Assignment - 04 Jay Vikrant
EE22BTECH11025

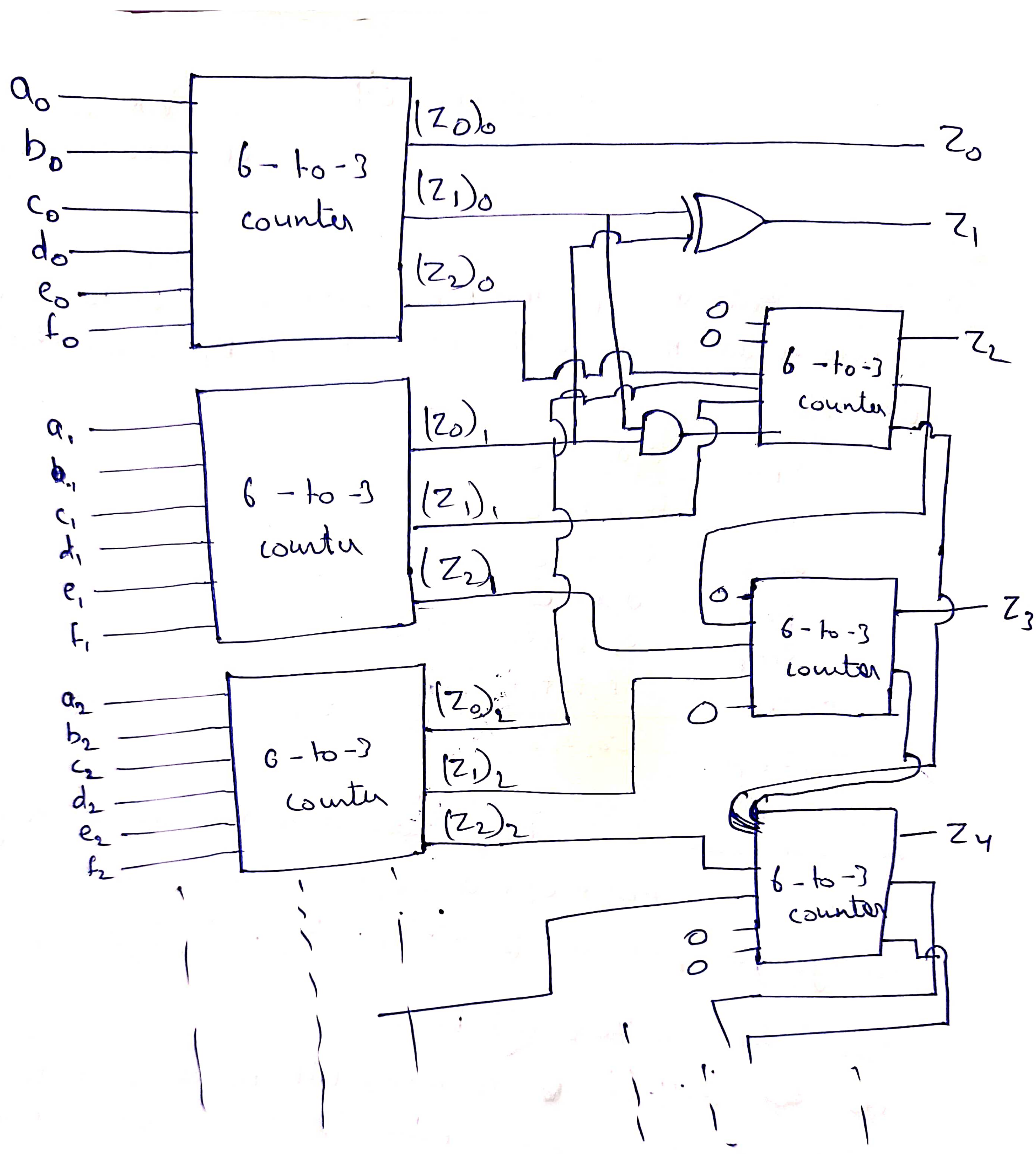
Q3)
a) To generate a 6 operand n-bit adder from 6- to -3 counters, we need to add make a circuit which can perform addition like this.

$$\begin{array}{r}
a_{n-1} \cdots a_3\, a_2\, a_1\, a_0 \\
b_{n-1} \cdots b_3\, b_2\, b_1\, b_0 \\
c_{n-1} \cdots c_3\, c_2\, c_1\, c_0 \\
d_{n-1} \cdots d_3\, d_2\, d_1\, d_0 \\
e_{n-1} \cdots e_3\, e_2\, e_1\, e_0 \\
+\ f_{n-1} \cdots f_3\, f_2\, f_1\, f_0 \\
\hline
z_n\, z_{n-1} \cdots z_2\, z_1\, z_0
\end{array}$$

we can use a 6 - to -3 counter to find sum of every column i.e $a_i + b_i + c_i + d_i + e_i + f_i$, from $i=0$ to $n-1$ we get output of $(z_0)_i, (z_1)_i, (z_2)_i$ for $i=0$ to $n-1$

