USN □□□□□□□□□□

CMRIT
CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC
CELEBRATING 25 YEARS

Internal Assessment Test 1 – NOV 2025

Sub:	**Digital Design and Computer Organization**				Sub Code:	BCS302	Branch:	CSE
Date:	6/11/25	Duration:	90 mins	Max Marks:	50	Sem/Sec:	3rd semesterA,B,C	OBE

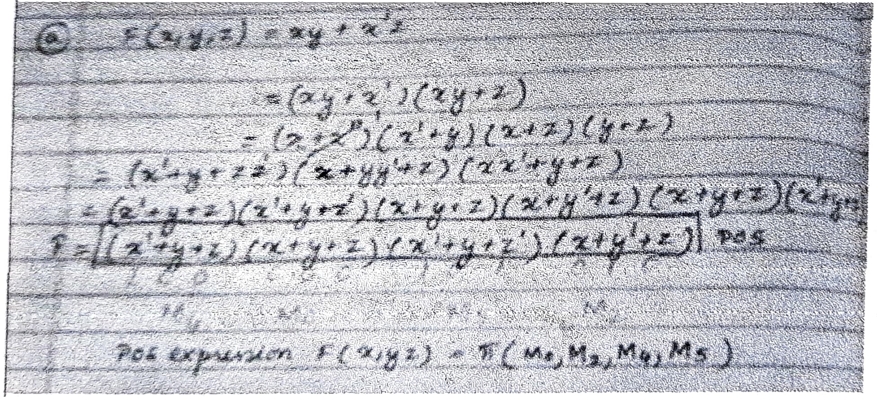
	Answer any FIVE FULL Questions	MARKS	CO	RBT
1a. b.	Express F(x,y,z)=xy+x’z as product of maxterms(pos). Express F(x,y,z)=xy’z+x’y+x’z+yz as sum of products(sop).	[5+5]	CO1	L2,L3
2 a. b.	Obtain the minimum POS of the function $Y = X'Z' + WYZ + W'Y'Z' + X'Y$ using K-MAP. Determine the complement of the following function(i) $F = XY' + X'Y$.(ii) $F = X'YZ' + X'Y'Z$.	[7+3]	CO1	L2,L3
3	Design a BCD to Excess-3 Code converter and obtain the expression for excess -3 code using K-MAP and draw the circuit.	[10]	CO2	L2,L3
4 a. b.	Explain a 4-bit adder subtractor with the circuit diagram? Implement a full adder circuit with truth table, circuit diagram and expression for sum and carry.	[5+5]	CO2	L2,L3
5.a b	What is a multiplexer? Design a 4*1 mux write truth table and draw the logic circuit and give the expression for output? Implement F (A, B, C,D) =∑ (1,3,4,11,12,13,14,15) using a 8:1 MUX?	[6+4]	CO2	L1,L2
6.	Explain the carry look ahead adder with expressions for carry generate and carry propagate and circuit diagram?	[10]	CO2	L2,L3

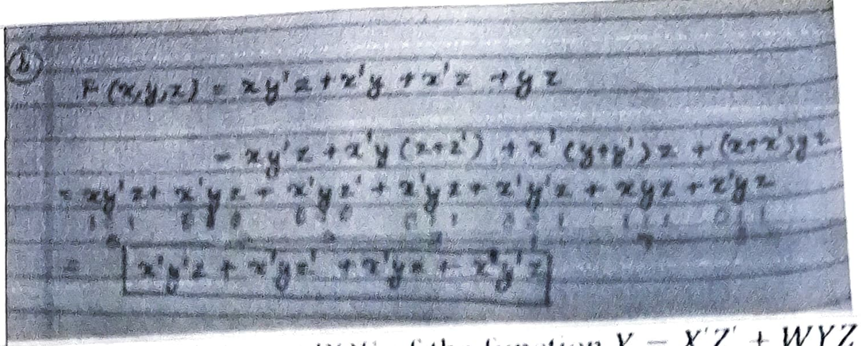
CI　　　　CCI /Anchor Teacher　　　　HOD

CELEBRATING 25 YEARS
CMRIT
CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC

Internal Assessment Test 1 – NOV 2025

Sub:	**Digital Design and Computer Organization**				Sub Code:	BCS302	Branch:	CSE	
	Ms. Smruthi Nair/Ms. Lini								
Date:	6/11/25	Duration:	90 mins	Max Marks:	50	Sem/Sec:	3rd semesterA,B,C	OBE	

