

ECE124: Discussion

Discussion #5

Yeonsik Noh, PhD

2.15 Simplify the following Boolean functions T1 and T2 to minimum number of literals:

¹	A	B	C	T	T
0	0	0	0	1	0
0	0	0	1	1	0
0	1	0	0	1	0
0	1	1	0	0	1
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	0	0	1

m0 m1 m2

$$t1(abc) = m0 + m1 + m2$$

$$A'B'C' + A'B'C + A'BC'$$

$$A'B'(C' + C) + A'BC'$$

$$A'B' + A'BC'$$

$$A'(B' + BC')$$

$$A'B' + A'C'$$

- For the Boolean function: $F(a, b, c, d) = (c' + d)(b + c')$
- (a) Express the function in sum of minterms and product of maxterms forms.
- (b) Obtain the truth table of F.

A	B	C	D	$F = C' + BD$
0	0	0	0	1
0	0	0	1	0
0	1	0	0	0
0	1	0	1	1
1	0	0	0	0
1	0	0	1	1
1	1	0	0	1
1	1	0	1	1



$abcd$	$abc'd$	$abc'd$
0101	0010	0010
0111	0110	0011
1101	1010	1010
1111	1110	1011

$$f = \pi(2 3 6 10 11 14)$$

$$f = \sum(0 1 4 5 7 8 9 12 13 15)$$

- For the Boolean function: $F(a, b, c, d) = (c' + d)(b + c')$

(c) Use Boolean algebra to simplify the function to a minimum number of literals.

$$F = (c+d)(b+c')$$

$$(c'+d)b + (c'+d)c'$$

$$bc' + bd + c' + c'd$$

$$c'(b+1+d) + bd = c'bd$$

2.19 Express the following function as a sum of minterms and as a product of maxterms:

$$F(A, B, C, D) = B'D + A'D + BD$$

ab'cd

0001

0010

1000

1010

f= sum(1 3 5 7 11

13 15)

pi(0 2 4 6 8 10 12

2.29 Determine whether the following Boolean equation is true or false.

$$x'y' + x'z + x'z' = x'z' + y'z' + x'z$$

$f_1 = x'y' + x'z + x'z'$	$x'yz'$	$xy'z'$	$x'yz$
sum(0 1 2 3)	000	000	001
	010	100	011

2.30 Write the following Boolean expressions in sum of products form and in sum of minterms form.

$$F(a, b, c, d) = (b + d)(a' + b' + c)$$

$f = (b+d)(a'+b'+c)$	abcd	$a'b'cd$
$(b+d)a' + (b+d)b' + (b+d)c$	000	1100
$a'b + a'd + b'd + bc + cd$	0010	1101
	1000	
$f = \pi(0 \ 2 \ 8 \ 10 \ 12 \ 13)$	1010	

* Simplify function $F(x, y) = x'y' + x'y + xy'$ using Boolean algebra

$$f(xy) = x'y' + x'y + xy'$$

$$x(y'+y) + xy'$$

$$x' + xy'$$

$$(x' + x)(x'y')$$

$$x' + y'$$

$$x' + (xy')$$

$$y' + y = 1$$

$$y'y = 0$$

* Simplify function $X(A, B, C) = AB + ABC + AB'C' + AC'$ using Boolean algebra

$$X(ABC) = AB(1+c) + Ac'(b'+1)$$

$$AB+AC'$$

$$A(B+C')$$

$$A+0)(B+c')$$

* Simplify function $F = A'B'C + AB'C + ABC' + ABC$ using Boolean algebra

$$F(ABC) = (A'B+AB')C + AB(C'+C)$$

$$(A'B+AB')C + AB$$

$$(A'B)+A \quad \{A'B+A(B'+B)\}(C+AB)$$

$$((A'+A)(B+A))(C+AB)$$

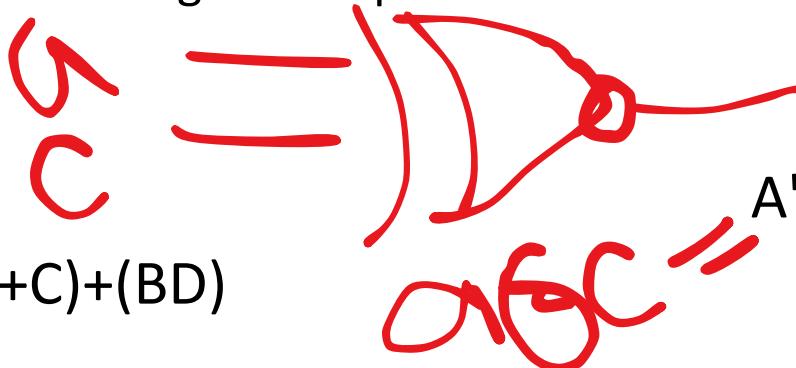
$$(B+A)(C+AB)$$

$$BC+AC+AB$$

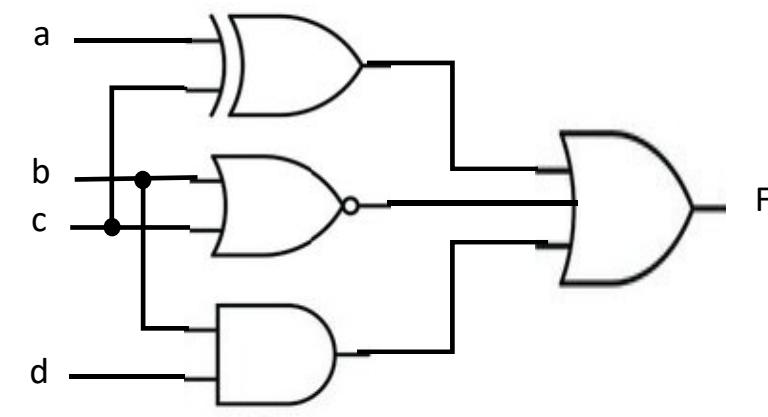
*From the logic diagram,

(a) Express the function F in Boolean expression.

(b) Construct the truth table describing the outputs



$$F = \text{SUM}(0 \ 1 \ 2 \ 3 \ 5 \ 6 \ 7 \ 8 \ 9 \ 12 \ 13 \ 15)$$



a	b	c	d	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

ECE124: Discussion

Discussion #7

Yeonsik Noh, PhD

Minimization via K-Maps

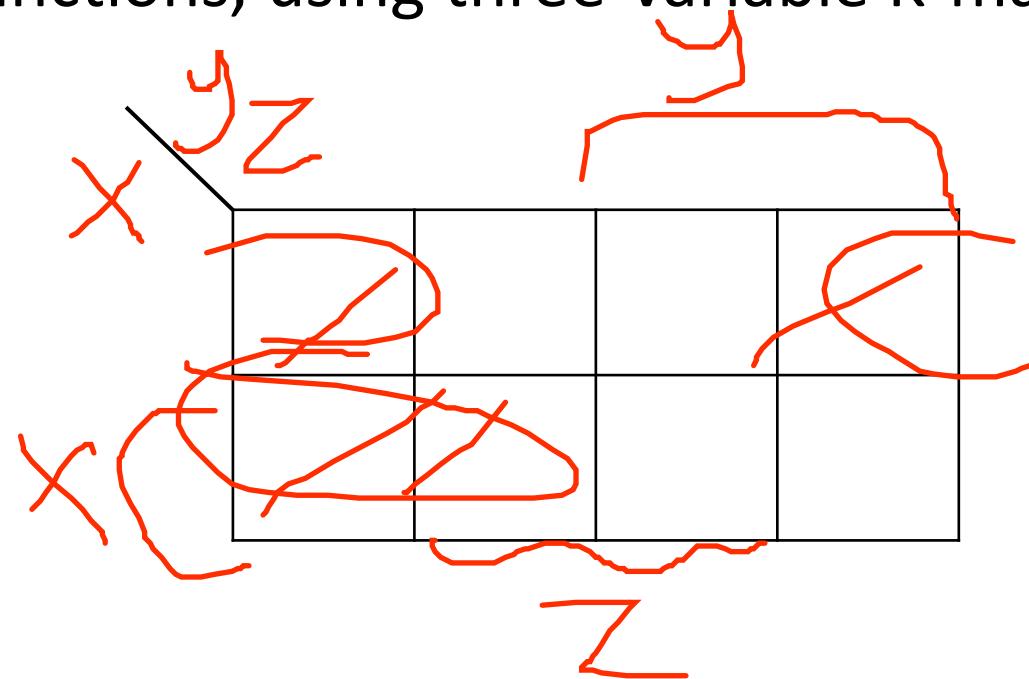
To obtain minimized SOP expression for a Boolean function F...

- Step1: Translate truth table to K-map
- Step2: Cover all 1's in the smallest number of largest blocks
- Step3: Extract one product term for each block
- Step4: Sum of extracted product terms=minimized SOP expression!

