

①

cd \ ab	00	01	11	10
00	1			1
01		1	1	1
11		d		
10	d			1



1			1
	1	1	1
	d		
d			1

$$1) f(a, b, c, d) = b' \cdot d' + c' \cdot d \cdot a + a' \cdot b \cdot d$$

1			1
	1	1	1
	d		
d			1

$$2) f(a, b, c, d)$$

$$= b' \cdot d' + c' \cdot d \cdot b + a \cdot c' \cdot d$$

1			1
	1	1	1
	a		
d			1

$$3) f(a, b, c, d)$$

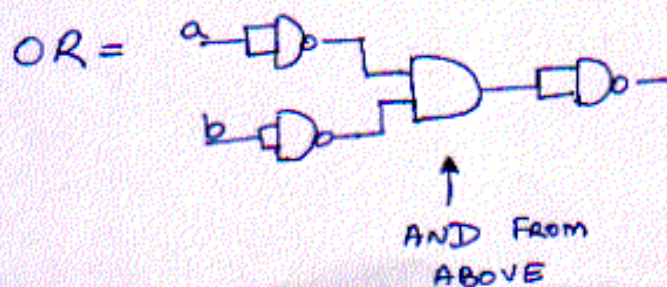
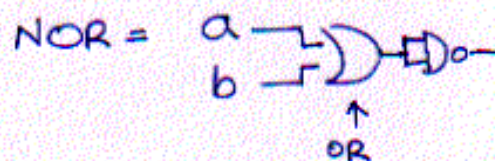
$$= b' \cdot d' + c' \cdot d \cdot b + c' \cdot a \cdot b'$$

III
a)

A set of gates is said to be universal if any combinational system can be implemented using gates from that set. So for example $\{ \text{NAND} \}$ is a universal set. Any of $\{ \text{AND}, \text{OR}, \text{NOT}, \text{NAND}, \text{NOR}, \text{XOR}, \text{XNOR} \}$ can be implemented using only NAND gates.

b) #) We need to see if the following gates can be implemented in each of the cases:
 $\{ \text{AND}, \text{OR}, \text{NOT}, \text{NAND}, \text{NOR}, \text{XOR}, \text{XNOR} \}$

i) $\{ \text{NAND}, \text{NOR} \}$



b i) $xor = a' \cdot b + a \cdot b'$

This can be implemented using the AND, NOT & OR gates seen earlier.

Thus this set $\{NAND, NOR\}$ is universal.

ii) $\{xor, NOT\}$

a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

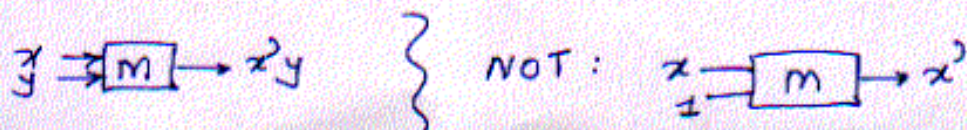


We see that any modifications to XOR, only yields a NOT gate. Inverting XOR, leaves us with another combination of AND's & OR's -

$$\begin{aligned}(a \oplus b)' &= (a' \cdot b + a \cdot b')' \\ &= (a + b') \cdot (a' + b)\end{aligned}$$

Hence this set is NOT universal.

iii) The single gate we have here is $f(xy)$
i.e let us denote this module as



AND: $x \xrightarrow{\text{NOT}} \quad y \xrightarrow{\quad} \boxed{M} \rightarrow x \cdot y$

NAND: Since we have AND & NOT we have NAND

OR: $x \xrightarrow{\text{NOT}} \quad y \xrightarrow{\text{NOT}} \left. \begin{array}{l} \text{ } \\ \text{ } \end{array} \right\} \text{AND} \rightarrow \text{NOT} \rightarrow x + y$

Since NOT & AND have been seen, we can do OR

At this point we could conclude that all gates can be derived since we have AND, NAND, OR & NOT, which form a universal set in more than 1 way.

IV) $x \xrightarrow{\quad} y \xrightarrow{\quad} \boxed{M} \rightarrow (x+y)z'$

NOT: $1 \xrightarrow{\quad} y \xrightarrow{\quad} \boxed{M} \rightarrow z'$

OR: $x \xrightarrow{\quad} y \xrightarrow{\quad} \boxed{M} \rightarrow (x+y)$

NOR: Since we have OR & NOT, we have NOR.