For $i=0$ i.e $(z_0)_0$ will be z_0
Now $(z_1)_i$ ($i = 1$ to $n-2$) will be supplied to another 6 - to -3 counter having $(z_2)_{i-1}$ ($i = 01$ to $n-2$) and $(z_0)_{i+1}$ ($i = 1$ to $n-2$) as another inputs and in some cases z_1 output of previous 6- to -3 counter and in some cases z_2 output of previous 6 -to -3 counter

The z_0 of this second layer of 6 to 3 counter will be representing our sum

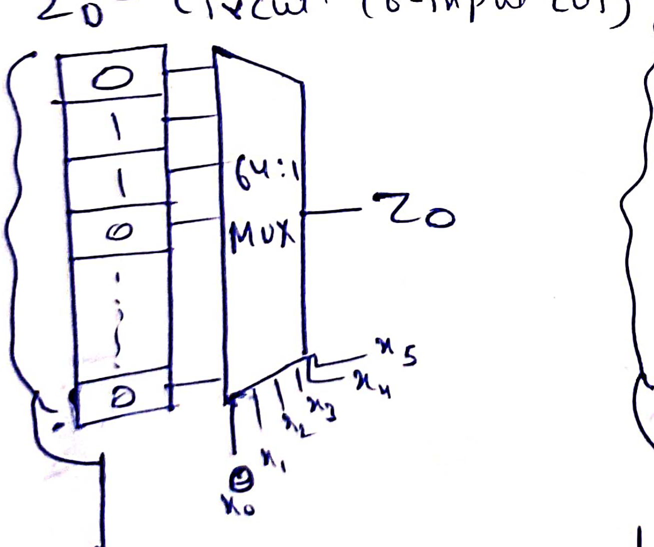


a_0
b_0
c_0
d_0
e_0
f_0
6-to-3 counter
$(Z_0)_0$
$(Z_1)_0$
$(Z_2)_0$
Z_0
Z_1
0
0
6-to-3 counter
Z_2
a_1
b_1
c_1
d_1
e_1
f_1
6-to-3 counter
$(Z_0)_1$
$(Z_1)_1$
$(Z_2)_1$
0
6-to-3 counter
Z_3
0
a_2
b_2
c_2
d_2
e_2
f_2
6-to-3 counter
$(Z_0)_2$
$(Z_1)_2$
$(Z_2)_2$
6-to-3 counter
Z_4
0
0

b) We know the truth table for a 6-to-3 counter we can implement 6-input LUT by using memory table with a 64:1 multiplexer with six select lines as $x_0, x_1, \ldots, x_5$

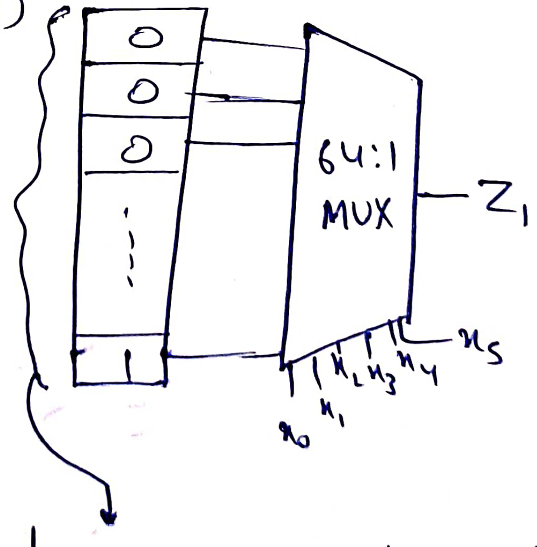
x_0	x_1	x_2	x_3	x_4	x_5	Z_2	Z_1	Z_0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	0	1
0	0	0	0	1	0	0	1	0
⋮								
1	1	1	1	1	1	1	1	0

Z_0 - circuit (6-input LUT)



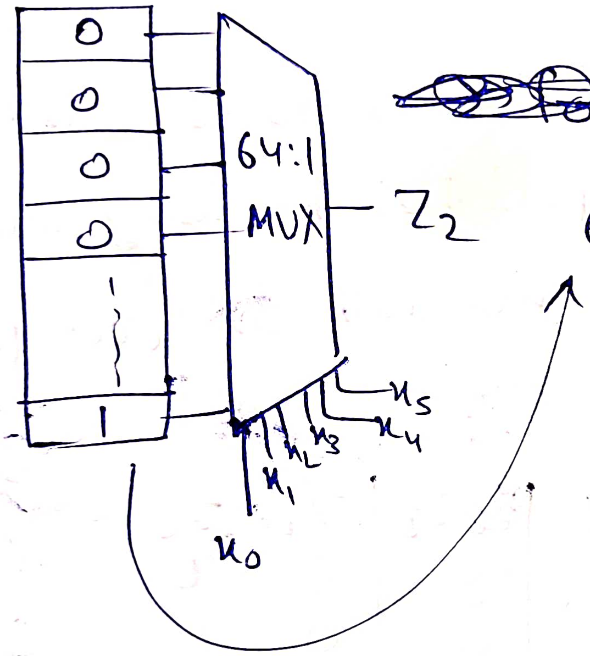
64 Memory elements based on truth table

Z_1 - circuit



64 memory elements based on truth table Z_1

Z_2 - circuit



~~for Z_2 circuit~~

64 memory element based on truth table for Z_2

Q.4)