3.1 Simplify the following Boolean functions, using three-variable K-maps:

(a) $F(x, y, z) = \sum (0, 2, 4, 5)$

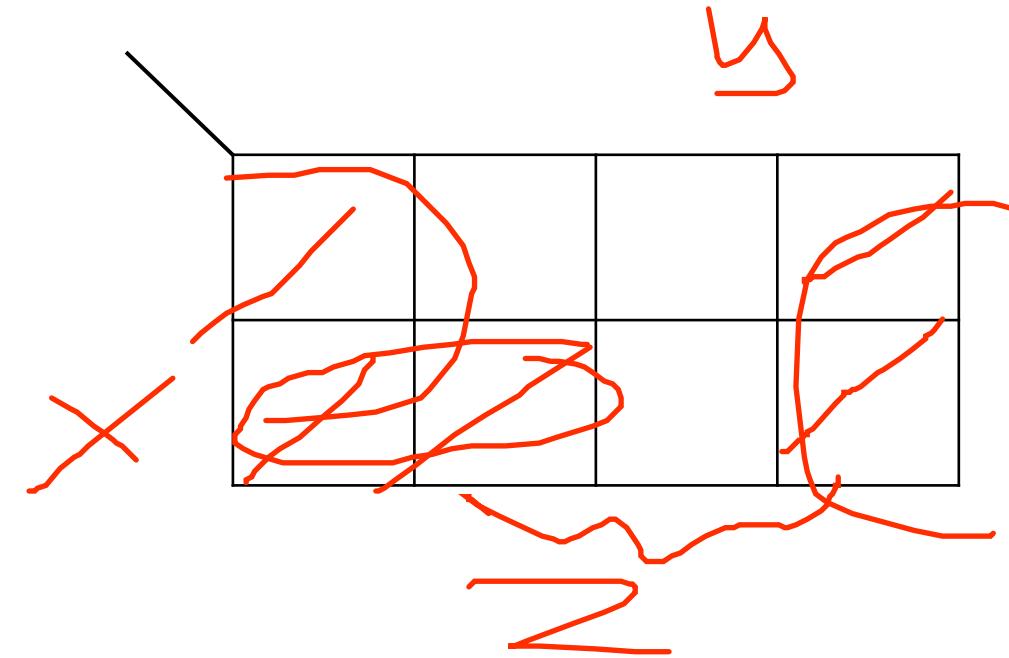
$$\cancel{xy'} + \cancel{xz'}$$



3.1 Simplify the following Boolean functions, using three-variable K-maps:

(b) $F(x, y, z) = \sum (0, 2, 4, 5, 6)$

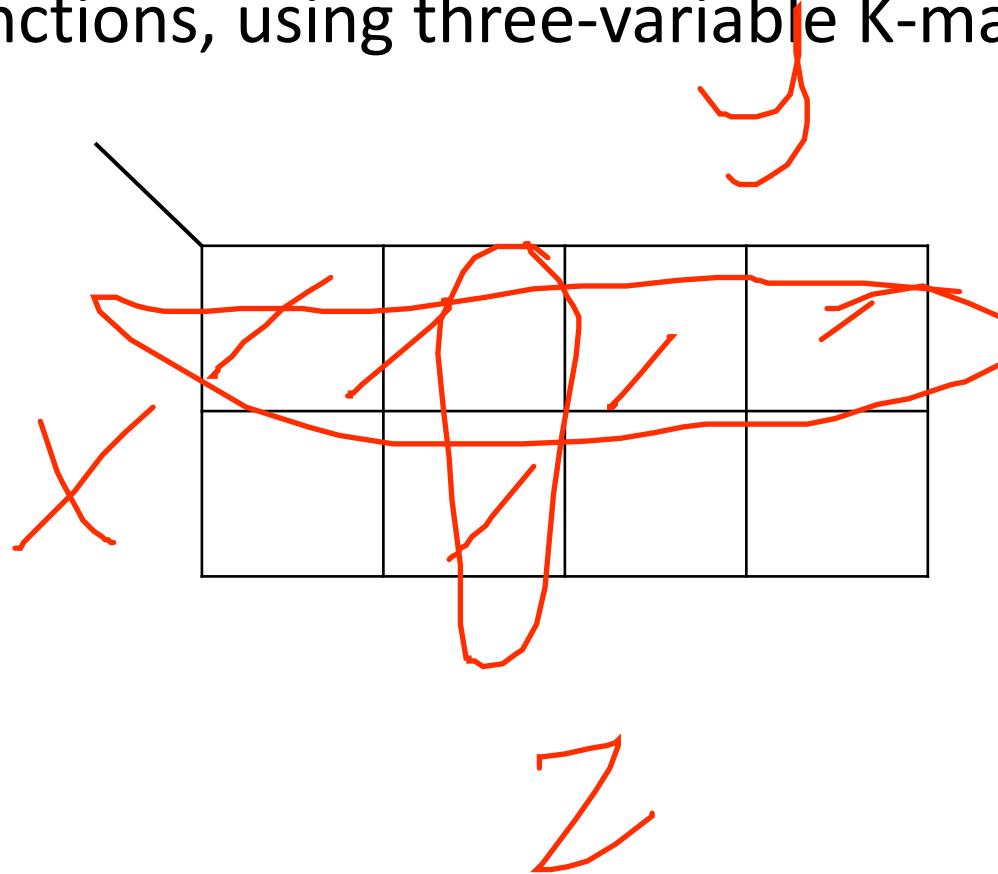
$$F = \bar{z}' + x\bar{y}'$$



3.1 Simplify the following Boolean functions, using three-variable K-maps:

(c) $F(x, y, z) = \sum (0, 1, 2, 3, 5)$

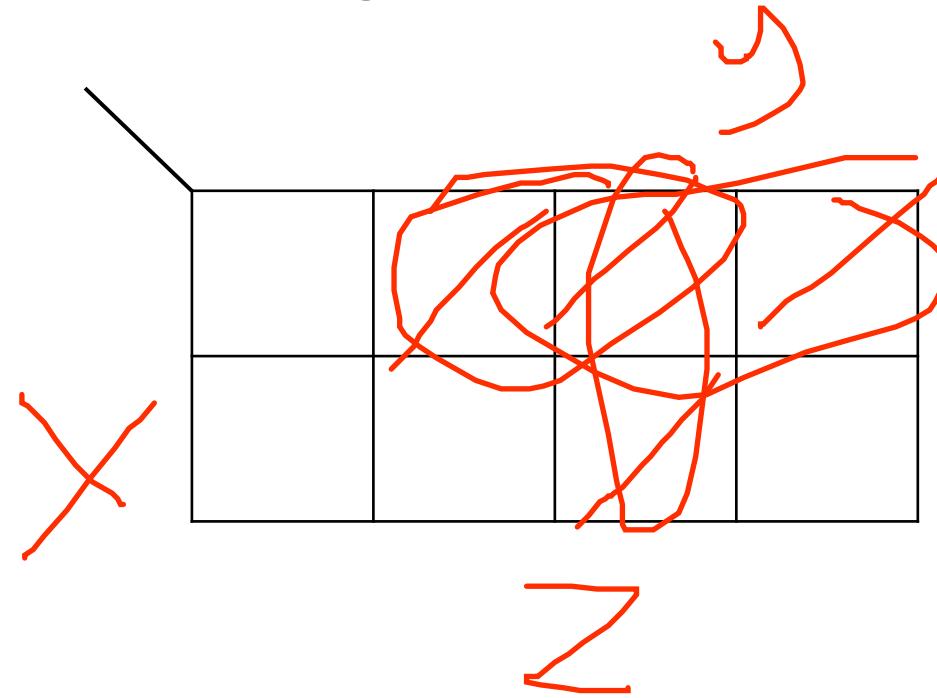
$$F = \overline{x} \cdot \overline{y} + x \cdot \overline{y} \cdot \overline{z} + x \cdot y \cdot \overline{z}$$



3.1 Simplify the following Boolean functions, using three-variable K-maps:

(d) $F(x, y, z) = \sum (1, 2, 3, 7)$

$$f = x'z + yz + x'z$$



- 1(b). Boolean expression for F.

	m_1		m_2
m_4		m_7	
m_8	m_3		m_{14}
	m_1	m_4	

* All minterms are exclusive.

$$\begin{aligned}
 F &= m_1 + m_2 + m_4 + m_7 + m_8 + m_{11} + m_{13} + m_{14} \\
 &= u'x'y'z + u'x'y'z' + u'xy'z' + u'xyz + u'x'y'z' + u'xyz + ux'y'z + ux'yz' + ux'yz' \\
 &\stackrel{\textcircled{1}}{\underset{=}{\text{L}}} \quad \stackrel{\textcircled{1}}{\underset{=}{\text{L}}} \\
 &= u'x'(y'z + yz') + u'x(y'z' + yz) + ux'(y'z' + yz) + ux(y'z + yz') \\
 &\stackrel{\textcircled{2}}{\underset{=}{\text{L}}} = (u'x' + ux)(y'z + yz') + (u'x + ux')(y'z' + yz) \\
 &= \frac{(u \oplus x)'}{A'} \left(\frac{y \oplus z}{B} \right) + \frac{(u \oplus x)}{A} \left(\frac{y \oplus z}{B'} \right)
 \end{aligned}$$

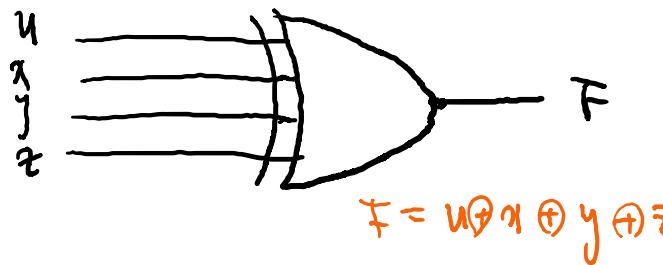
$$\begin{aligned}
 * A \oplus B &= A'B + AB' \\
 (\underline{A \oplus B})' &= A'B' + AB \\
 &= A \odot B
 \end{aligned}$$

$$\begin{aligned}
 * A'B' + AB &\neq 1 \\
 \text{if, } A=B &\rightarrow A'B' + AB = 1 \\
 &= (u \oplus x) \oplus (y \oplus z) \\
 &= \underline{u \oplus x \oplus y \oplus z}
 \end{aligned}$$

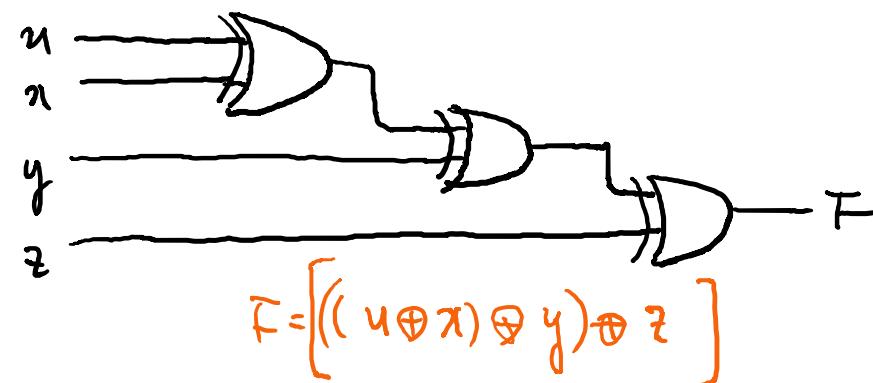
- 1(c). Draw a logic diagram for F.

$$F = u \oplus x \oplus y \oplus z$$

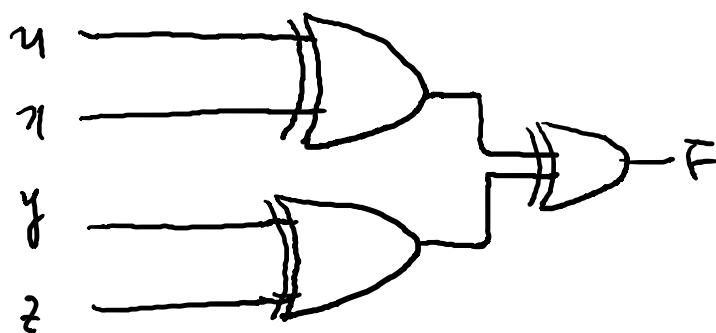
i)



ii)

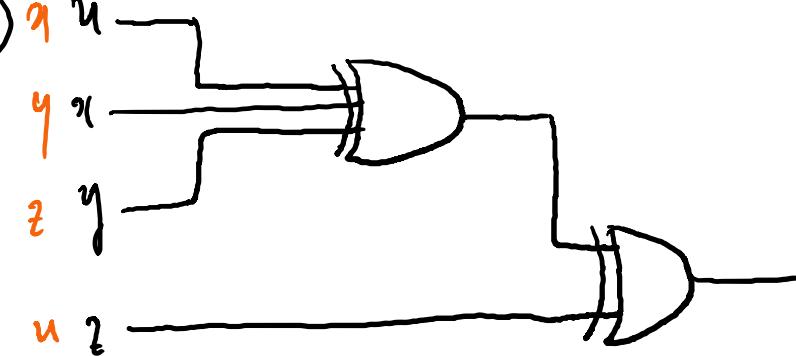


iii)



$$F = (u \oplus x) \oplus (y \oplus z)$$

iv)



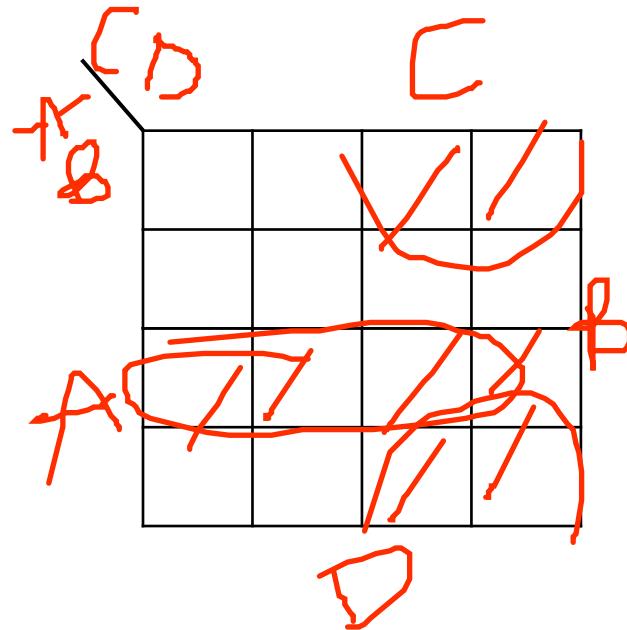
$$\begin{aligned} F &= (u \oplus x \oplus y) \oplus z \\ &= u \oplus (x \oplus y \oplus z) \end{aligned}$$

