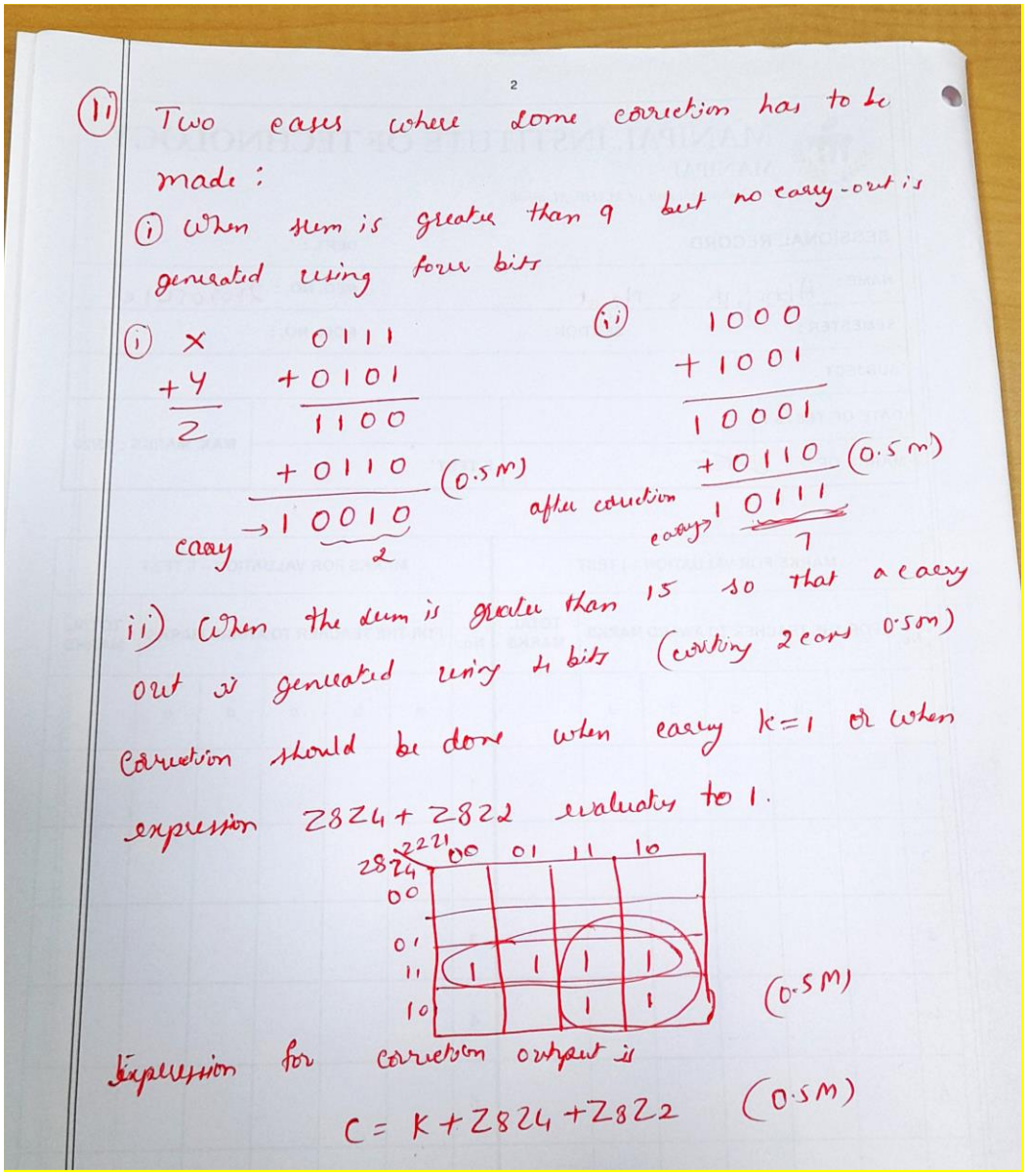
Q11. With example show the two cases where correction has to be made during the addition of two BCD digits. Also show the result after correction. Using four bit binary adders, design the single digit BCD adder by deriving the expression for the above correction. (4)



