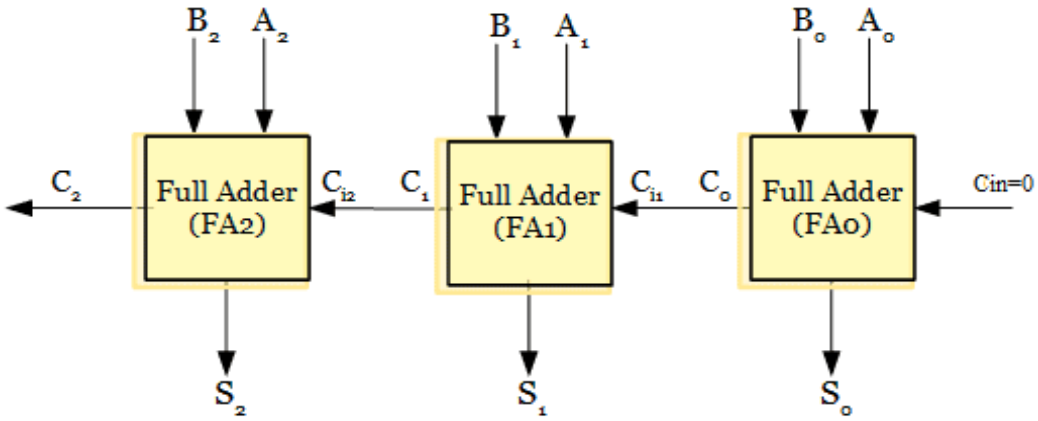
**5(a). Analyze the working of 3-bit parallel adder.** **5+5=10 M**

In order to add binary numbers with more than one bit, additional full-adders must be employed.

A n-bit, parallel adder can be constructed using n-number of full adder circuits connected in parallel.

Figure shows the block diagram of 3-bit parallel adder using three full-adder circuits connected in cascade, i.e. the carry output of each adder is connected to the carry input of the next higher-order adder.

It should be noted that either a half-adder can be used for the least significant position(LSB) or the carry input of a full-adder is made 0 because there is no carry into the LSB.




Block diagram of 3-bit parallel adder

As shown in the figure, firstly the full adder $FA_0$ adds $A_0$ and $B_0$ along with the carry $C_{in}$ to generate the sum $S_0$ (the first bit of the output sum) and the carry $C_0$ which is connected to the next adder in chain.

Next, the full adder $FA_1$ uses this carry bit to add with the input bits $A_1$ and $B_1$ to generate the sum $S_1$ (the second bit of the output sum) and the carry $C_1$ which is again further connected to the next adder in chain and so on.

The process continues till the last full adder FAn uses the carry bit Cn to add with its input An and Bn to generate the last bit of the output along last carry bit Cout.

**5(b).Explain the working of full subtractor using truth-table , Boolean expressions and logic diagram. 2+2+2+4= 10M**

A full-subtractor is combinational circuit that performs a subtraction between two binary bits, taking into account borrow of the lower significant stage.

- This circuit has three inputs and two outputs.
- The three inputs are A,B and Bin denote the minuend, subtrahend and previous borrow, respectively.
- The two outputs,D and Bout, represent the difference and output borrow, respectively.

Truth table for full subtractor

| Inputs | | | Outputs | |
|---|---|---|---|---|
| A | B | $B_{in}$ | D | $B_{out}$ |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

K-Maps for full subtractor

**For D**

| A \ $BB_{in}$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |

$D = \overline{A}\overline{B}B_{in}+\overline{A}B\overline{B}_{in}+A\overline{B}\overline{B}_{in}+ABB_{in}$

**For $B_{out}$**

| A \ $BB_{in}$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 |

$B_{out} = \overline{A}B_{in}+\overline{A}B+BB_{in}$

**Boolean function for D (difference) can be further simplified as follows:**

$$D = \overline{A}\,\overline{B}\,B_{in} + \overline{A}\,B\,\overline{B}_{in} + A\,\overline{B}\,\overline{B}_{in} + A\,B\,B_{in}$$