- 1) Consider the Boolean function: $F(A, B, C, D) = AB + B'C + ACD + ABD' + ACD'$
- 2) Extract a minimized Boolean expression for F using the Karnaugh map of 1).
- 3) Simplify the Boolean expression F and Compare it with the minimized Boolean expression of 2)

$$F = AB + B'C + ACD + ABD' + ACD'$$

$$AB(1+D') + AC(D+D') + B'C$$

$$AB + AC + B'C$$



$$F = AB + B'C$$

ABCD	abcd	$AB'CD$
1100	1010	0010
1101	1011	0011
1110	1110	1010
1111	1111	1011

consensus

$$AB + AC + B'C$$

$$(A+B)(A+C)(B'+C)$$

$$x'y + yz + xz$$

$$A'B + (B+B')AC + B'C$$

$$AB + ABC + AB'C + B'C$$

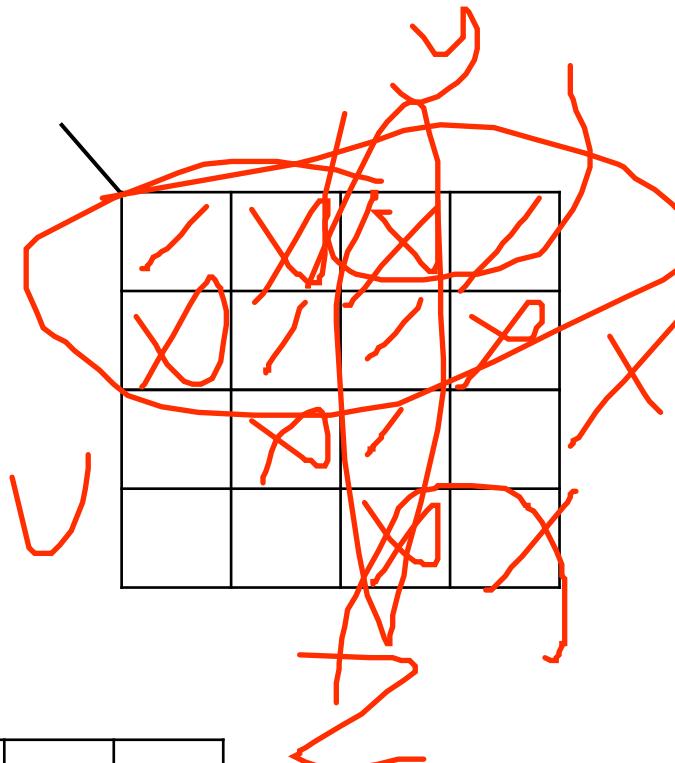
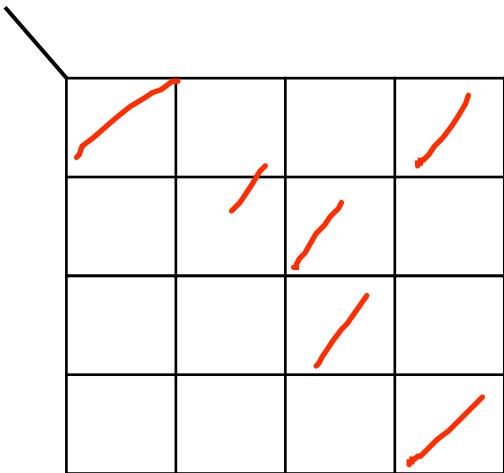
$$AB(1+C) + B'C(A+1)$$

- 2(a). Consider the Karnaugh map for F together with the don't care conditions, d. Show the Karnaugh map for G.

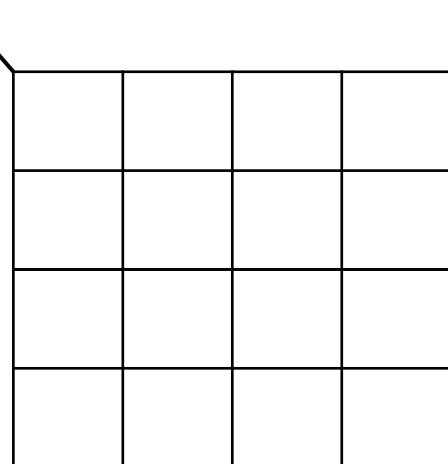
$$F(u, x, y, z) = \sum(0, 2, 5, 7, 10, 15)$$

$$d(u, x, y, z) = \sum(1, 3, 4, 6, 11, 13)$$

$$G(u, x, y, z) = F + d$$



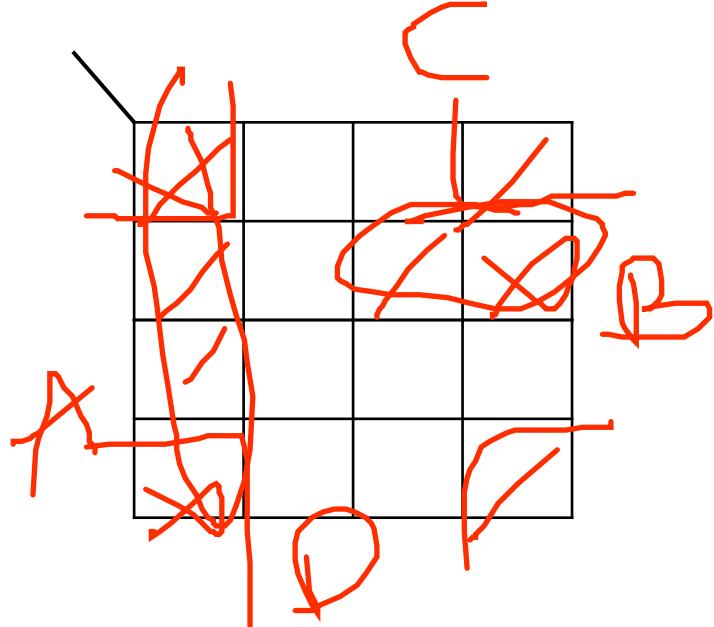
$$F = u' + yz + x'y$$



3.15 Simplify the following Boolean function F, together with the don't care conditions d, and then express the simplified function in sum-of-minterms form:

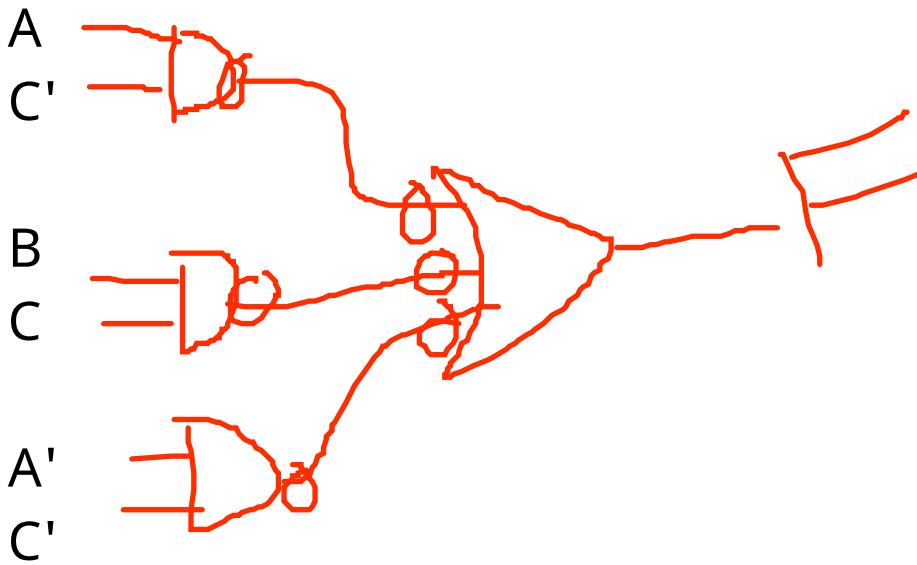
$$(d) F(A, B, C, D) = \Sigma(4, 12, 7, 2, 10)$$

$$d (A, B, C, D) = \Sigma(0, 6, 8)$$



$$F = B'D' + C'D' + A'BC$$

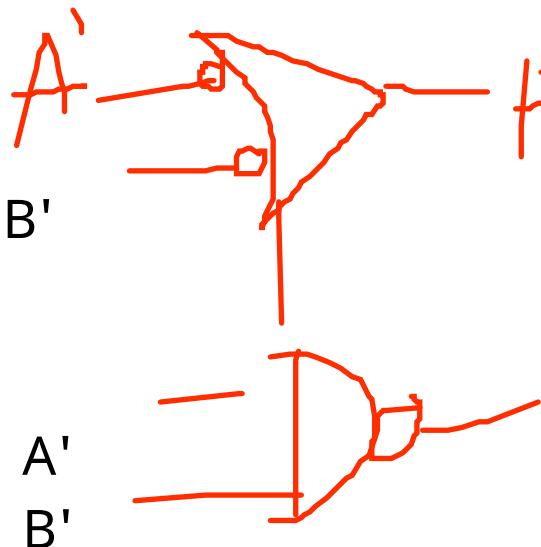
- Convert the Boolean expression $F=AC' + BC + A'C'$ to only NAND gates.



$$F = AC' + BC + A'C'$$

$$F = (F')' = \{(AC' + BC + A'C')\}' = \{(AC')'(BC)'(A'C')'\}'$$

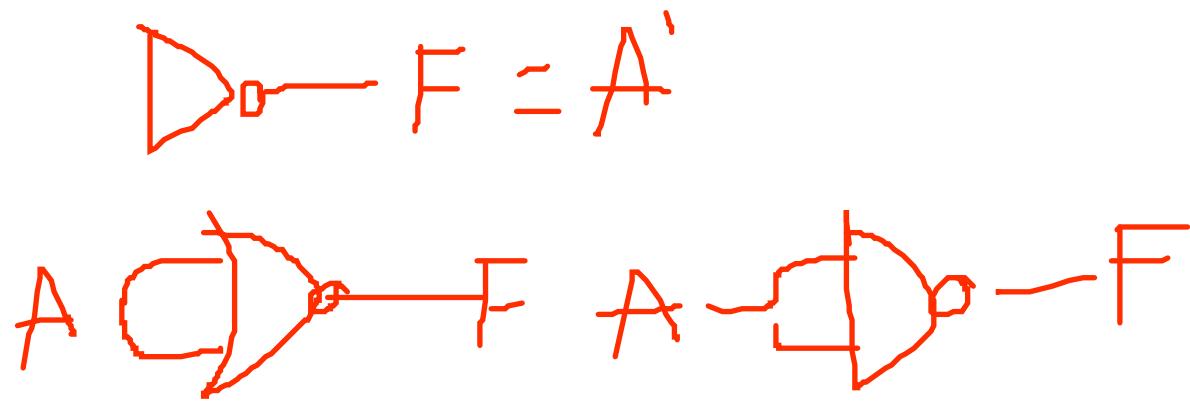
- Draw a logic diagram for $F = A + B$ using only NAND gates.



$$F' = (A + B)'$$

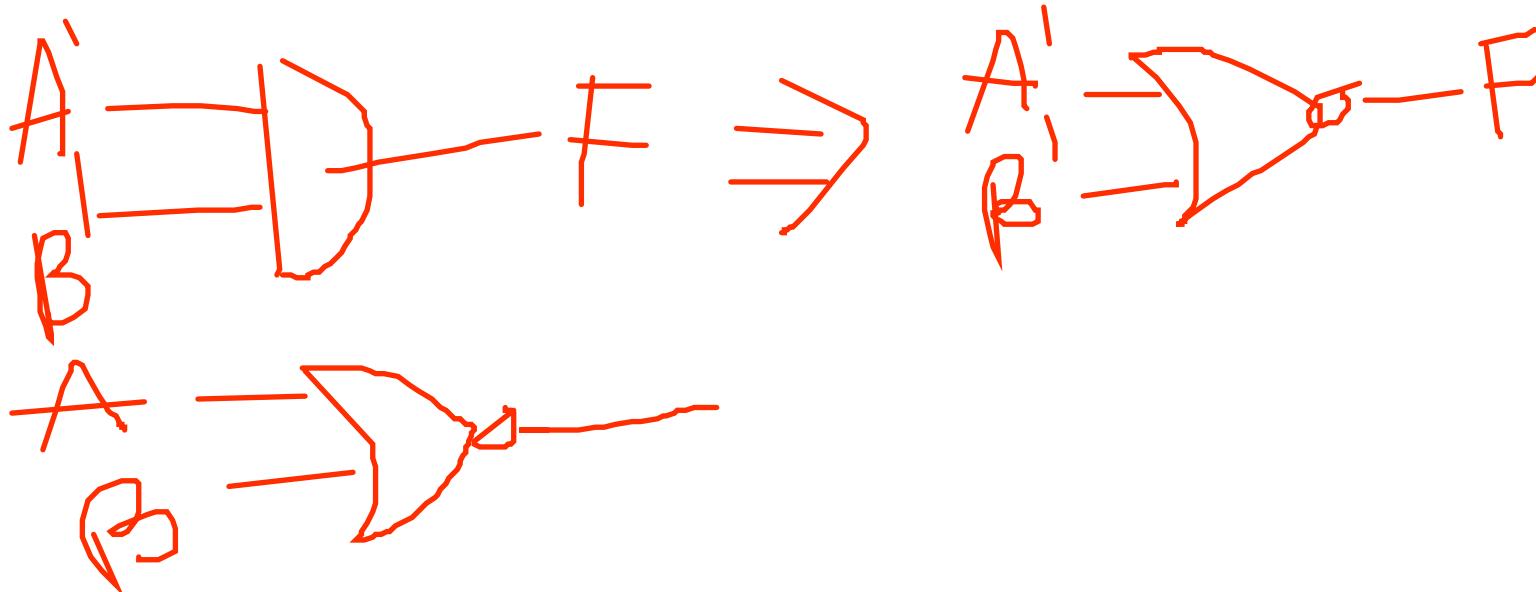
$$F = (A'B')'$$

- Draw a logic diagram for $F=A'$ using only NOR gates.