P.S	N.S $x=0$	$x=1$
A	B,1	H,1
B	F,1	D,1
C	D,0	E,1
D	C,0	F,1
E	D,1	C,1
F	C,1	C,1
G	C,1	D,1
H	C,0	A,1

a) $P_0 = A, B, C, D, E, F, G, H$

$P_1 = (ABEFG)(CDH)$

$P_2 = (AB)(EFG)(CDH)$

$P_3 = (A)(B)(EFG)(CD)(H)$

$$P_1 \begin{array}{l} \xrightarrow{x=0} \begin{array}{cccccccc} A & B & E & F & G & C & D & H \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ B & F & D & C & C & D & C & C \end{array} \\ \xrightarrow{x=1} \begin{array}{cccccccc} A & B & E & F & G & C & D & H \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ H & D & C & C & D & E & F & A \end{array} \end{array}$$

∵ There is no further distinguishable state

∴ The equivalence pattern for the above machine is (A)(B)(EFG)(CD)(H)

b) The standard form of the corressponding reduced machine is

P.S	N.S $x=0$	$x=1$
$(A) \to \alpha$	$(\beta, 1)$	$(\epsilon, 1)$
$(B) \to \beta$	$(\gamma, 1)$	$(\delta, 1)$
$(EFG) \to \gamma$	$(\delta, 1)$	$(\delta, 1)$
$(CD) \to \delta$	$(\delta, 0)$	$(\gamma, 1)$
$(H) \to \epsilon$	$(\delta, 0)$	$(\alpha, 1)$

P.S	N.S x=0	N.S x=1
$\alpha \to$ (A)	(B, 1)	(E, 1)
$\beta \to$ (B)	(C, 1)	(D, 1)
$\gamma \to$ (C)	(D, 1)	(D, 1)
$\delta \to$ (D)	(D, 0)	(C, 1)
$\epsilon \to$ (E)	(D, 0)	(A, 1)

c) As A & B distinguish in P_3, so the length of sequence is 3.

- For going from P_0 to P_1, let input be $x=0$ for A & B
  ∴ $z=1$ & N.S is B, F
- For going from P_1 to P_2, let input be $x=0$ for A & B & F
  $z=1$ & N.S is F, C
- For going from P_2 to P_3, let input be $x=0$ for F & C for both C & F
  Now output z is different for both C & F

∴ Minimum length sequence to distinguish state A, from B is 000

Q5) Let M be a machine with n states namely

$A_1, A_2, A_3, A_4, \ldots, A_{n-1}, A_n$

Making partition of M (such that each partition P_i, state A_i is distinguished from other states)

$P_0 = (A_1, A_2, \ldots, A_{n-1}, A_n)$

$P_1 = (A_1)(A_2, A_3, \ldots, A_{n-1}, A_n)$

$P_2 = (A_1)(A_2)(A_3, A_4, \ldots, A_{n-1}, A_n)$

$P_3 = (A_1)(A_2)(A_3)(A_4, A_5, \ldots, A_{n-1}, A_n)$

$\vdots \qquad \vdots \qquad \vdots \qquad \vdots$

$P_{n-2} = (A_1)(A_2) \ldots (A_{n-2})(A_{n-1}, A_n)$

$P_{n-1} = (A_1)(A_2) \ldots (A_{n-1})(A_n)$

there are n states in P_{n-1} which represents the equivalence pattern.

Now, the state A_{n-1} & A_n are distinguishable by sequence of length n-1 which are the last two states to be distinguishable as all other states are distinguished much before

∴ Two states S_i & S_j of a n-state machine M are distinguishable by a sequence of length n-1 or less.

Q.6) X (Input) : 0 0 0 0 1 0 1 0 0 0 1 0
Z (Output) : 1 0 1 0 0 1 1 0 0 0 0 1

As the machine is one-input & has 3-states
As the machine is initially in state A
when A is given X=0 as input, then it must go to a new state B as for same input, output is different

when B is given X = 0 as input, then it must go to a different state either back to A or to a new state C

let it goes back to A

Again after 00 as two consecutive input to A ; machine again comes back to A.
when X = 1 is given as input at A, machine can either goes to C or stay at A.
lets consider it stays at A
for X = 0 to A, N.S is B
when X = 1 is given to B, it goes to N.S C
when X = 0 is given to C, it stay at C.
also, the next two input & output are same, N.S C.
when X = 1 is given to C, it goes to A
when X = 0 is given to A, it goes to B

Thus, our state diagram look like

$$\begin{tikzcd}
A \arrow[loop left, "1/0"] \arrow[r, bend left, "0/1"] & B \arrow[l, bend left, "0/0"] \arrow[r, bend left, "1/1"] & C \arrow[loop right, "0/0"] \arrow[ll, bend left=50, "1/0"]
\end{tikzcd}$$

∴ Reduced standard form,

P.S	N.S	
	X = 0	X = 1
A	B, 1	A, 0
B	A, 0	C ~~0~~, 1
C	C, 0	A, 0