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1a.	Express F(x,y,z)=xy+x'z as product of maxterms(pos).  ⓐ $F(x,y,z) = xy + x'z$ $= (xy + x')(xy + z)$ $= (x + x')(x' + y)(x + z)(y + z)$ $= (x' + y + zz')(x + yy' + z)(xx' + y + z)$ $= (x' + y + z)(x' + y + z')(x + y + z)(x + y' + z)(x + y + z)(x' + y$ $F = \boxed{(x' + y + z)(x + y + z)(x' + y + z')(x + y' + z)}$ pos Pos expression $F(x,y,z) = \pi(M_0, M_2, M_4, M_5)$	[5+5]	CO1	L2,L3
).	Express F(x,y,z)=xy'z+x'y+x'z+yz as sum of products(sop).			



(b)

$F(x,y,z) = xy'z + x'y + x'z + yz$

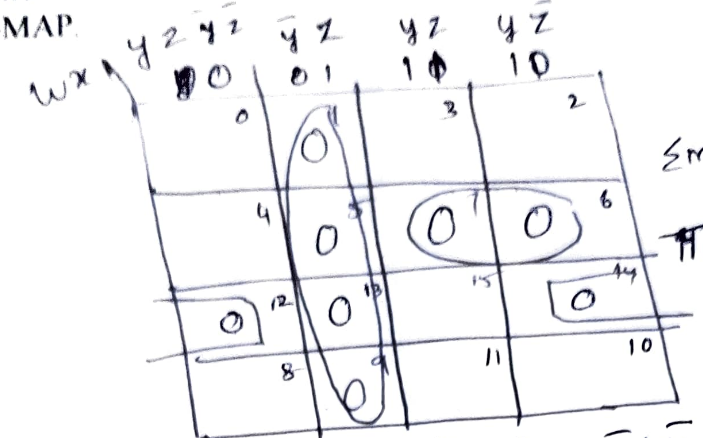
$= xy'z + x'y(z+z') + x'(y+y')z + (x+x')yz$

$= xy'z + x'yz + x'yz' + x'yz + x'y'z + xyz + x'yz$

101 011 010 011 001 111 011

$= \boxed{x'y'z + x'yz' + x'yz + x'y'z}$

2 a. Obtain the minimum POS of the function $Y = X'Z' + WYZ + W'Y'Z' + X'Y$ using K-MAP. [7+3] CO1 L2,L3



$\Sigma m = (0,2,3,4,\cancel{5},10,11,15)$

$\Pi M (1,5,7,6,9,12,13,14)$

$Y = (y+\bar{z})(w+\bar{x}+\bar{y})(\bar{w}+\bar{x}+z$

b. Determine the complement of the following function(i) $F = XY' + X'Y$.(ii) $F = X'YZ' + X'Y'Z$.

(I) F=(X'+Y)(X+Y')

(II) (II) (X+Y'+Z)(X+Y+Z')

3 Design a BCD to Excess-3 Code converter and obtain the expression for excess -3 code using K-MAP and draw the circuit. [10] CO2 L2,L3

A code converter is a circuit that makes the two systems compatible

even though each uses a different binary code.

- Since each code uses four bits to represent a decimal digit, there must be four input variables and

four output variables. We designate the four input binary variables by the symbols A, B, C, and D,

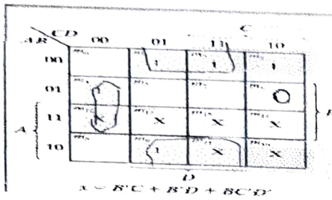
and the four output variables by w, x, y , and z .

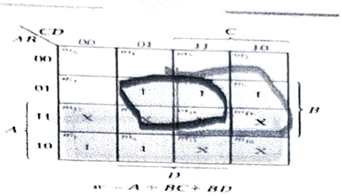
- ADD 3 to BCD to get Excess -3 Code

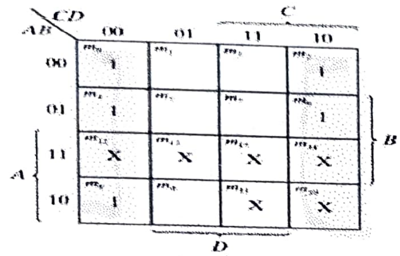
Truth Table for Code Conversion Example

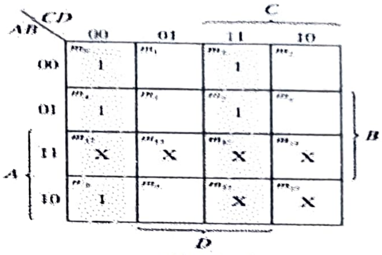
Input BCD				Output Excess 3 Code			
A	B	C	D	w	x	y	z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