X	Y	NOR	NAND
0	0	1	1
0	1	1	0
1	0	0	1
1	1	0	0

- Draw a logic diagram for $F=AB$ using only NOR gates.



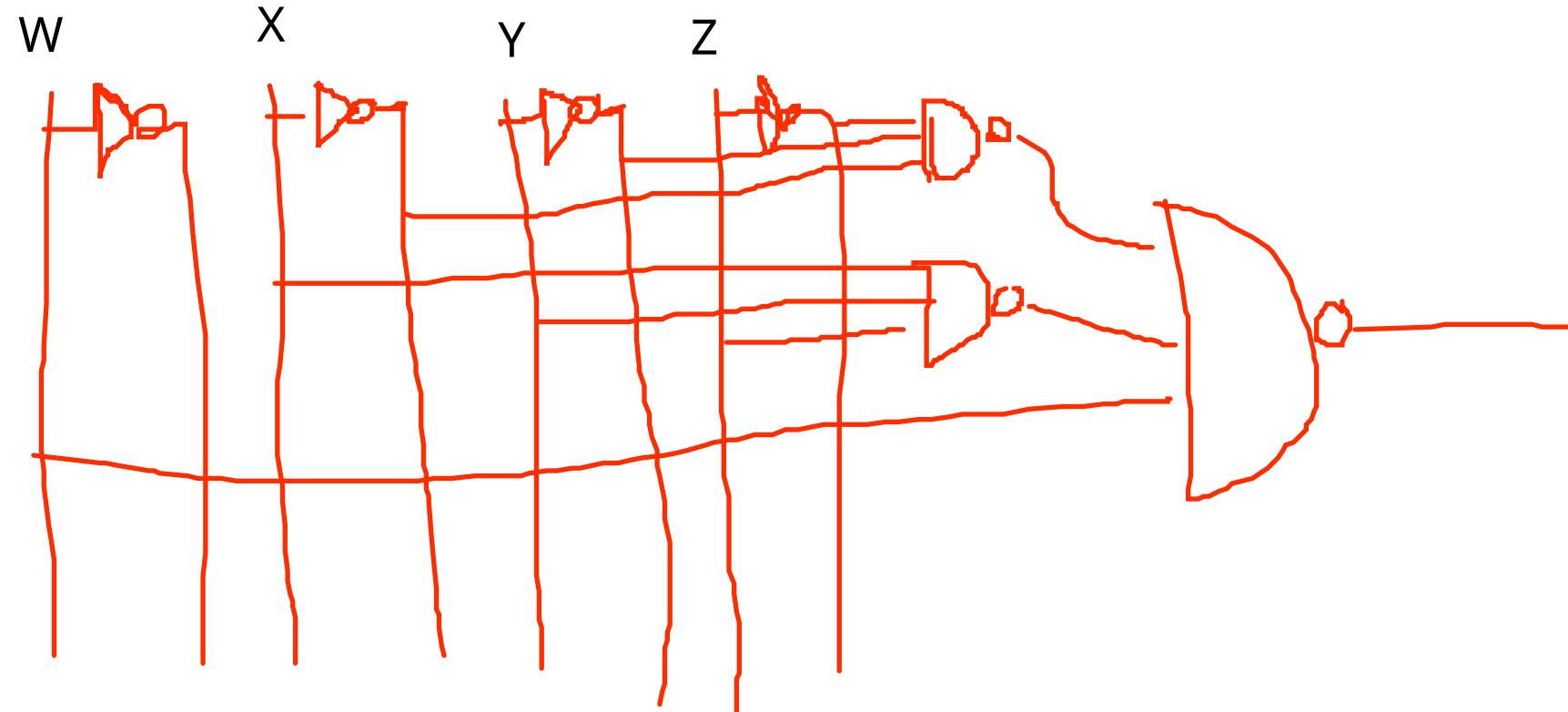
$$\begin{aligned}A'B' &= (A+B)' \\F &= (A'+B')'\end{aligned}$$

3.21 Draw (a) the multiple-level NAND circuit for the following expression and (b) repeat (a) for a NOR circuit.

$$w(x + y + z) + xyz$$

$$X+Y+Z = (X'Y'Z')'$$

$$W(X'Y'Z')' + xyz = (W(X'Y'Z')'(XYZ))'$$



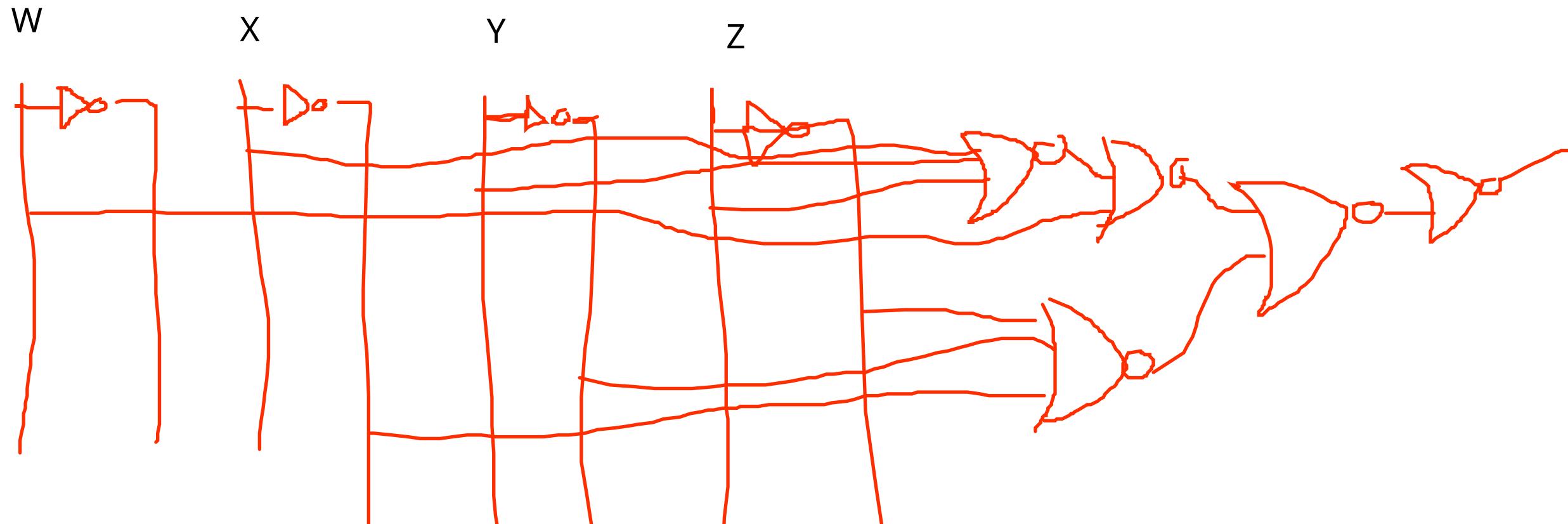
3.21 (b) repeat (a) for a NOR circuit.

$$w(x + y + z)' + xyz$$

$$W(X+Y+Z) = (W' + (X+Y+Z)')'$$

$$xyz = (X'+y'+z')'$$

$$(W' + (X+Y+Z)')' + (X'+y'+z')'$$



ECE124: Discussion

Discussion #8

Yeonsik Noh, PhD

3.21 Draw (a) the multiple-level NAND circuit for the following expression and (b) repeat (a) for a NOR circuit.

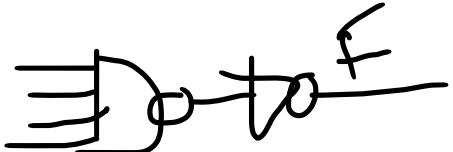
$$w(x + y + z) + xyz$$

$$F(wxyz) = w(x+y+z) + xyz$$

$$(w(x+y+z) + xyz)'$$

$$(wx+wy+wz+xyz)'$$

$$(wz)'(wy)'(wz)'(xyz)'$$



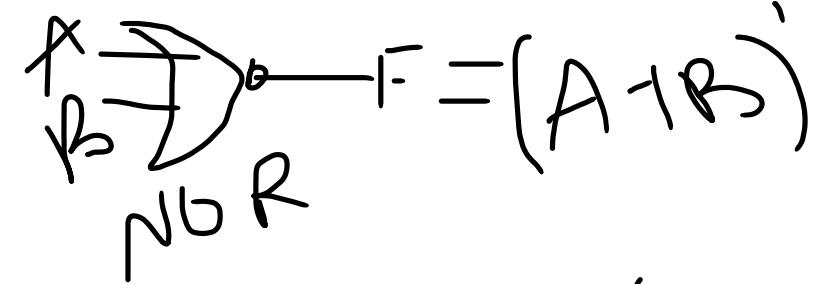
$$(F')' = ((wz)'(wy)'(wz)'(xyz)')'$$

$$\begin{aligned} F' &= (w(x'y'z')' + xyz)' \\ &= (w(x'y'z')')'(xyz)' \\ &= ((w(x'y'z')')'(xyz)')' \end{aligned}$$



3.21 (b) repeat (a) for a NOR circuit.