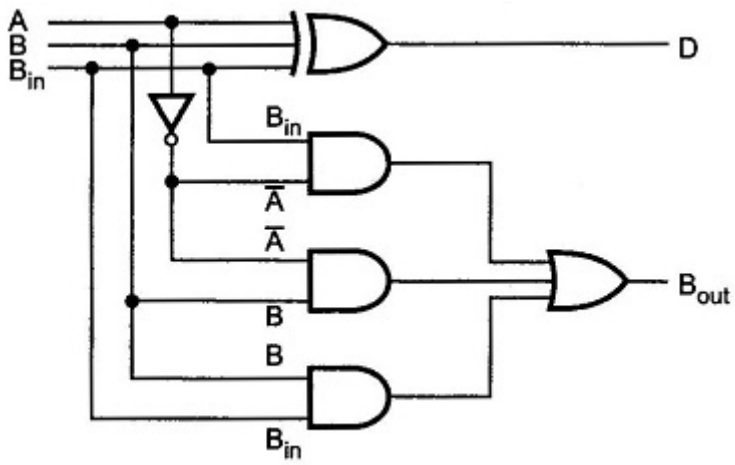
$$= B_{in}\,(\overline{A}\,\overline{B} + AB) + \overline{B}_{in}\,(\overline{A}\,B + A\,\overline{B})$$