(11) Two cases where some correction has to be made:

(i) When sum is greater than 9 but no carry-out is generated using four bits

(i) X + 4 = 2

0111 + 0101 = 1100

+ 0110 (0.5M)

carry → 10010 = 2

(ii) 1000 + 1001 = 10001

+ 0110 (0.5M)

after correction carry 1 0111 = 7

ii) When the sum is greater than 15 so that a carry out is generated using 4 bits (writing 2 cases 0.5M)

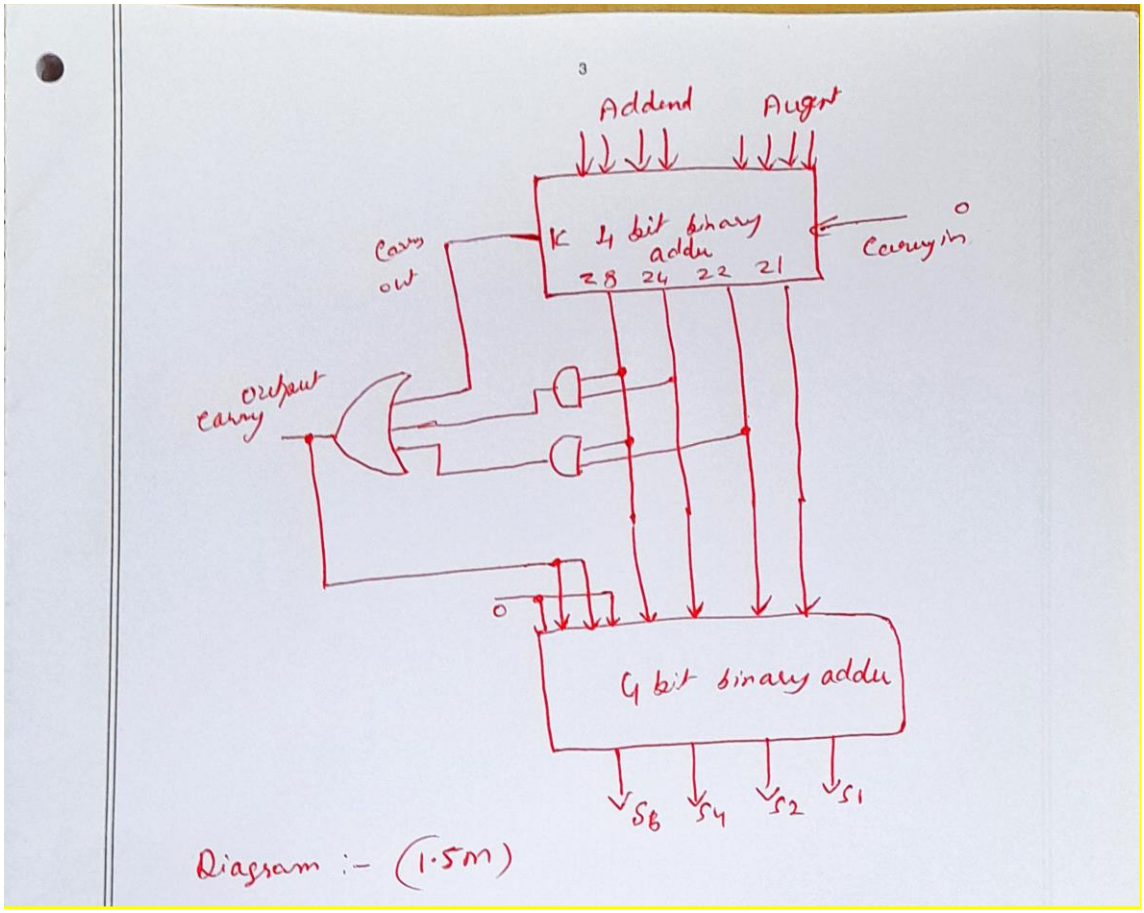
Correction should be done when carry k=1 or when expression Z8Z4 + Z8Z2 evaluates to 1.