$$w(x + y + z) + xyz$$

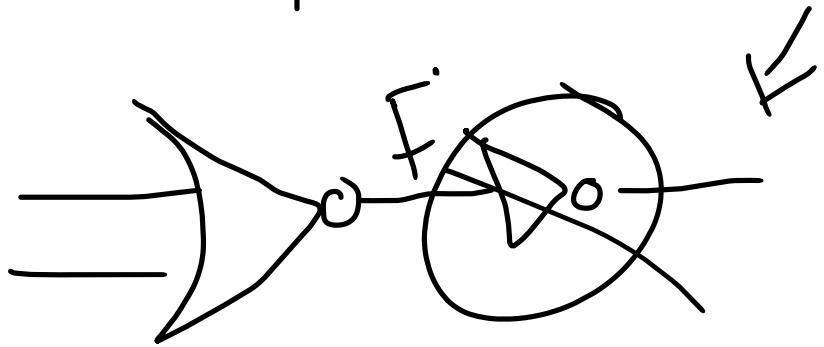


$$F' (w(x+y+z)+xyz)'$$

$$(w(x+y_z))'(xyz)'$$

$$((w'+(x+y+z)')(x'+y'+z'))'$$

$$(((w'+(x+y+z)')'+(x'+y'+z')')')'$$



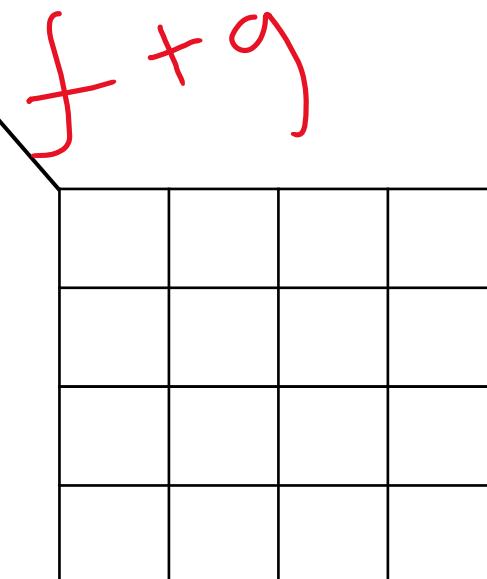
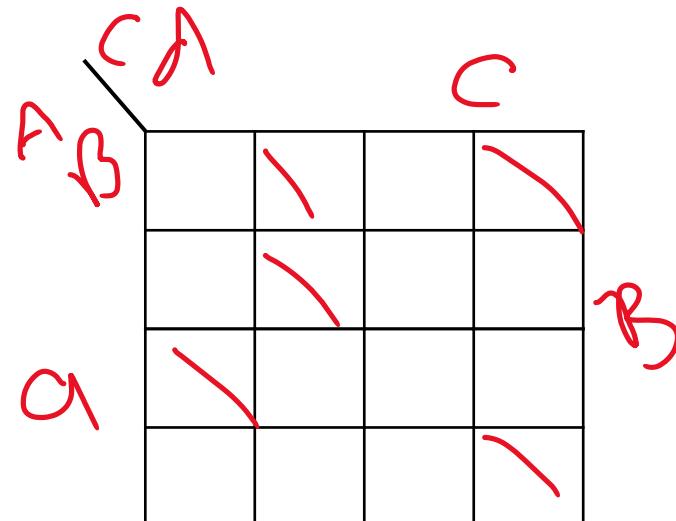
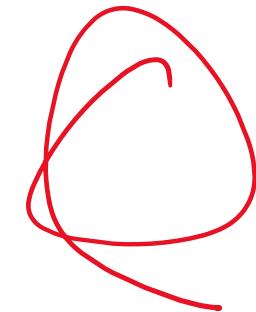
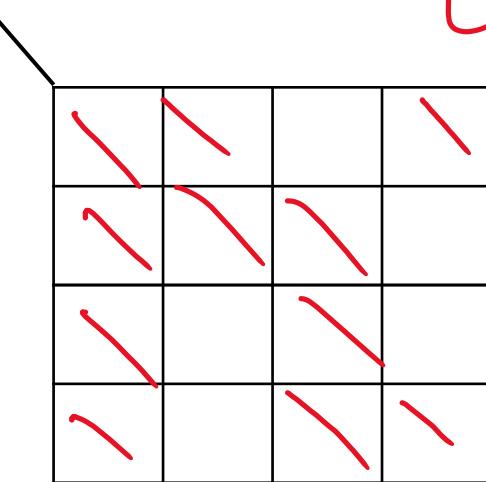
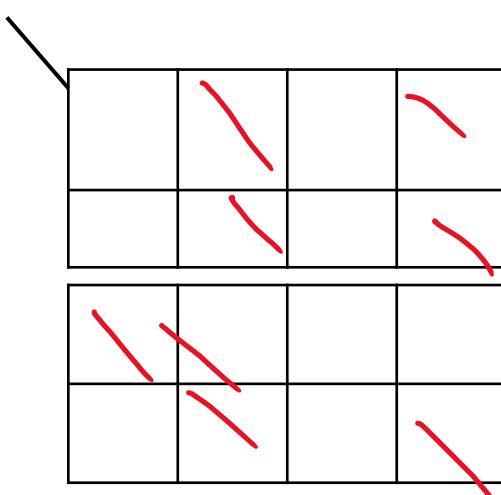
3.26 With the use of maps, find the simplest sum-of-products form of the function $F = fg$, where

$$f = abc' + c'd + a'cd' + b'cd'$$

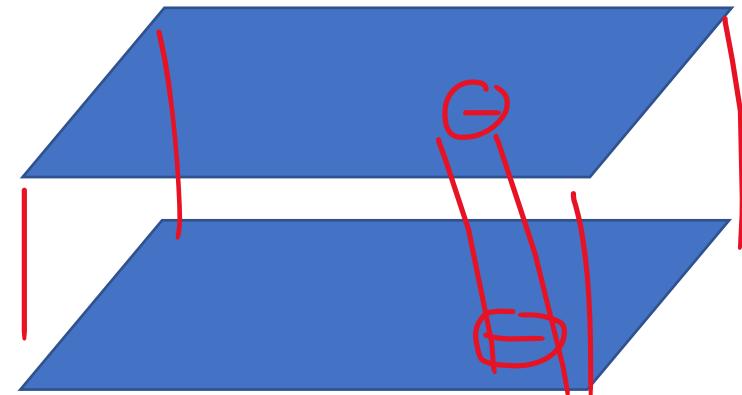
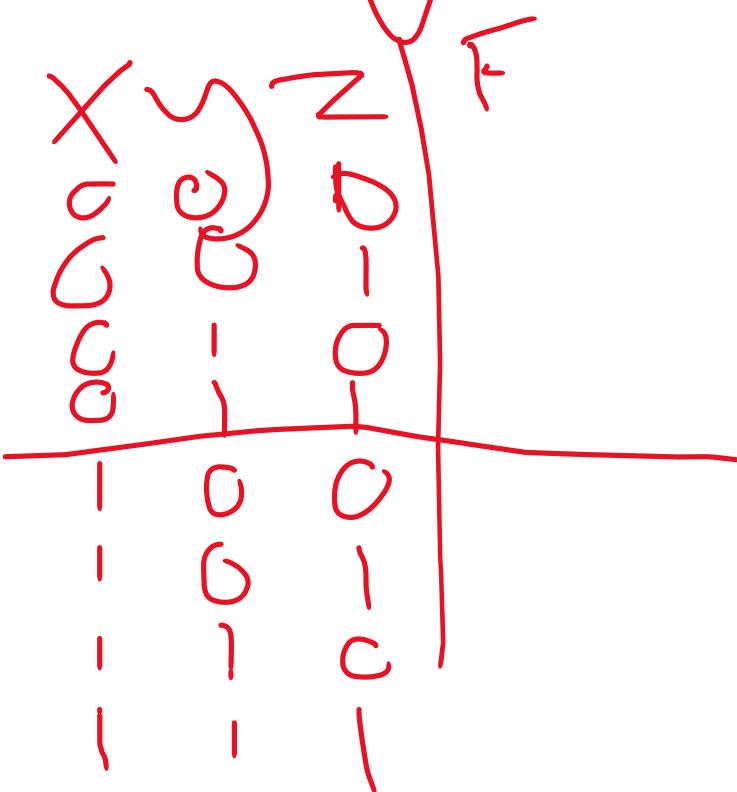
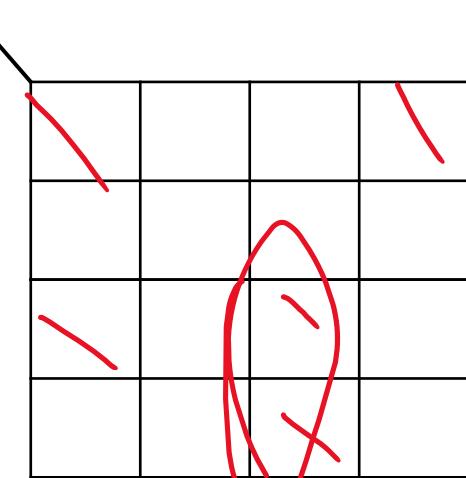
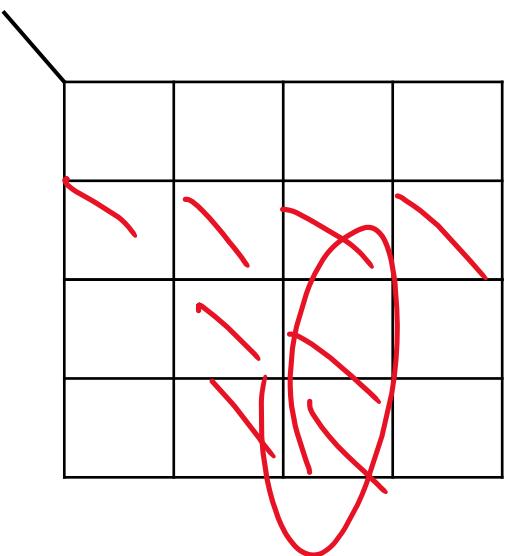
$$g = (a + b + c' + d')(b' + c' + d)(a' + c + d')$$

$$a' \perp c + d'$$

$$\sim a, \sim c, \sim d'$$

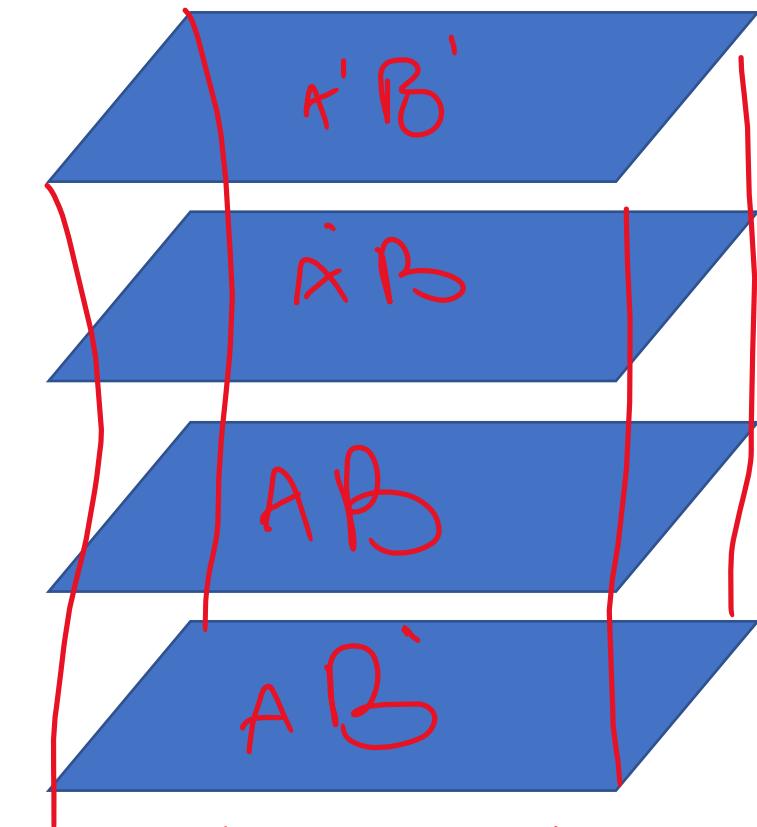
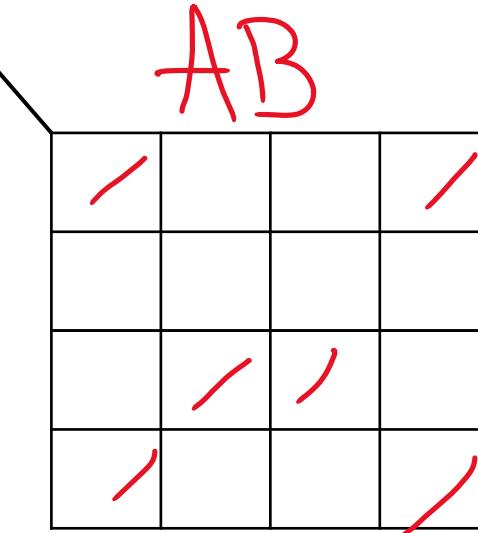
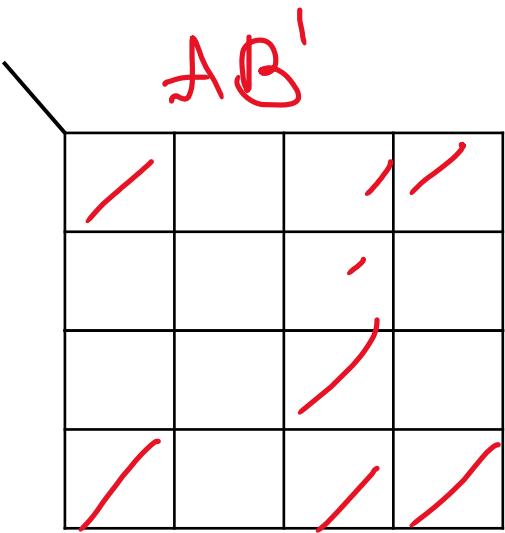
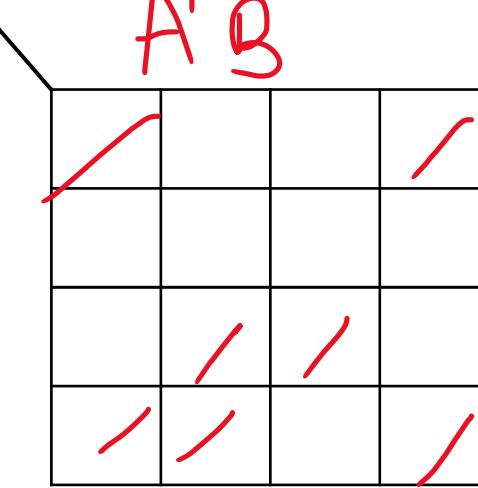
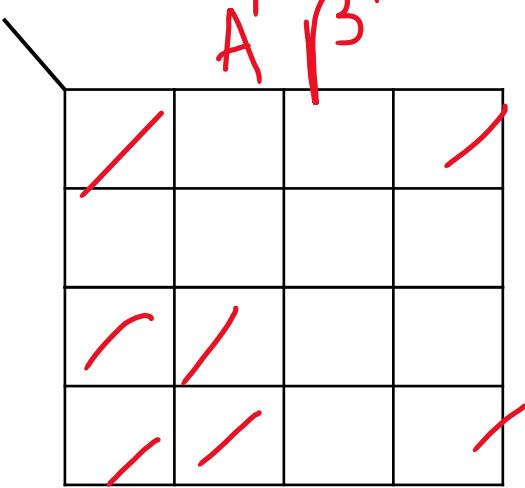