$$= B_{in}\,(A \odot B) + \overline{B}_{in}\,(A \oplus B)$$

$$= B_{in}\,(\overline{A \oplus B}) + \overline{B}_{in}\,(A \oplus B)$$

$$= B_{in} \oplus (A \oplus B)$$

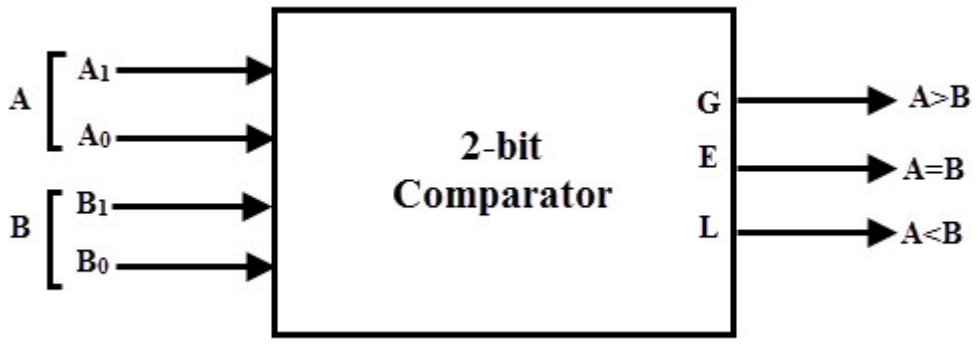
Full subtractor Logic Diagram




**6(a). Design two bit magnitude comparator using logic gates. 1+1+2+3+3=10 M**

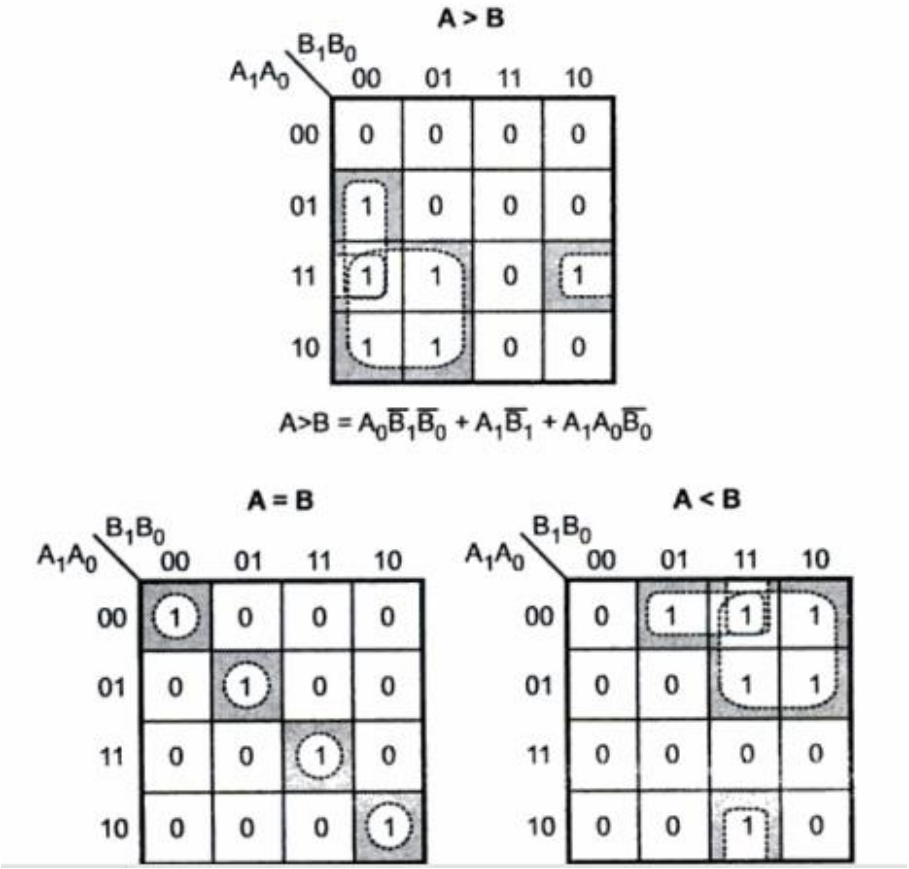
A comparator is a special combinational circuit designed primarily to compare the relative magnitude of two binary numbers. Fig. shows the block diagram of a 2-bit comparator It receives two 2-bit numbers A and B as inputs and the outputs are :A> B, A=B and A<B .Depending upon the relative magnitudes of the two number, one of the outputs will be high.