Z8Z4 \ Z2Z1	00	01	11	10
00				
01				
11	1	1	1	1
10			1	1

(0.5M)

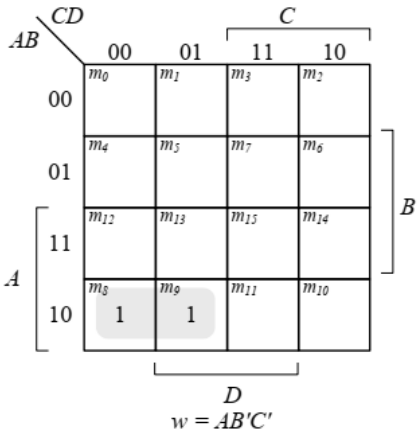
Expression for correction output is

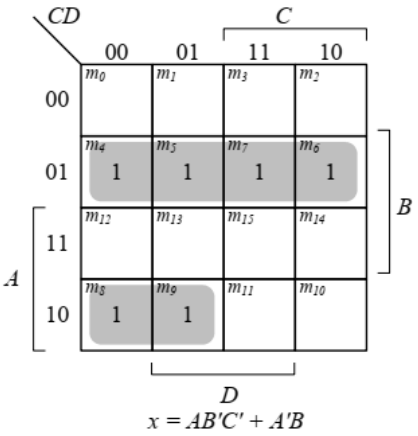
C = K + Z8Z4 + Z8Z2 (0.5M)

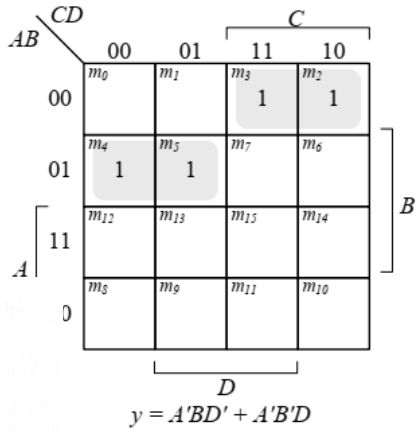


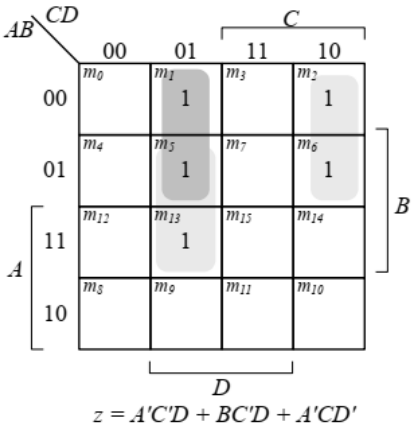
Q12. Design a logic diagram that converts single digit BCD into its equivalent gray code (4)

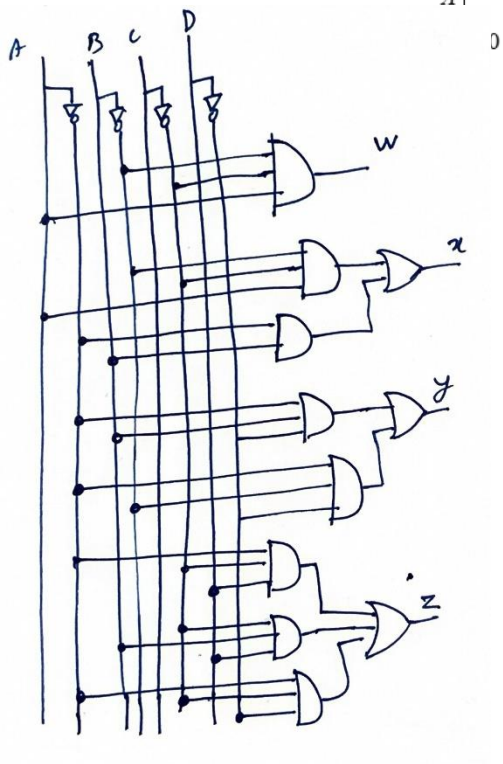
8421 ABCD	Gray wxyz
0000	0000
0001	0001
0010	0011
0011	0010
0100	0110
0101	0111
0110	0101
0111	0100
1000	1100
1001	1101