- 5 variables K-map: $F(A, B, C, D, E) = \Sigma(4, 5, 6, 7, 9, 11, 13, 15, 16, 18, 27, 28, 31)$



- 6 variables K-map:

$$F(A, B, C, D, E, F) = \Sigma(0, 2, 8, 9, 10, 12, 13, 16, 18, 24, 25, 26, 29, 31, 32, 34, 35, 39, 40, 42, 43, 47, 48, 50, 56, 58, 61, 63)$$



$$\text{O} : (\bar{A}\bar{B}' + \bar{A}'B)$$

$$\alpha (C F F')$$

4.13 The adder-subtractor circuit of Fig. 4.13 has the following values for mode input M and data inputs A and B. In each case, determine the values of the four SUM outputs, the carry C, and overflow V.

	M	A	B
(a)	0	0111	0110
(b)	1	1000	1001

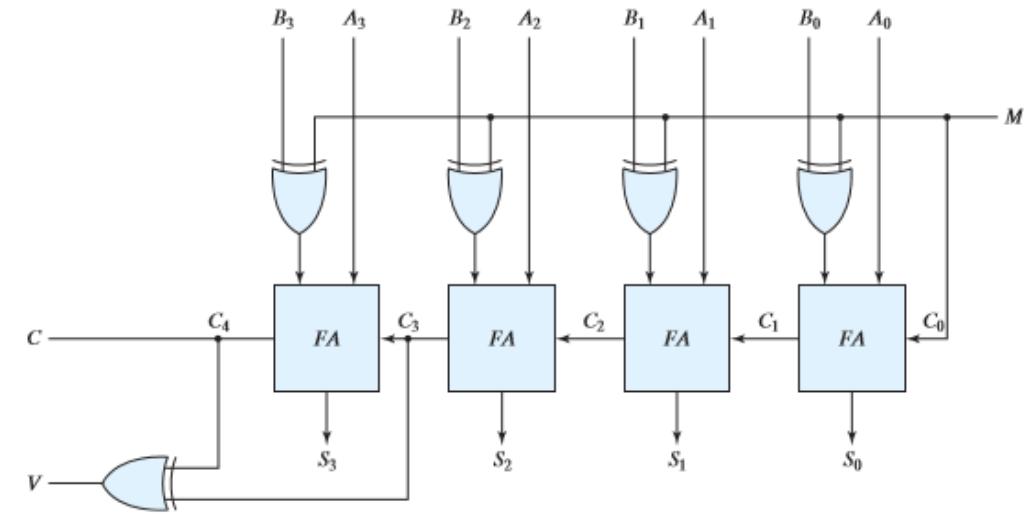


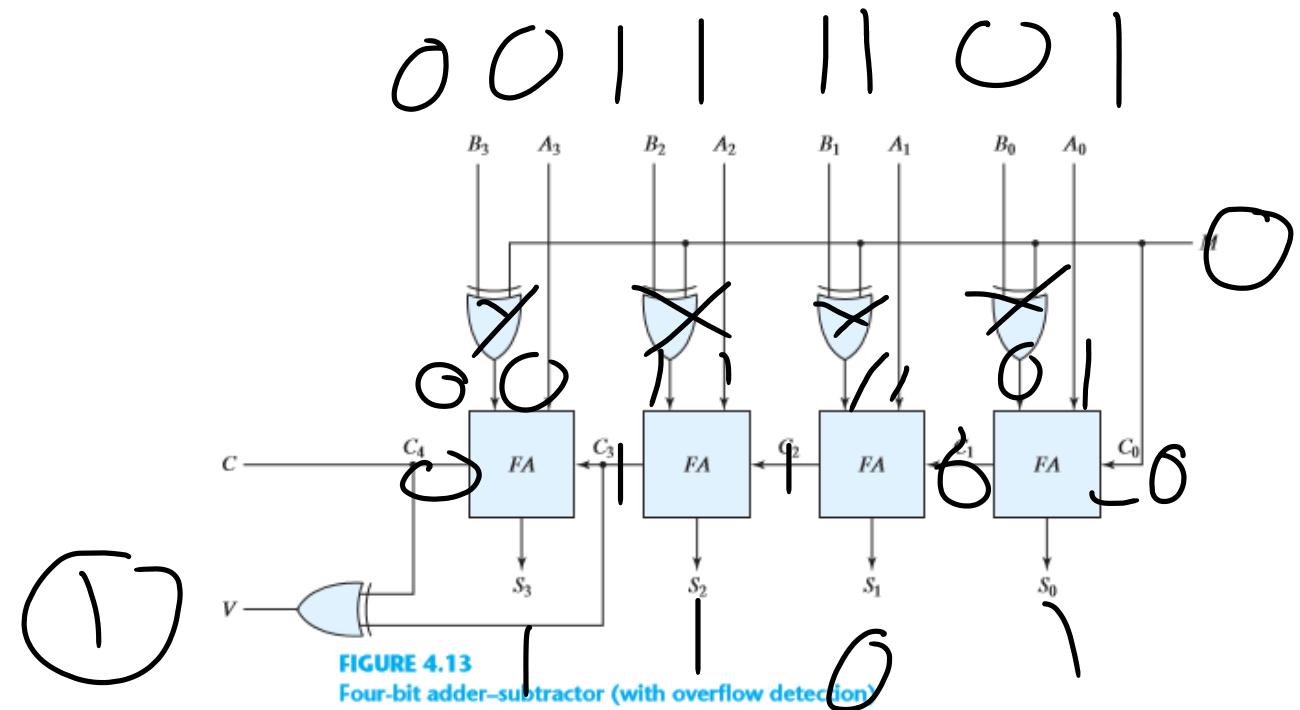
FIGURE 4.13
Four-bit adder-subtractor (with overflow detection)

4.13 The adder-subtractor circuit of Fig. 4.13 has the following values for mode input M and data inputs A and B. determine the values of the four SUM outputs, the carry C, and overflow V.

(a) M A B
 0 0111 0110
 1011(?)

-8<4 bit signed # < +7

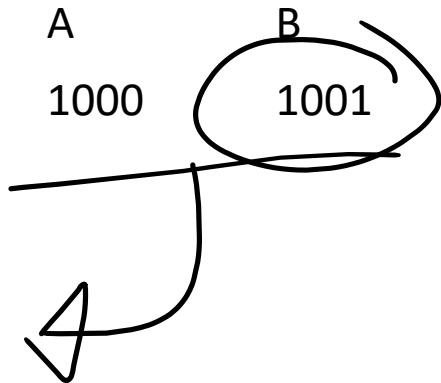
$$7+(-5) = 2$$



4.13 The adder-subtractor circuit of Fig. 4.13 has the following values for mode input M and data inputs A and B.

determine the values of the four SUM outputs, the carry C, and overflow V.

(b) M
1



(a)-(b)
(-a)-(-b)
(-a)+(-(-b))

1000
1000

1001
0111

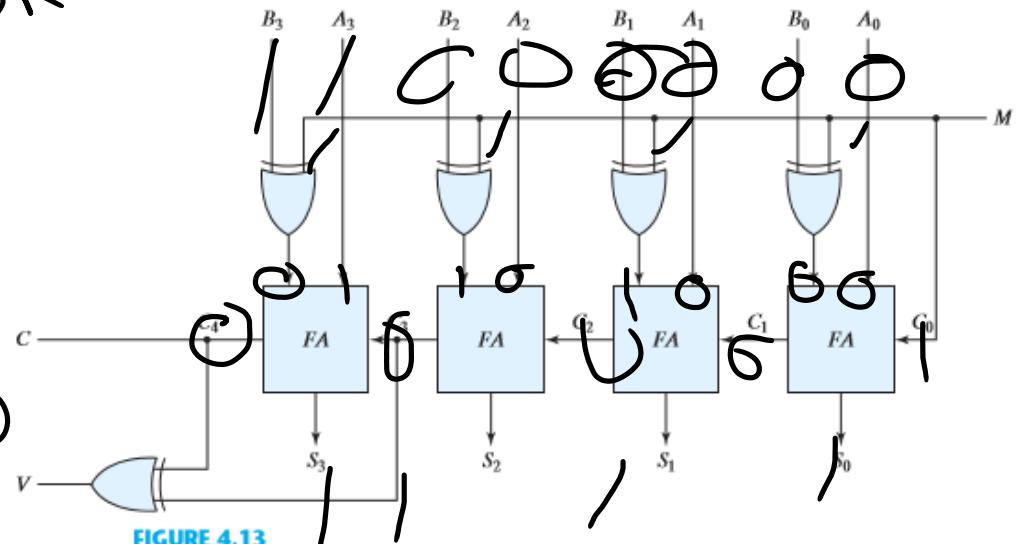
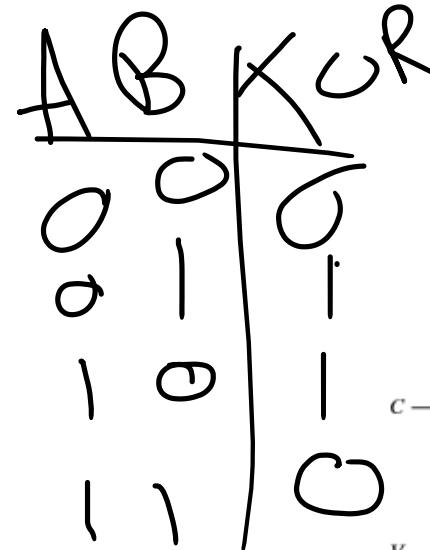


FIGURE 4.13
Four-bit adder-subtractor (with overflow detection)

- Carry Lookahead Equations

$$\begin{array}{l} \text{full add} \\ \left[\begin{array}{l} C_{i+1} = A_i B_i + (A_i \oplus B_i) C_i \\ S_i = A_i \oplus B_i \oplus C_i \end{array} \right] \end{array}$$