Block Diagram



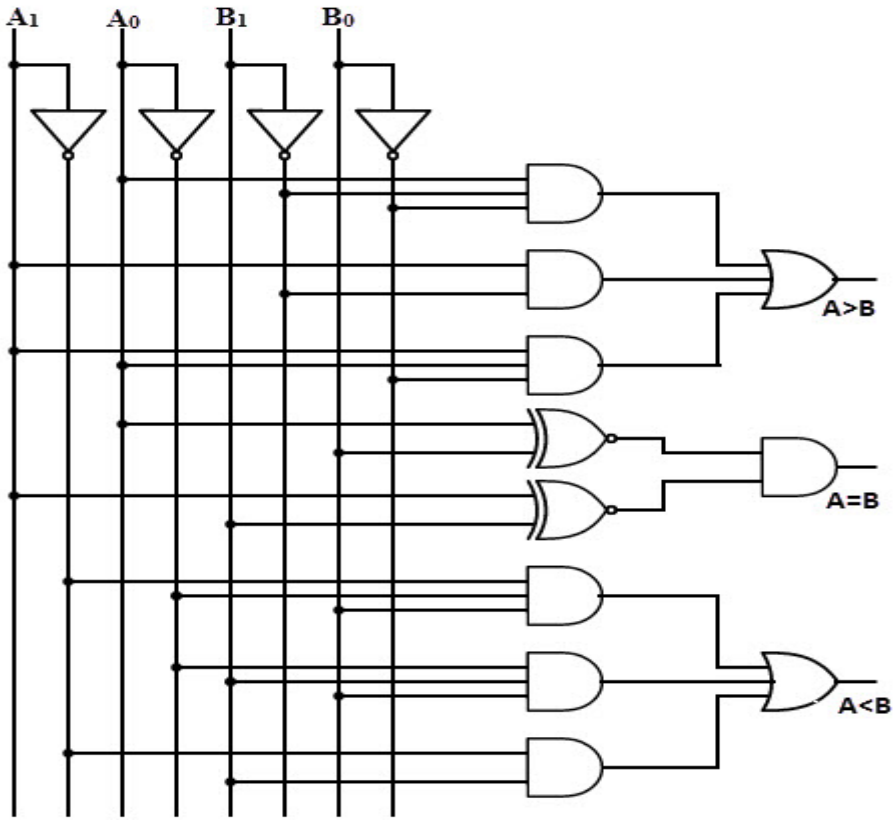
2-Bit Magnitude comparator Truth table

| 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 |




K-Maps

2-bit comparator Logic diagram




**6(b). (i) Compare serial and parallel adder. Any five** **1*5=5 M**

| Sl No | Serial Adder | Parallel Adder |
|---|---|---|
| 1 | It is used to add two binary numbers in serial form. Single bit is added at a time | It is used to add two binary numbers in parallel form. All bits are added at a time with exception of propagation delay present from input carry to output carry |
| 2 | A serial adder uses shift registers | A parallel adder uses registers with parallel loads |
| 3 | It requires single full adder | It requires multiple full adders. Number of required full adder is equal to the number of bits in the binary number |
| 4 | Carry flip-flop is used in serial adder | Ripple carry adder is used in parallel adder |
| 5 | Serial adder is a sequential circuit | Parallel adder is a combinational circuit |
| 6 | Addition is slower | Addition is faster |

**6(b) (ii) List the various multiplexer ICs and their functions. Any five** **1*5=5 M**

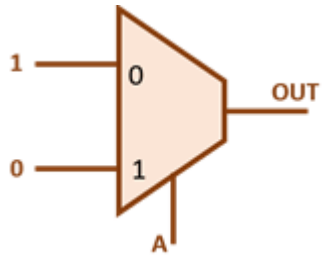
| IC number | Function |
|---|---|
| 74150 | 16:1 multiplexer |
| 74151 | 8:1 multiplexer |
| 74352 | Dual 4:1 multiplexer with inverse output |
| 74153 | Dual 4:1 multiplexer |
| 74157 | Quad 2:1 multiplexer |
| 74158 | Quad 2:1 multiplexer with inverse input |