Logic diagram: 1M

TT: 1M

Each K-MAP: 0.5M*4=2M

Q13. Identify whether the overflow is there or not after performing the 2's complement addition for the following numbers

a) +7 + (+2) b) +7 + (-2) (3)

$$\begin{array}{r} (+7) \\ +\ (+2) \\ \hline (+9) \end{array} \qquad \begin{array}{r} 0\,1\,1\,1 \\ +\ 0\,0\,1\,0 \\ \hline 1\,0\,0\,1 \end{array}$$

$c_4 = 0$

$c_3 = 1$

1M

$$\begin{array}{r} (+7) \\ +\ (-2) \\ \hline (+5) \end{array} \qquad \begin{array}{r} 0\,1\,1\,1 \\ +\ 1\,1\,1\,0 \\ \hline 1\,0\,1\,0\,1 \end{array}$$

$c_4 = 1$

$c_3 = 1$

1M

a) Overflow exists as C4 XOR C3 =1 0.5M b) No overflow as C4 XOR C3 =0 0.5M

Q14. Design a full adder using basic logic gates starting from the truth table. Write the Verilog code for full adder. (3)

c_i	x_i	y_i	c_{i+1}	s_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

(a) Truth table

c_i \ x_iy_i	00	01	11	10
0		1		1
1	1		1	

$s_i = x_i \oplus y_i \oplus c_i$

c_i \ x_iy_i	00	01	11	10
0			1	
1		1	1	1

$c_{i+1} = x_iy_i + x_ic_i + y_ic_i$

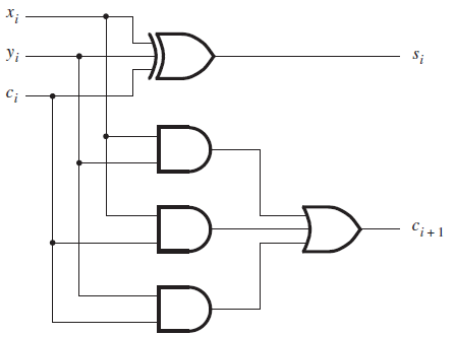
$$s_i = \overline{x}_i y_i \overline{c}_i + x_i \overline{y}_i \overline{c}_i + \overline{x}_i \overline{y}_i c_i + x_i y_i c_i$$

$$s_i = (\overline{x}_i y_i + x_i \overline{y}_i)\overline{c}_i + (\overline{x}_i \overline{y}_i + x_i y_i)c_i$$