Recall $\begin{cases} G_i = A_i B_i \leftarrow \text{"carry generate" for stage } i \\ P_i = A_i \oplus B_i \leftarrow \text{"carry propagate" for } i \end{cases}$

$$\begin{cases} C_{i+1}' = G_i + P_i C_i \\ S_i' = P_i' \oplus C_i \end{cases}$$

C_0 = input carry

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$

$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0) = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

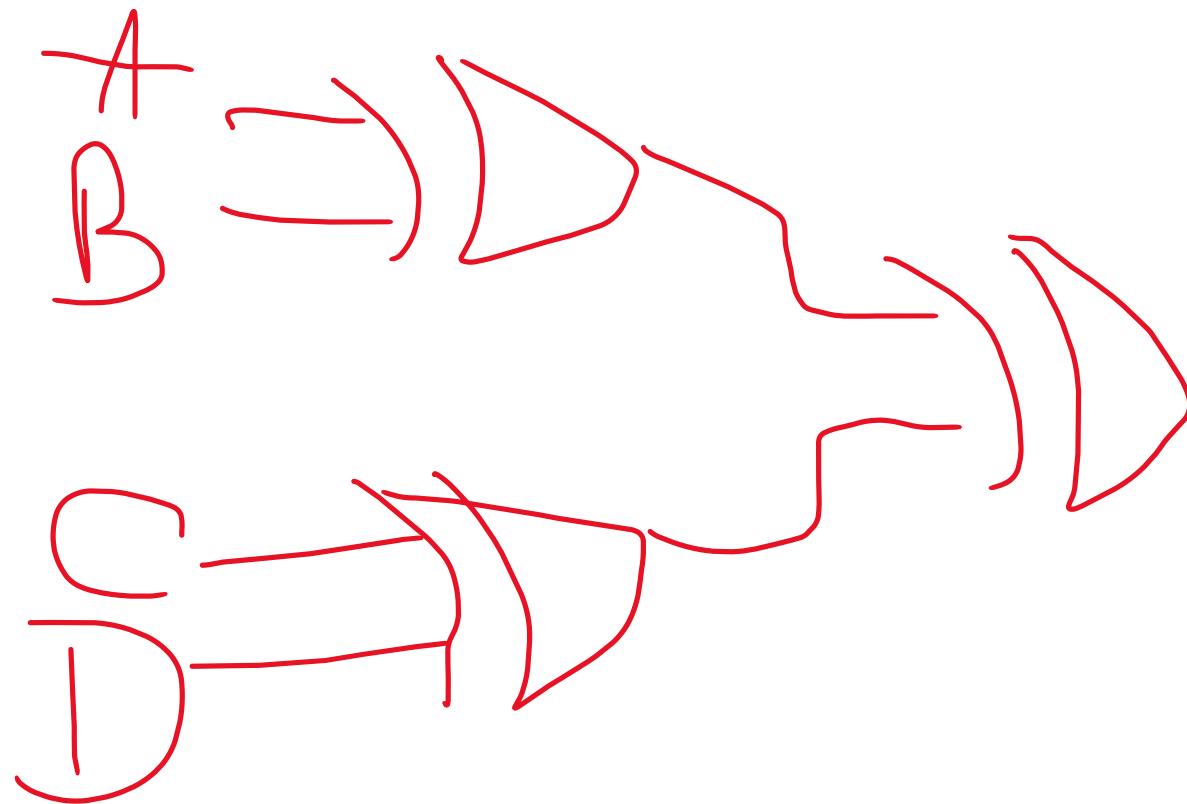
$$S_0 = P_0 \oplus C_0$$

$$S_1 = P_1 \oplus C_1$$

$$S_2 = P_2 \oplus C_2$$

$$S_3 = P_3 \oplus C_3$$

- Design a combinational circuits for parity generator/checker



ECE124: Discussion

Discussion #9

Yeonsik Noh, PhD

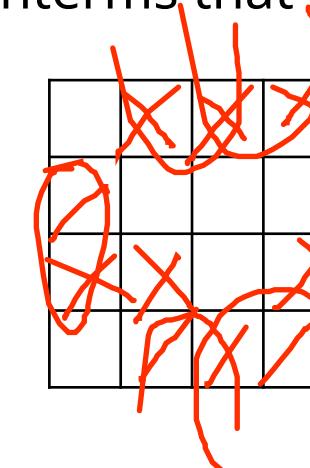
- Design a code converter that converts a decimal digit from

(a) The 8, 4, -2, -1 code to BCD

don't care conditions are the minterms that exist outside the range
bc 1 digit

W	X	Y	Z	A	B	C	D
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

$A = \overline{W}\overline{X} + \overline{W}Y' + \overline{X}Z + X\overline{Y} + XY'Z'$



$$B = X'Z + X'Y + XY'Z'$$

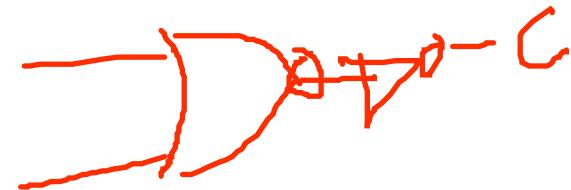
$$C = YZ' + Y'Z$$

NAND

$$C' = (YZ' + Y'Z)' = (YZ')'(Y'Z)'$$

$$C'' = (((YZ')'(Y'Z)')'$$

$$C = (Y'+Z)' + (Y+Z')'$$



- Carry Lookahead Equations

$$\begin{array}{l} \text{full add} \\ \left[\begin{array}{l} C_{i+1} = A_i B_i + (A_i \oplus B_i) C_i \\ S_i = A_i \oplus B_i \oplus C_i \end{array} \right] \end{array}$$

Recall $\begin{cases} G_i = A_i B_i & \leftarrow \text{"carry generate" for stage } i \\ P_i = A_i \oplus B_i & \leftarrow \text{"carry propagate" for } i \end{cases}$

$$\begin{cases} C_{i+1}^1 = G_i + P_i C_i \\ S_i^1 = P_i^1 \oplus C_i \end{cases}$$

C_0 = input carry

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$

$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0) = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

$$S_0 = P_0 \oplus C_0$$

$$S_1 = P_1 \oplus C_1$$

$$S_2 = P_2 \oplus C_2$$

$$S_3 = P_3 \oplus C_3$$

- Add two number $A = 1101$ and $B = 1001$ with input carry $C_0 = 0$ using Boolean expression of IC type 74283, Carry Lookahead Adder.

$$P_n = A_n \oplus B_n, G_n = A_n B_n, S_n = \begin{cases} S_n = P_n \oplus C_n \\ C_{n+1} = G_n + P_n C_n \end{cases}$$

$$C_0 = 0$$

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0)$$

$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0)$$

$$S_0 = P_0 \oplus C_0$$

$$S_1 = P_1 \oplus C_1$$

$$S_2 = P_2 \oplus C_2$$

$$S_3 = P_3 \oplus C_3$$

$$\begin{array}{r} A = \begin{array}{r} 1 & 0 & 0 & 1 \end{array} \\ + B = \begin{array}{r} 1 & 1 & 0 & 1 \end{array} \\ \hline \end{array} \Rightarrow \begin{array}{l} C_4 = 1 \quad \therefore \text{overflow} \\ C_3 = 0 \\ C_2 = 0 \\ C_1 = 1 \end{array}$$

$v = C_4 \oplus C_3 = 1 \oplus 0 = 1$

$$P_0 = A_0 \oplus B_0 = 1 \oplus 1 = 0, \quad G_0 = A_0 B_0 = 1$$

$$P_1 = A_1 \oplus B_1 = 0 \oplus 0 = 0, \quad G_1 = A_1 B_1 = 0$$

$$P_2 = A_2 \oplus B_2 = 1 \oplus 0 = 1, \quad G_2 = A_2 B_2 = 0$$

$$P_3 = A_3 \oplus B_3 = 1 \oplus 1 = 0, \quad G_3 = A_3 B_3 = 1$$

$$C_0 = 0$$

$$C_1 = G_0 + P_0 C_0 = 1 + (0 \cdot 0) = 1$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_0 = 0 + (0 \cdot 1) + 0 \cdot 1 \cdot 0 = 0$$

$$C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0 = 0 + 0 + 0 + 0 = 0$$

$$\begin{aligned} C_4 &= G_3 + \overline{P_3 G_2} + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0 \\ &= 1 + 0 + 0 + 0 + 0 = 1 \end{aligned}$$

$$S_0 = P_0 \oplus C_0 = 0 \oplus 0 = 0$$

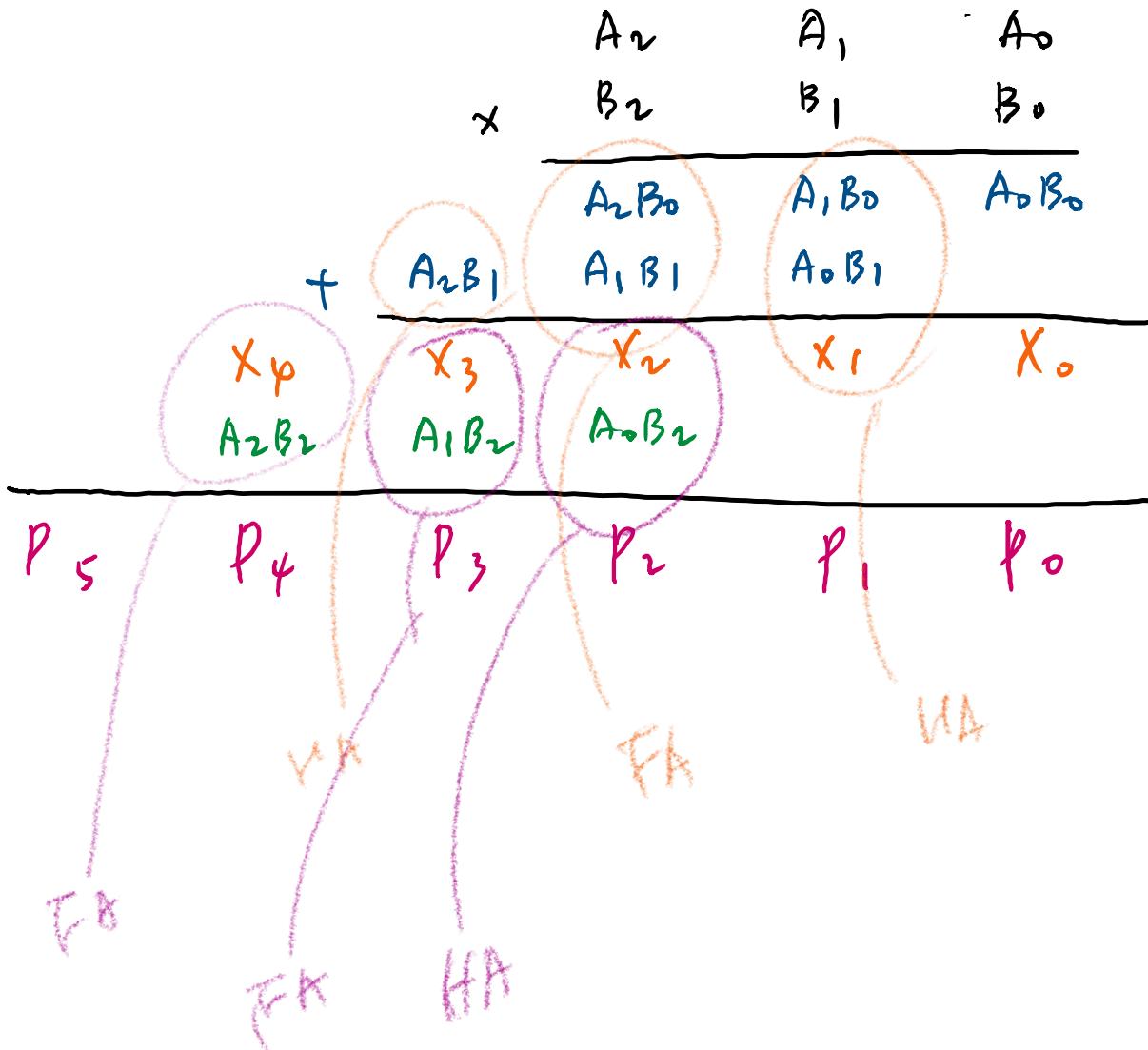
$$S_1 = P_1 \oplus C_1 = 0 \oplus 1 = 1$$