=

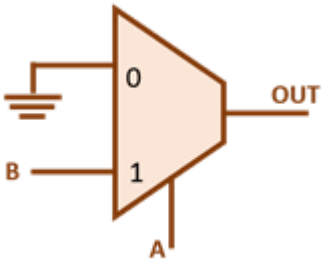
# SECTION - IV

**7. (a) (i) Realize basic gates using 2:1 Mux.** **2+2+2= 6M**

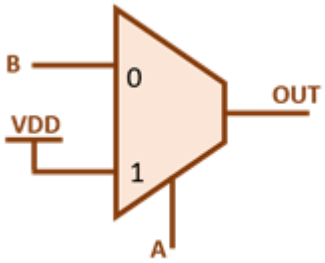
**NOT Gate :**



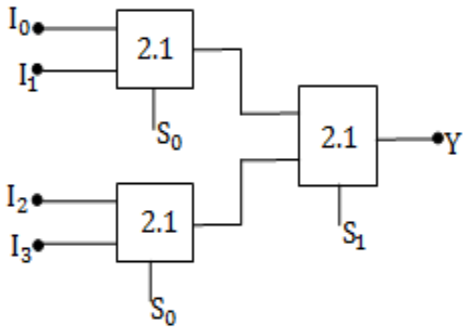

**AND Gate:**




**OR Gate:**

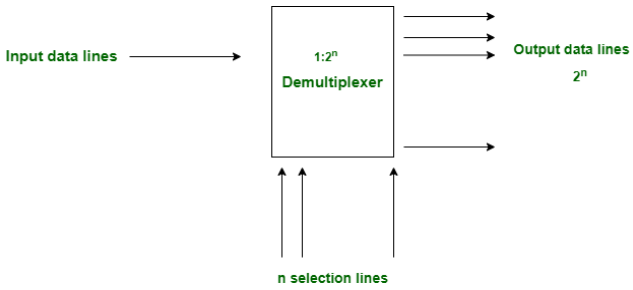


**(ii) Realize 4:1 Mux using 2:1 Mux** **4M**

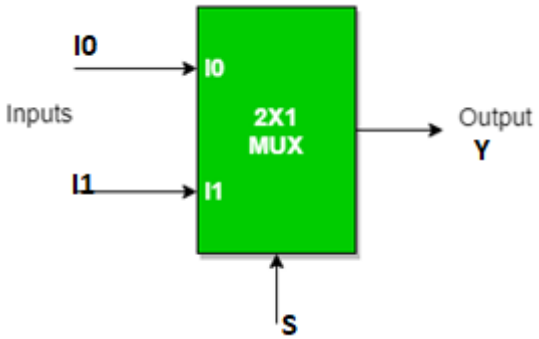



**(b) What are Demultiplexers? Write the truth table, equation and logic diagram of 2:1 MUX.** **2+3+2+3=10M**

**De-Multiplexer is a combinational logic circuit that receives the information on a single input and transmits the same information over one of N possible output lines. It consists of 1 input line, 'n' output lions and 'm' select lines**




Logic diagram of 2:1Mux:




**Truth Table of 2:1 Mux:** **Equation: Y= SI0 + S'I1**

| S | Y |
|---|---|
| 0 | I1 |
| 1 | I0 |

**8.(a) Write the truth table, equation and logic diagram and explain decimal to**

**BCD Encoder** **3+2+3+2=10M**

**To convert a decimal number into binary a Decimal to BCD Encoder is used**. In the BCD system, the decimal number is represented as the four-digit binary. It can convert the decimal numbers from 0 to 9 into the binary stream.

**Equations:**

Y0= I8 + I9

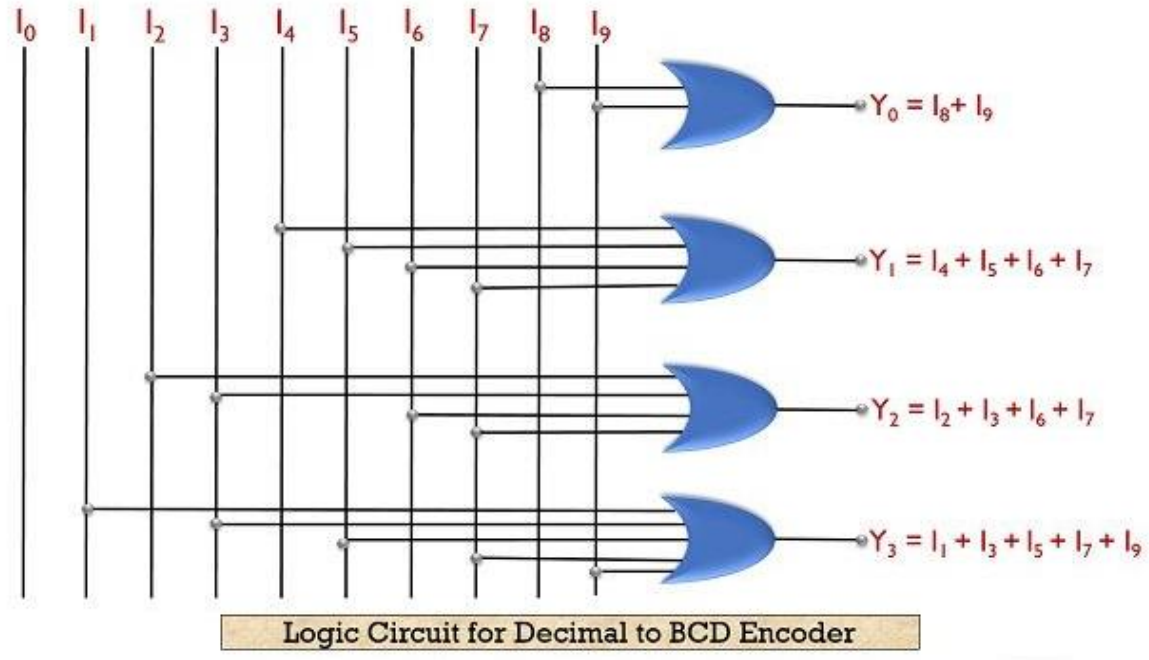
Y1= I4 + I5 + I6 +I7

Y2= I2 + I3 + I6 + I7

Y3= I1 + I3 + I5 + I7 + I9

Truth Table:

