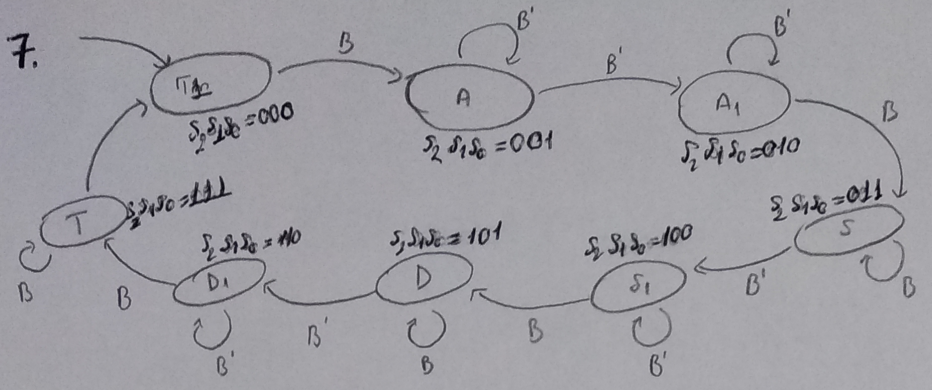
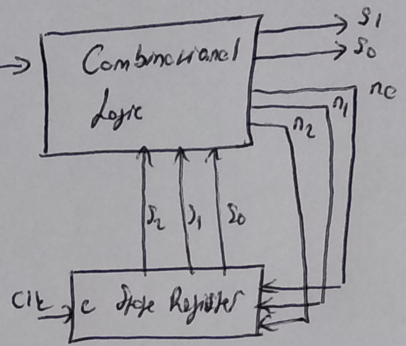
7.



Step-1

Step-2 A, B

- 3 bit state register (8 state)
- Output s_1, s_0
- Next state n_2, n_1, n_0



Step 2C

s_2	s_1	s_0	B	n_2	n_1	n_0	s_1	s_0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0
0	0	1	0	0	1	0	0	1
0	0	1	1	0	0	1	0	1
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0
0	1	1	1	0	1	1	1	0
1	0	0	0	1	0	0	1	0
1	0	0	1	1	0	1	1	0
1	0	1	0	1	0	0	1	[illegible]
1	0	1	1	1	1	0	1	1
1	1	0	0	1	1	0	1	1
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	0	0

Step-2D

$$n_2 = s_2' s_1 s_0 B' + s_2 s_1' + s_2 s_0' + s_2 B$$

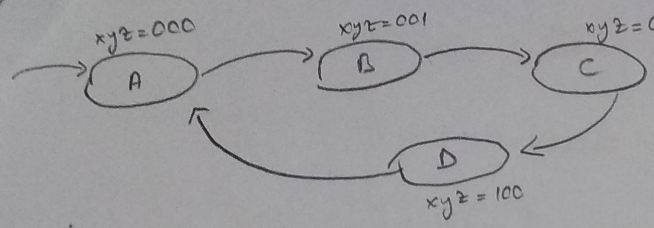
$$n_1 = s_1 s_0' + s_1 B + s_2 s_0 B + s_2' s_1' s_0 B'$$

$$n_0 = s_0' B + s_2' B + s_1 B + s_2 s_1' s_0 B'$$

$$s_1 = s_2 s_0' + s_2 s_1' + s_2' s_1 s_0$$

$$s_0 = s_1 \text{ XOR } s_0$$

6.)

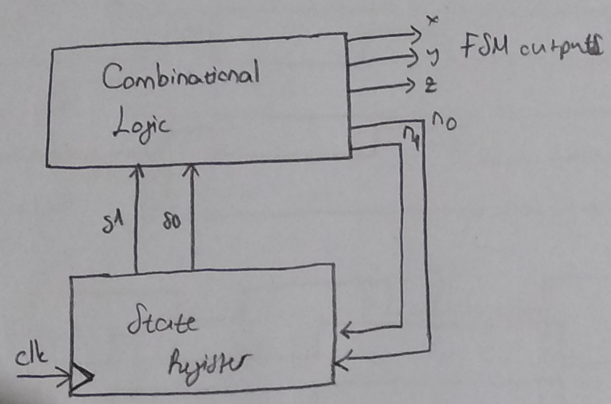


$\Rightarrow$ FSM Step-1

Step-2 A-B

$\Rightarrow$ 2 bit state register (4 state "A,B,C,D")

- Output x, y, z
- Next state signals n1, n0



Step 2-C

Input		Output				
S_1	S_0	x	y	z	n_1	n_0
0	0	0	0	0	0	1
0	1	0	0	1	1	0
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Step-2D