Note that four binary variables may have 16 bit combinations, but only 10 are listed in the truth table. The six bit combinations not listed for the input variables are don't- care combinations


$x = B'C + B'D + BC'D'$

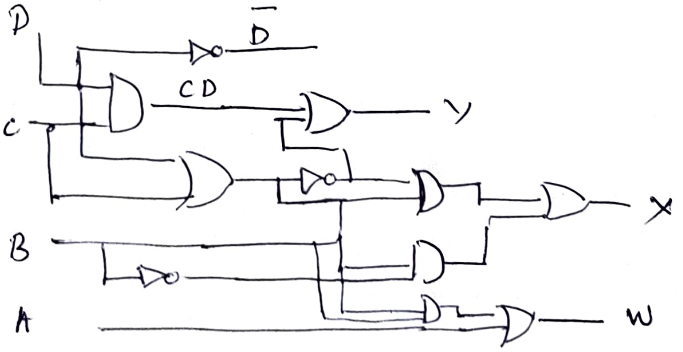

$w = A + BC + BD$


$z = D'$


$y = CD + C'D'$

implemented with three or more levels of gates:

$$z = D'$$
$$y = CD + C'D' = CD + (C + D)'$$
$$x = B'C + B'D + BC'D' = B'(C + D) + BC'D'$$
$$= B'(C + D) + B(C + D)'$$
$$w = A + BC + BD = A + B(C + D)$$



4 a Explain a 4-bit adder subtractor with the circuit diagram?

[5+5] CO2 L2,L3

The addition and subtraction operations can be combined into one circuit with one common binary adder by including an exclusive-OR gate with each full-adder.

The mode input M controls the operation as the following:

If M=0 → the circuit is an adder

M=1 → the circuit becomes a subtractor

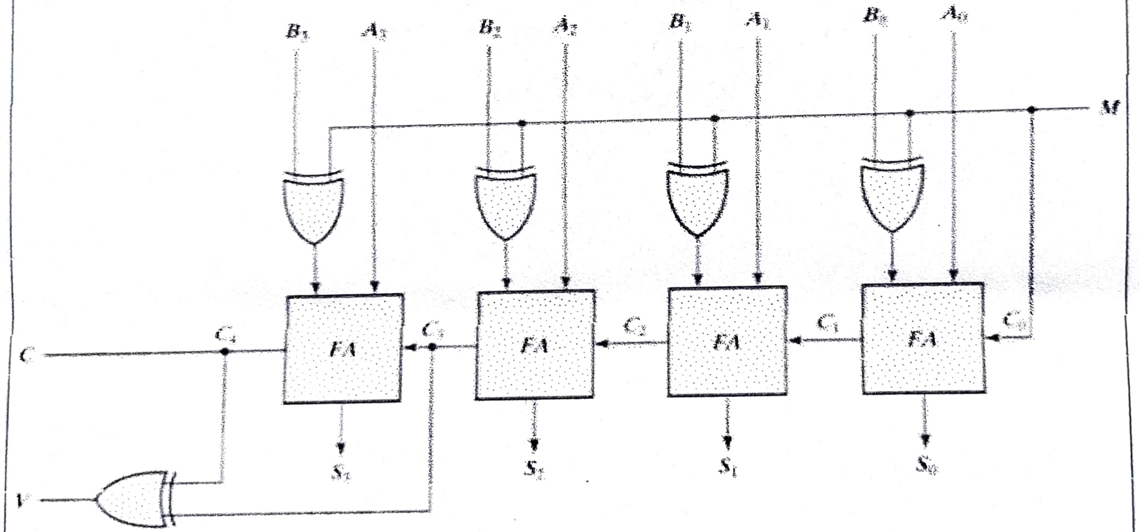
Each exclusive-OR gate receives input M and one of the inputs of B.

value of B, the input carry is C0 = 0and the circuit performs A + B

- When M = 1, we have B XOR 1 = $\overline{\text{B}}$and C0 = 1. The B inputs are all complemented and a 1 is added through the input carry. The circuit performs the operation A + (the 2's complement of B)= A –B

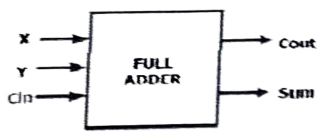
- The exclusive-OR with output V is for detecting an overflow.