| Decimal Digit | BCD Code | | | |
|---|---|---|---|---|
| | Y0 | Y1 | Y2 | Y3 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 |
| 6 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 |




Logic Circuit for Decimal to BCD Encoder

**(b) (i) Write any Five applications of Multiplexers. 1*5=5M**

1. Communication System. A communication system has both a communication network and a transmission system. ...
2. Computer Memory. ...
3. Telephone Network. ...
4. Transmission from the Computer System of a Satellite. ...
5. Communication System. ...
6. Arithmetic Logic Unit. ...
7. Serial to Parallel Converter. ...
8. Photo Credits.

**(ii) Write any FIVE applications of Encoders. 1*5=5M**

- Automatic health monitoring systems.
- RF-based home automation system.
- Robotics vehicle with the metal detector.
- War field flying robot with a using night-vision flying camera.
- Speed synchronization of multiple motors in industries.
- Encoder for CNC machines.
- Encoder for the medical industry most common for breast cancer treatment in the world.
- Encoder for the electronics industry.
- Aerospace. Encoders provide high-precision motion feedback while
- operating in extreme environmental conditions.
- Autonomous Vehicles & Robots. ...
- Backstop Gauging. ...

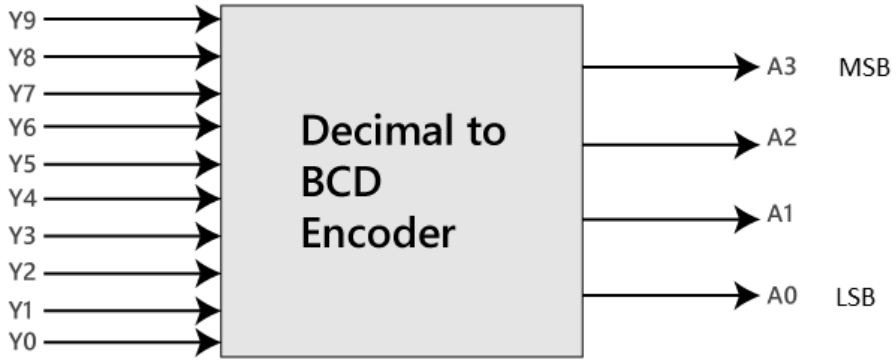
- Ball Screw Positioning. ...
- Converting. ...
- Conveying. ...
- Cut-to-Length. ...

# SECTION – V

**9.** ***a) Write the logic circuit, truth table, logic expressions and gate-level circuit for decimal -to-BCD encoder.*** ***2+3+2+3=10M***