Implement Combinational Logic

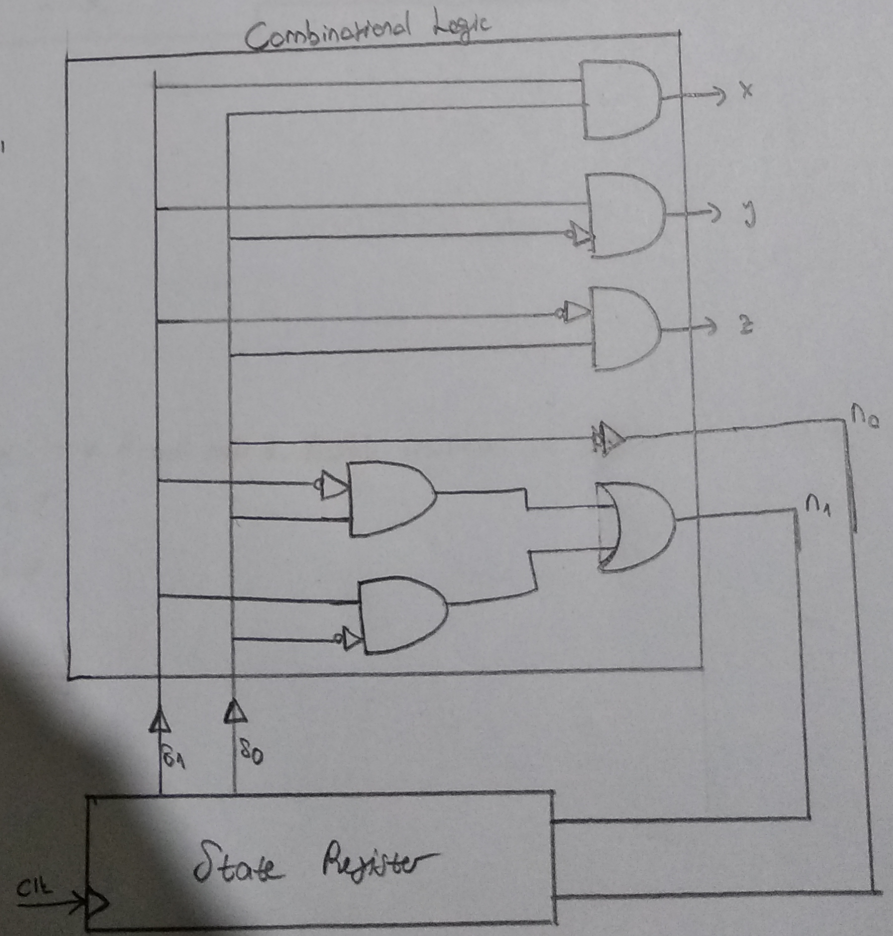
$n_0 = S_1'S_0' + S_1S_0' = S_0'(S_1 + S_1') = S_0'$

$n_1 = S_1'S_0 + S_1S_0'$

$z = S_1'S_0$

$y = S_1S_0'$

$x = S_1S_0$



1 -

a) $P = 1/50\text{khz} = 1/50\times10^3 s = 1/50\ ms = 0.02\ ms = 20\ \mu s$

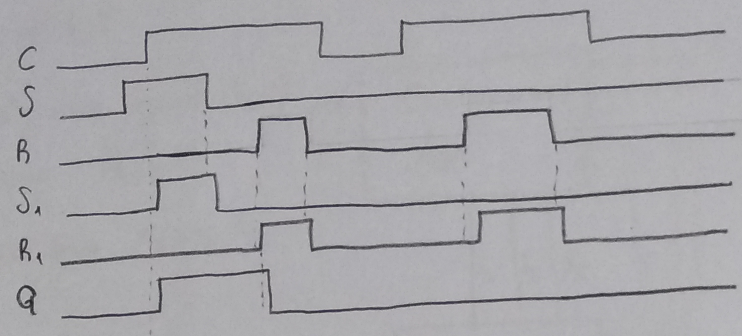
b) $P = 1/300\text{MHz} = 1/300\times10^6 s = 1/300\ \mu s = 3.33\times10^{-3}\ \mu s = 3.33\ ns$

c) $P = 1/3.4\text{GHz} = 1/3.4\times10^9 s = 1/3.4\ ns = 0.294\ ns = 294\ ps$

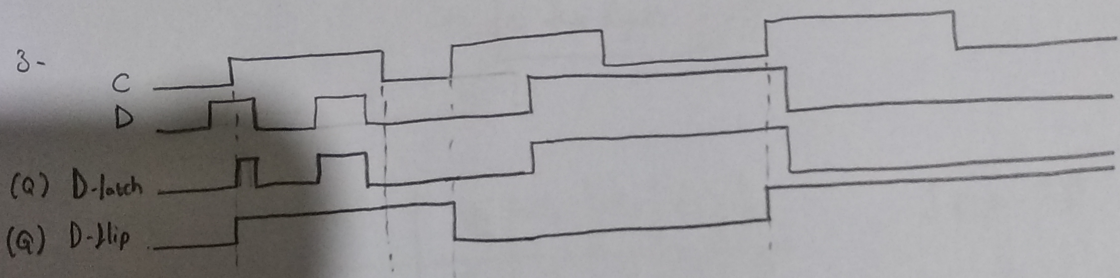
d) $P = 1/10\text{GHz} = 1/10\times10^9 s = 1/10\ ns = 0.1\ ns = 100\ ps$

c) $P = 1/1\text{THz} = 1/10^{12} s = 1\ ps$

2 -



3 -



4 - a) Number of bits = $\log_2 4 = 2$

b) Number of bits = $\log_2 8 = 3$

c) " $= \log_2 9 = 3.17 \Rightarrow 4$

d) " $= \log_2 23 = 4.52 \Rightarrow 5$

e) " $= \log_2 900 = 9.8 \Rightarrow 10$

5 -) Assume that there are two states, state A and state B. Possible transitions are s-A to s-B and s-B to s-A.

Thus the possible transitions $2\times1 = 2$

So possible transitions $n.(n-1)$