The binary adder–subtractor circuit with outputs C and V is shown in Fig. If the two binary numbers are considered to be unsigned, then the C bit detects a carry after addition or a borrow after subtraction. If the numbers are considered to be signed, then the V bit detects an overflow. If V = 0 after an addition or subtraction, then no overflow occurred and the n -bit result is correct. If V = 1, then the result of the operation contains n + 1 bits, but only the rightmost n bits of the number fit in the space available, so an overflow has occurred. The 1n + 12 th bit is the actual sign and has been shifted out of position

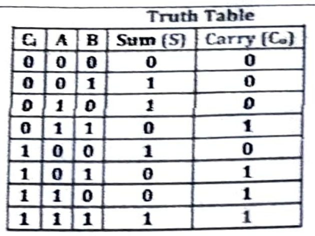


b. Implement a full adder circuit with truth table, circuit diagram and expression for sum and carry.

Logic Diagram



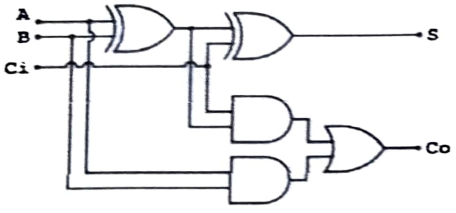
Truth Table



C_i	A	B	Sum (S)	Carry (C_o)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Circuit Diagram

USING BASIC AND XOR GATES

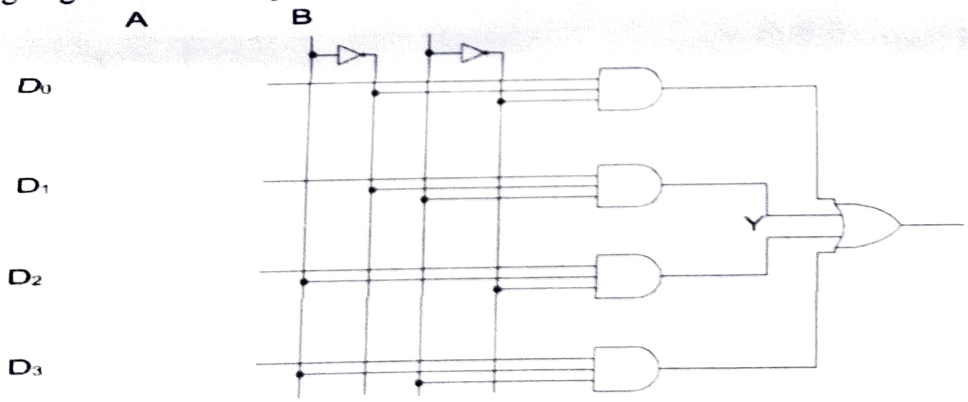


5.a What is a multiplexer? Design a 4*1 mux write truth table and draw the logic circuit and give the expression for output? [6+4] CO2 L1,L2

Multiplexer is a data selector ,a circuit with many inputs

but only one output.

Designing of 4 to 1 multiplexer shown below:



Truth table for 4 to 1 MUX:

A	B	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

Working principle of 4 to 1 multiplexer:

From the above diagram, the Logic Equation for 4 to 1 multiplexer is

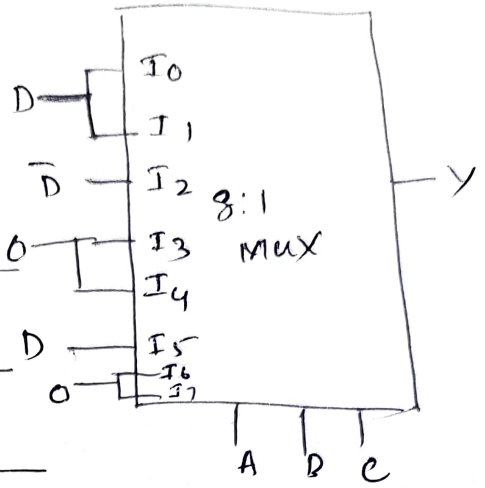
$Y = A'B'D_0 + A'BD_1 + AB'D_2 + ABD_3$

$If\ A = 0,\ B = 0\ then,\ Y = 0'0'D_0 + 0'0D_1 + 00'D_3 + 00D_3 = 1.1.D_0 + 1.0.D_1 + 0.1.D_2 + 0.0.D_3 = D_0$

Similarly, if A=0 and B=1 then Y=D_1 , if A=1 and B=0 then Y=D_2 and

Implement F (A, B, C,D) =∑ (1,3,4,11,12,13,14,15) using a 8:1 MUX?