$$= (x_i \oplus y_i)\overline{c}_i + \overline{(x_i \oplus y_i)}c_i$$

$$= (x_i \oplus y_i) \oplus c_i$$

$$s_i = x_i \oplus y_i \oplus c_i$$



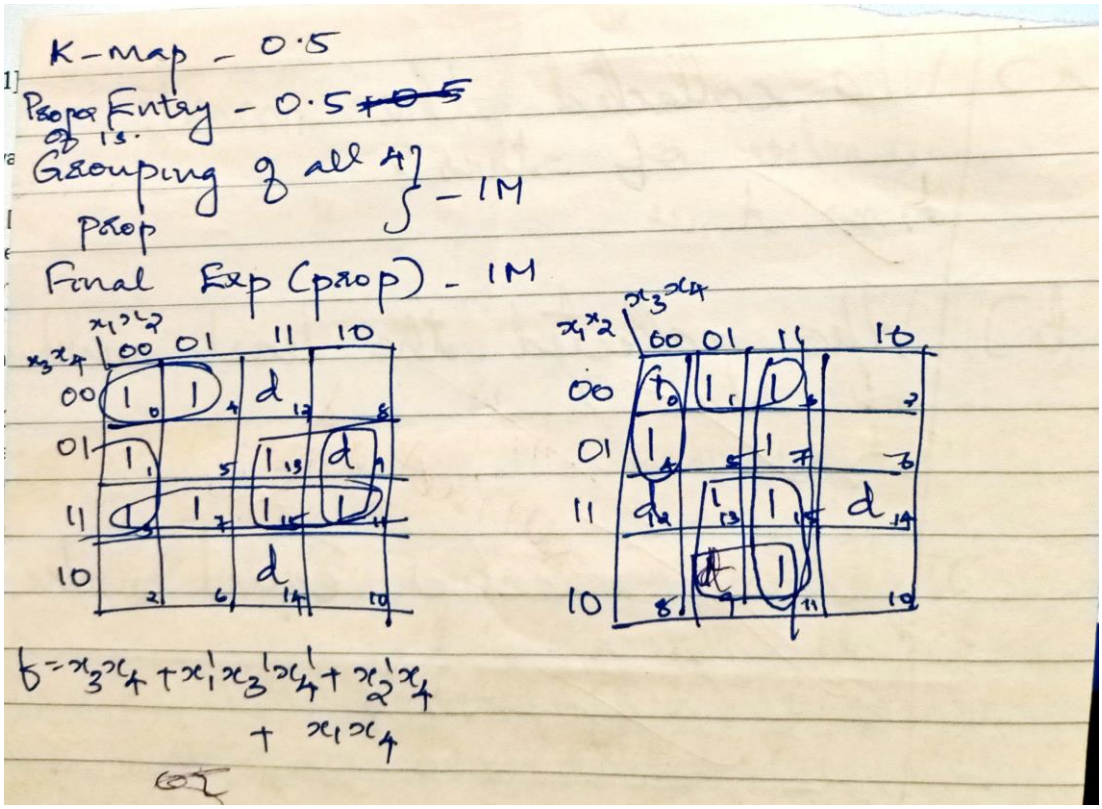
(c) Circuit

TT – 1M

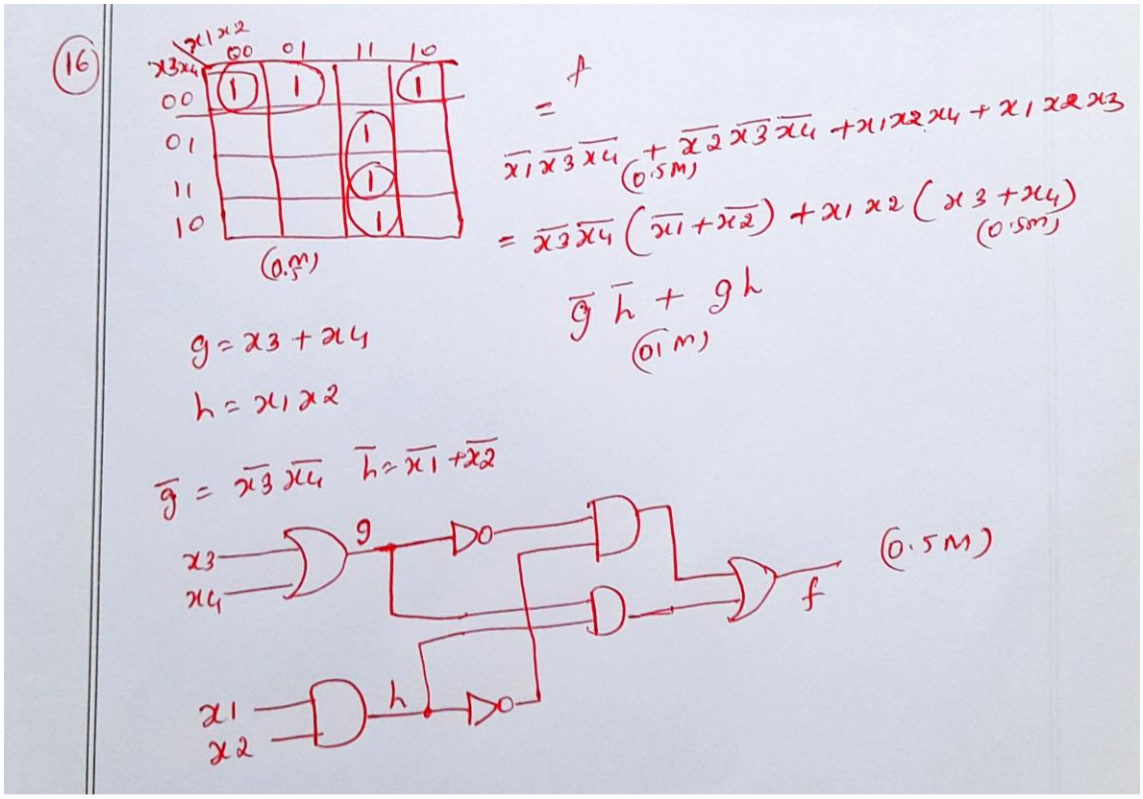
Diagram and expressions- 1M