**Answer:**

**Logic circuit/diagram**




**Logic Expressions**

$A3 = Y_9 + Y_8$

$A2 = Y_7 + Y_6 + Y_5 + Y_4$

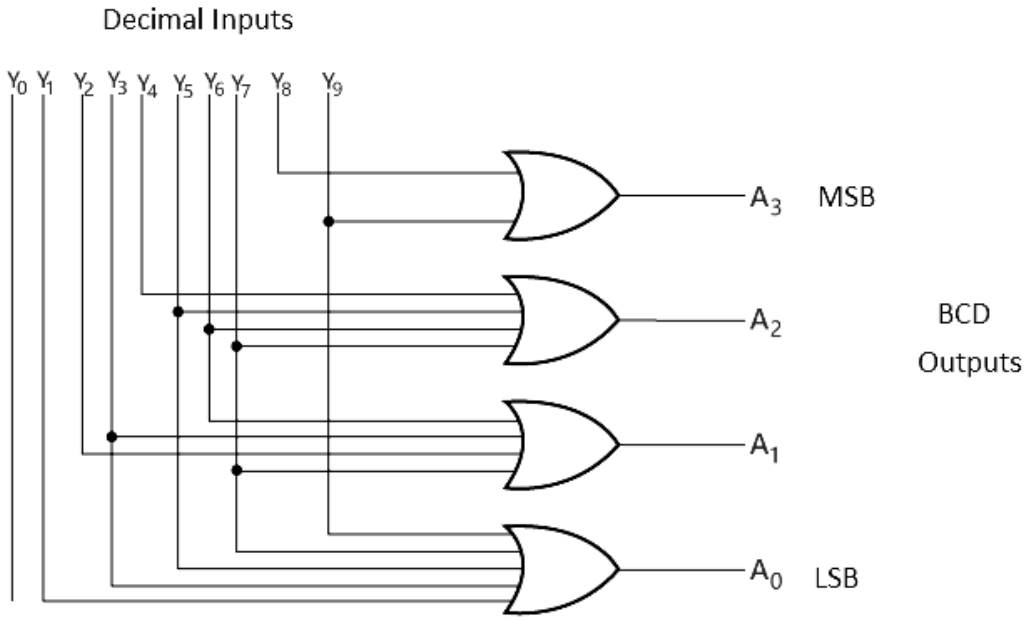
$A1 = Y_7 + Y_6 + Y_3 + Y_2$

$A0 = Y_9 + Y_7 + Y_5 + Y_3 + Y_1$

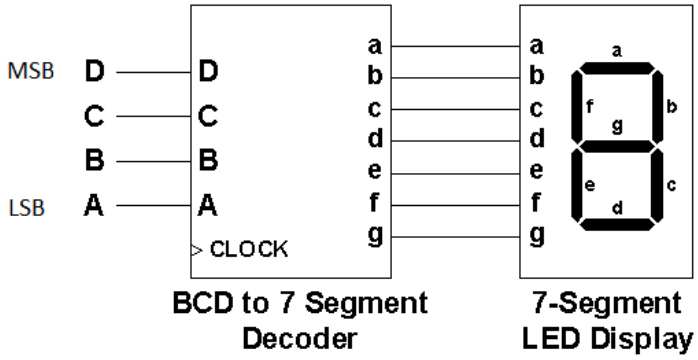
**Truth Table**

| INPUTS | | | | | | | | | | OUTPUTS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $Y_9$ | $Y_8$ | $Y_7$ | $Y_6$ | $Y_5$ | $Y_4$ | $Y_3$ | $Y_2$ | $Y_1$ | $Y_0$ | $A_3$ | $A_2$ | $A_1$ | $A_0$ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

**Gate-level circuit**

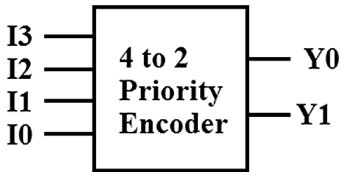



***b) Show how BCD-to-7-segment decoder can be interfaced to 7-segment display.*** ***7M***




***c) Write a note on priority encoder.*** **3M**

Priority encoder is a combinational logic circuit with $2^n$ inputs and $n$ outputs. Normally, encoder's response to multiple active inputs is invalid. This difficulty is overcome by prioritising encoder inputs in priority encoder. It is an encoder that responds to highest/lowest priority input when multiple inputs of an encoder are active. Priority encoders find applications in microprocessors/interrupt controllers, data networks etc. Following is a typical 4:2 priority encoder. Truth table shows that there will be a definite output even if more than one input is active. For example, $I_0$, $I_1$ and $I_2$ all 1s, then it will respond for $I_2$.




| $I_3$ | $I_2$ | $I_1$ | $I_0$ | $Y_1$ | $Y_0$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | X | 0 | 1 |
| 0 | 1 | X | X | 1 | 0 |
| 1 | X | X | X | 1 | 1 |