Since $\{NOR\}$ is a universal set, $\boxed{M}$ is

2. $f(a,b,c) = \sum m(0,1,2) + \sum d(6,7)$

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	d
1	1	1	d

Getting the 1's terms. and 'don't cares'

0	0	0	0
1	0	0	1
2	0	1	0
6	1	1	0
7	1	1	1

$\Rightarrow$

(0,1)	00-
(0,2)	0-0
(2,6)	-10

notice we'll skip (6,7) since they are both 'don't cares'.

	0	1	2	6	7
(0,1)	x	x			
(0,2)	x		x		
(2,6)			x	x	

So we keep $(0,1)$ & $(0,2)$ Since these two terms cover all the 1's.

$$f = a'c' + a'b'$$

$$\underline{4} \quad Q_0(t+1) = Q_0 \oplus (X'Q_1) \quad \text{--- (1)}$$

$$Q_1(t+1) = Q_1 \oplus (X + Q_0) \quad \text{--- (2)}$$

$$y(t) = Q_1 + Q_0'$$

We get the above two relationship because we know the network is using T-ff. The characteristic equation of T-ff is

$$Q(t+1) = T(t) \oplus Q(t)$$

a) Using first two equations we create the State table.

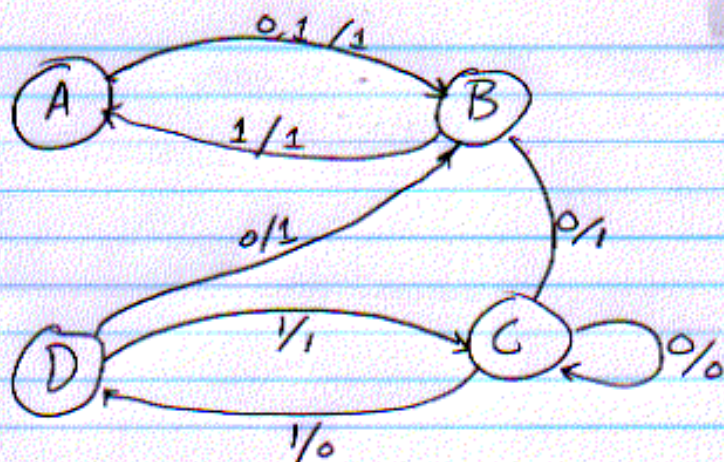
$Q_1 Q_0$	x		$y(t)$
	0	1	
0 0	10	10	1
0 1	~~00~~	00	0
1 0	~~01~~	11	1
1 1	10	01	1

$\Rightarrow$

S	x		
	0	1	
A	B	B	1
B	C	A	0
C	C	D	1
D	B	C	1

Let $A = 00$ $B = 01$ $C = 10$ $D = 11$

b)



* Note in part (a) Q_0, Q_1 are switched to their correct order Q_1, Q_0 . So the solution may look a little ~~bit~~ different from the discussion section on Thursday.

This solution is correct!

5

Implement JK using T flip flop.

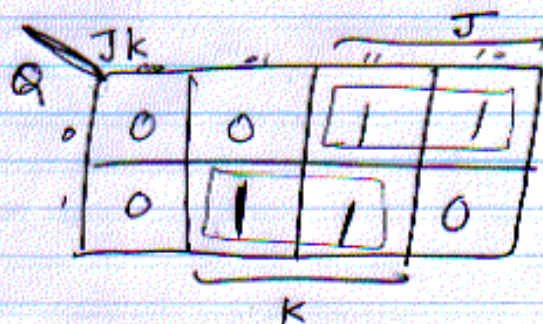
$$Q(t+1) = T(t) \oplus Q(t) = Q(t) \bar{K} + \bar{Q}(t) J$$

XORing both sides with $Q(t)$ we get

$$\cancel{T(t)} = (Q \bar{K} + \bar{Q} J) \oplus Q$$

Since we want a minimal expression using AND-NOT-OR gates, we will minimize the expression using k-maps.

J	K	Q	$Q \oplus \bar{K} + \bar{Q}J$	T (+)
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	0
1	1	0	1	1
1	1	1	0	1



From above k-map we get

$$T = KQ + J\bar{Q}$$