Verilog ode 1M

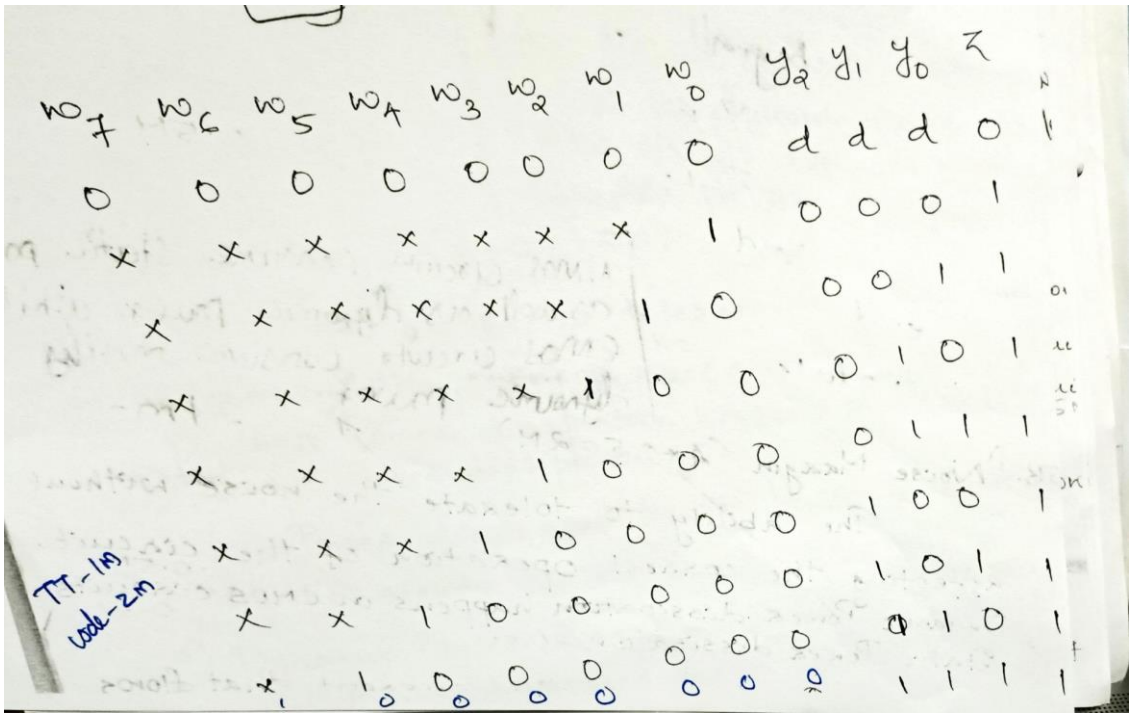
Q15. Simplify the function f (x1,..., x4) = x1'x3'x4' + x3x4 + x1'x2'x4 + x1x2x3'x4 using Karnaugh map to obtain the minimum-cost SOP expression assuming that there are also don't-cares defined as D = ∑m (9, 12, 14) (3)