***10(a) Compare the features of TTL with CMOS family*** **1*5=5M**

| Sl. No. | Parameter/ Factor | TTL | CMOS |
|---|---|---|---|
| 1 | Prominent Components | Transistors | NMOS and PMOS |
| 2 | Fan-out | 10 | >50 |
| 3 | Propagation Delay | 9nS | >50nS |
| 4 | Power Dissipation/gate | 10mW | 10nW |
| 5 | Supply voltage notation | $V_{CC}$ | $V_{DD}$ |
| 6 | Voltage levels | Low: 0V<br>High: 5V | 0 to 1.5V<br>3.3 to 15V |
| 7 | Applications | Medium speed applications<br>less popular now a days | Popular, used for all digital applications including microprocessors and memories. |
| 8 | Advantages | Medium to high speed, good noise immunity | Low power consumption, more input and less output resistance, excellent noise immunity and high density |
| 9 | Disadvantages | Power dissipation is more | Susceptible to electrostatic damage, don't leave inputs floating, slow speed, more propagation delay. |

***b) List the advantages and disadvantages of ICs*** **3+2=5M**

**Advantages of ICs:**

1. Reliability: Work faithfully as all components and interconnection are fabricated and no soldering of joints is done outside
2. Low power: Consumes less power (in terms of mW/µW/pW/nW)
3. Low propagation delay
4. High speed
5. Less weight
6. Occupy less space/Extremely small size
7. Low heating as the power consumption is less
8. Suitable for mass/bulk production
9. Low cost as mass production reduces the cost
10. Easy replacement is possible in a larger system
11. Greater ability of operating at extreme temperatures

**Disadvantages of ICs**

1. In an IC the various components are part of a small semiconductor chip and the individual component or components cannot be repaired, therefore, if any component in an IC fails, the whole IC has to be replaced by a new one.

2. Limited power rating as it is not possible to manufacture high power (say greater than 10 W) ICs.
3. Need of connecting inductors and transformers external to the semiconductor chip as it is not possible to fabricate inductor and transformers on the semiconductor chip surface.
4. Quite delicate in handling ICs as they cannot withstand rough handling/excessive heat/ESD.
5. Need of connecting capacitor exterior to the semiconductor chip as it is neither convenient nor economical to fabricate capacitances exceeding 30pF.

***c) Explain the classification of ICs based scale of integration 1*5=5M***

Scale of integration means the density of components or number of components/gates in a single chip. The following table provides the classification based on scale of integration.

| | |
|---|---|
| Small scale integration (SSI) | Up to 12 gates per chip |
| Medium scale of integration (MSI) | 12 to 100 gates per chip |
| Large scale integration (LSI) | 100 to 1000 gates per chip |
| Very large scale integration (VLSI) | 10000 to 1 lakh gates per chip |
| Ultra large scale integration (ULSI) | Above 1 lakh gates per chip |

*d)* ***Define fan-in, fan-out, propagation delay, power dissipation and speed-power product as applicable to ICs*** **1*5=5M**

**(i) Fan-in:** Maximum number of inputs that can be applied to a logic gate. For example, the fan-in of 2-input AND gate is 2.

**(ii) Fan-out:** The maximum number of load inputs that can be connected to a driver gate without affecting the functionality of the circuit. For example, fan-out of TTL gate is 10 implying that at most 10 TTL load gates can be connected to the driver gate.

**(iii) Propagation delay ($t_p$):** The delay between the application of input and the response or output.

**(iv) Power dissipation (P):** Power dissipation is a product of $V_{cc}$ and $I_{cc}$ where $I_{cc}$ is the average current drawn from power supply source $V_{cc}$ by the gate/IC normally in mW/ $\mu W$.

**(v) Speed-Power product:** It is a product of reciprocal of propagation delay ($1/t_p$) and power dissipation (P) represented in *mJ* or *μJ or nJ* where *J* is joule.

**********