A B C D	Y
0 0 0 0	D
0 0 0 1	
0 0 1 0	D
0 0 1 1	
0 1 0 0	$\overline{D}$
0 1 0 1	
0 1 1 0	0
0 1 1 1	
1 0 0 0	0
1 0 0 1	
1 0 1 0	D
1 0 1 1	
1 1 0 0	1
1 1 0 1	
1 1 1 0	1
1 1 1 1	



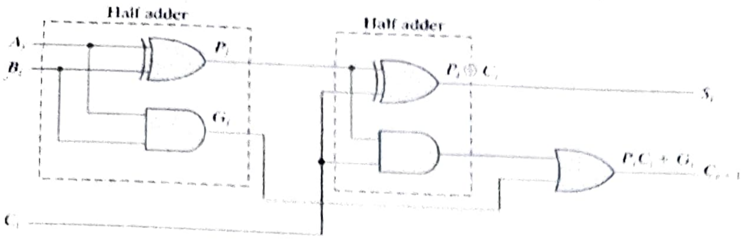
6. Explain the carry look ahead adder with expressions for carry generate and carry propagate and circuit diagram? [10]

CO2 L2,L3

Input carry C_3 does not settle to its final value until C_3 is available from the previous stage.

Similarly, C_2 has to wait for C_1 and so on down to C_0. Thus, *only after the carry propagates* and ripples through all stages will the last output S_3 and carry C_4 settle to their final correct value.

The problem faced by a 4 bit ripple adder is the delay There are several techniques for reducing the carry propagation time in a parallel adder.



Consider the circuit of the full adder s in If we define two new binary variables

$\mathbf{P i = A i \oplus B i G i = A i B i}$

❖ The output sum and carry can respectively be expressed as

$\mathbf{S i = P i \oplus C i C i + 1 = G i + P i C I}$

G i is called a carry generate, and it produces a carry of 1

when both A i and B i are 1, regardless of the input

carry C i .

G i indicates that the data into stage i generates a carry into stage i + 1.

P i is called a carry propagate, because it determines whether a carry into stage i will propagate into stage i + 1 (i.e., whether an assertion of C i will propagate to an assertion of C i + 1).

We now write the Boolean functions for the carry outputs substitute the value of each C i from the previous equations:

C 0 = input carry C 1 =

$G 0 + P 0 C 0$

$C 2 = G 1 + P 1 C 1$

$= G 1 + P 1 (G0 + P 0 C 0)$

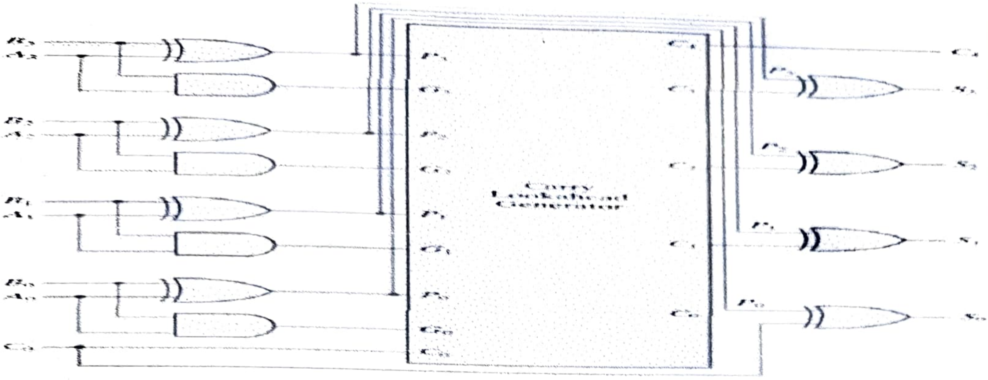
$= G 1 + P 1 G 0 + P 1 P 0 C 0$

$C 3 \quad = G 2 + P 2 C 2$

$= G 2 + P2 G 1 + P 2 P 1 G 0 + P 2 P 1 P 0 C 0$

Since the carry of the different stages depends on only CO we are able to execute the addition in a much faster way.

Circuit of carry look ahead adder:

	 B3 A3 B2 A2 B1 A1 B0 A0 C0 P3 G3 P2 G2 P1 G1 P0 G0 C0 Carry Lookahead Generator C4 C3 C2 C1 C4 P3 S3 P2 S2 P1 S1 P0 S0			

CI CCI /Anchor Teacher HOD