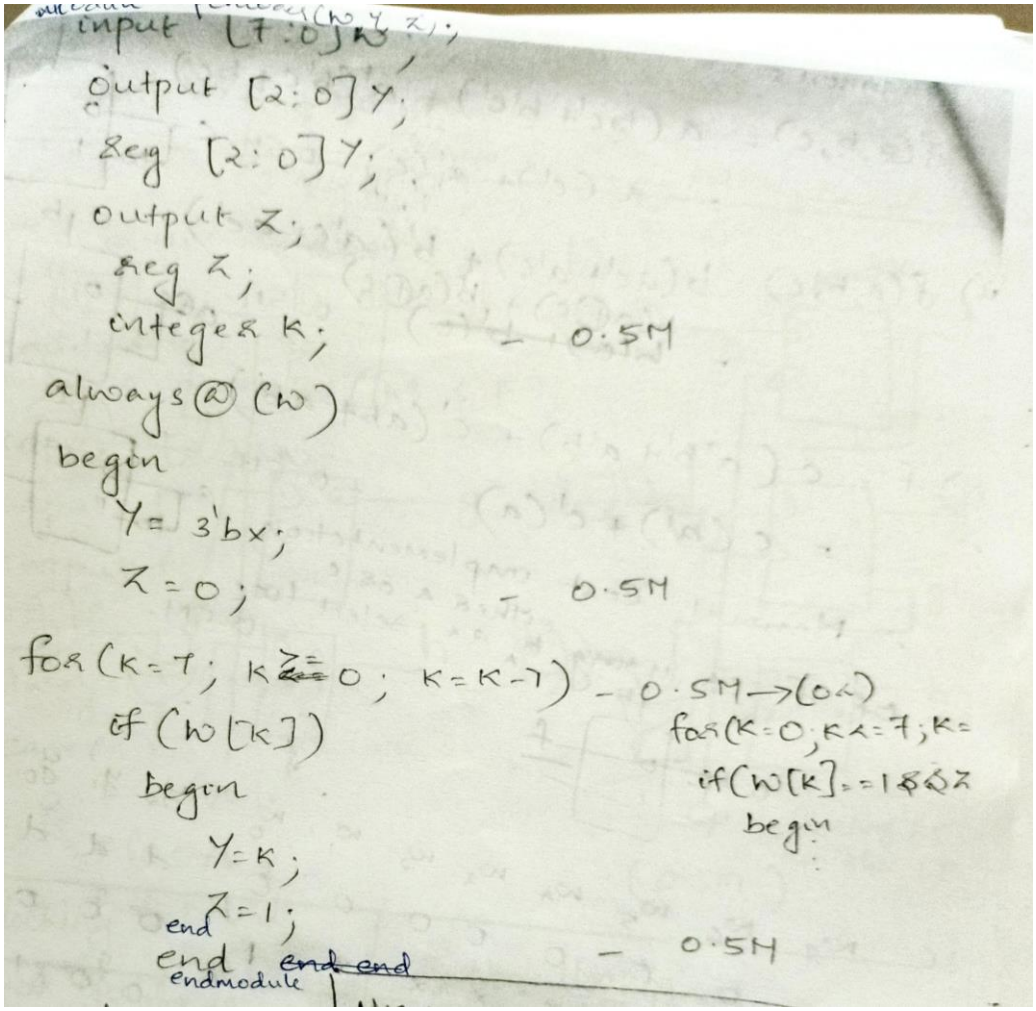


Q16. Find the minimum-cost circuit for the function *f (x1, . . . , x4) = ∑m(0, 4, 8, 13, 14, 15)* using functional decomposition. Assume that the input variables are available in uncomplemented form only. (3)



Q17. Specify the truth table for an 8 to 3 priority encoder. Provide an output 'Z' to indicate that at least one of the inputs is present. The input with the least subscript number has the highest priority. Implement the same by writing the behavioural Verilog code using for loop. (3)





Q18. Design and implement the function f(A,B,C,D)= ∑m(1,3,4,11,12,13,14,15) using 8:1 MUX and other necessary gates. Assume A, B and C as the select signals for 8:1 MUX. (2)