$$S_2 = P_2 \oplus C_2 = 1 \oplus 0 = 1$$

$$S_3 = P_3 \oplus C_3 = 0 \oplus 0 = 0$$

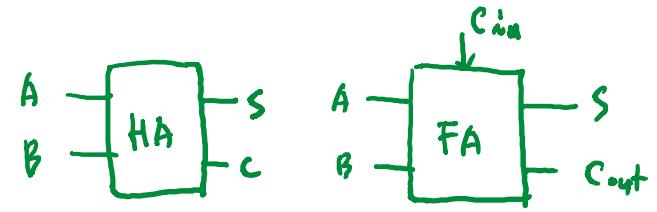
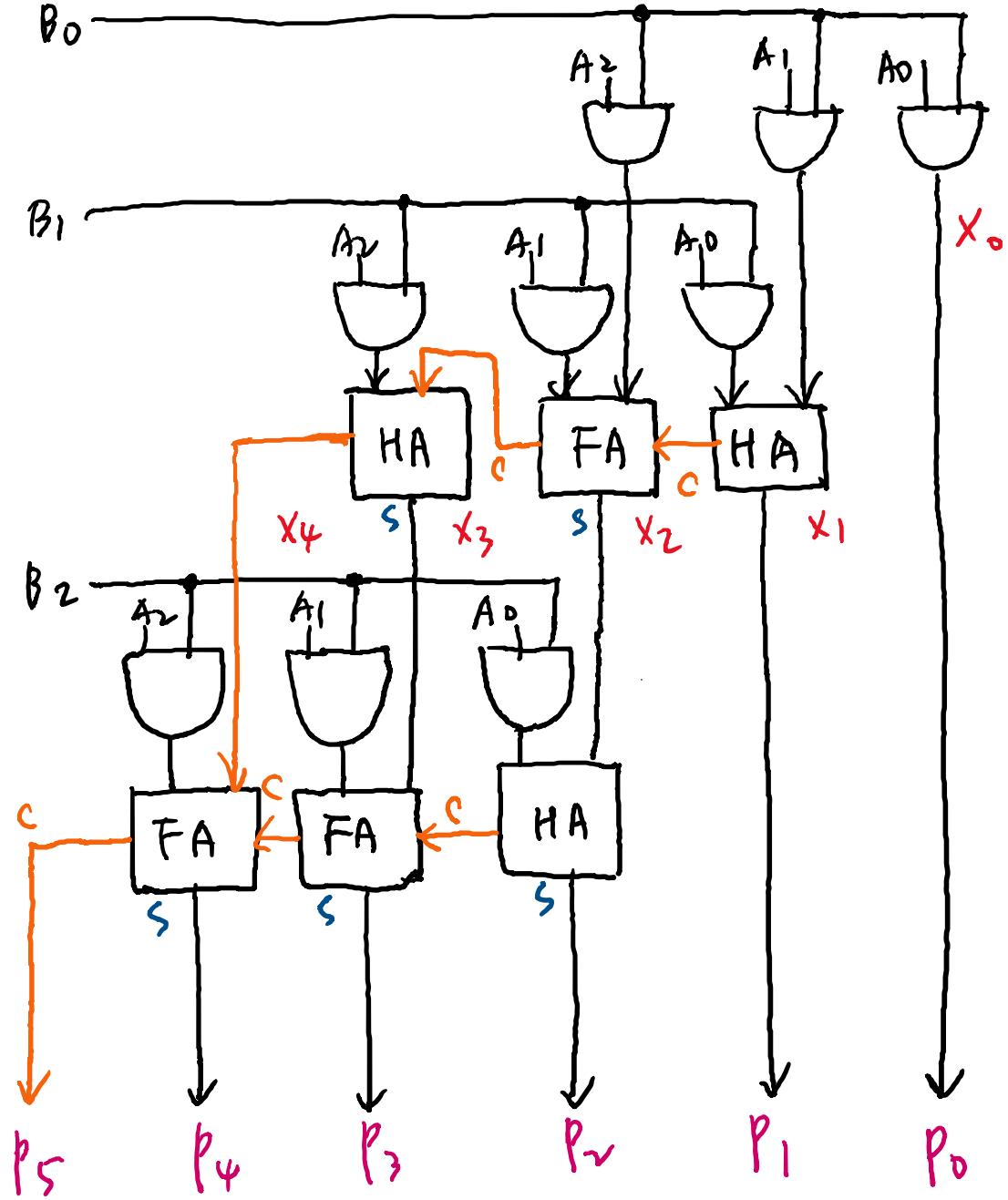
- Design a combinational circuits for 3×3 multiplier using half-adder and full-adder.

$A_2 A_1 A_0$, $B_2 B_1 B_0$



$$\left\{
 \begin{aligned}
 X_0 &= A_0 B_0 \\
 X_1 &= A_1 B_0 + A_0 B_1 \\
 X_2 &= A_2 B_0 + A_1 B_1 + \text{carry from } X_1 \\
 X_3 &= A_2 B_1 + \text{carry from } X_2 \\
 X_4 &= \text{carry from } X_3
 \end{aligned}
 \right.$$

$$\left\{
 \begin{aligned}
 P_0 &= X_0 \\
 P_1 &= X_1 \\
 P_2 &= X_2 + A_0 B_2 \\
 P_3 &= X_3 + A_1 B_2 + \text{carry from } P_2 \\
 P_4 &= X_4 + A_2 B_2 + \text{carry from } P_3 \\
 P_5 &= \text{carry from } P_4
 \end{aligned}
 \right.$$



$$\left\{ \begin{array}{l} X_0 = A_0 \oplus B_0 \\ X_1 = A_1 B_0 + A_0 B_1 \\ X_2 = A_2 B_0 + A_1 B_1 + \text{carry from } X_1 \\ X_3 = A_2 B_1 + \text{carry from } X_2 \\ X_4 = \text{carry from } X_3 \\ \\ P_0 = X_0 \\ P_1 = X_1 \\ P_2 = X_2 + A_0 B_2 \\ P_3 = X_3 + (A_1 B_2 + \text{carry from } P_2) \\ P_4 = X_4 + A_2 B_2 + \text{carry from } P_3 \\ P_5 = \text{carry from } P_4 \end{array} \right.$$

$$x_0 = A_0 B_0$$

$$x_1 = A_1 B_0 + A_0 B_1$$

$$x_2 = A_2 B_0 + A_1 B_1 + \text{carry from } x_1$$

$$x_3 = A_2 B_1 + \text{carry from } x_2$$

$$x_4 = \text{carry from } x_3$$

$$p_0 = x_0$$

$$p_1 = x_1$$

$$p_2 = x_2 + A_0 B_2$$

$$p_3 = x_3 + (A_1 B_2 + \text{carry from } p_2)$$

$$p_4 = x_4 + (A_2 B_2 + \text{carry from } p_3)$$

$$p_5 = \text{carry from } p_4$$

Ex)

	1	0	1		
X	1	1	1		
				1	0
					1

$$\begin{array}{r}
 + \quad \quad \quad 1 \quad 0 \quad 1 \\
 \hline
 p_5 \quad p_4 \quad p_3 \quad p_2 \quad p_1 \quad p_0 \\
 \hline
 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1
 \end{array}$$

$$\left\{
 \begin{array}{l}
 x_0 = 1 \\
 x_1 = 0 + 1 = 1, c=0 \\
 x_2 = 1 + 0 + 0 = 1, c=0 \\
 x_3 = 1 + 0 = 1, c=0 \\
 x_4 = 0
 \end{array}
 \right.$$

$$p_0 = x_0 = 1$$

$$p_1 = x_1 = 1$$

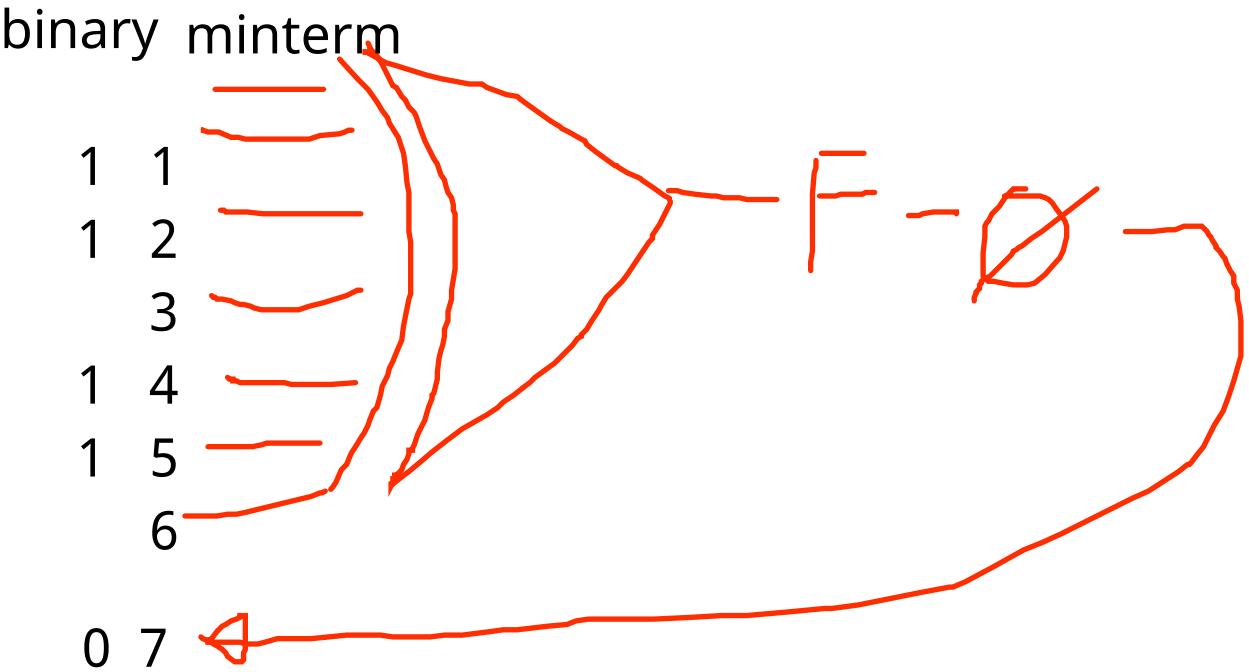
$$p_2 = 1 + 1 = 0, c=1$$

$$p_3 = 1 + 0 + 1 = 0, c=1$$

$$p_4 = 0 + 1 + 1 = 0, c=1$$

$$p_5 = 1$$

- Design a combinational circuits for parity generator/checker



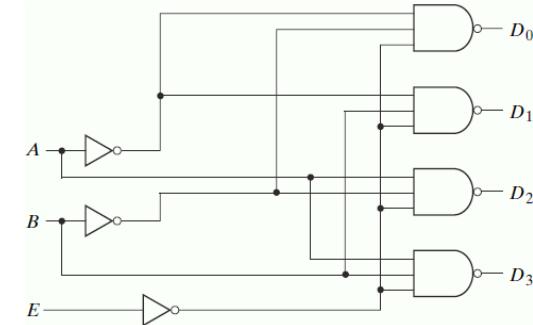
xor truth table

odd number of 1s
= true output

4.27 A combinational circuit is specified by the following three Boolean functions:

$$F_1(A, B, C) = \Sigma(1, 4, 6), F_2(A, B, C) = \Sigma(3, 5), F_3(A, B, C) = \Sigma(2, 4, 6, 7)$$

Implement the circuit with a decoder constructed with NAND gates and NAND or AND gates connected to the decoder outputs. Use a block diagram for the decode. Minimize the number of inputs in the external gates.



E	A	B	D ₀	D ₁	D ₂	D ₃
1	X	X	X	X	X	X
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

ECE124: Discussion

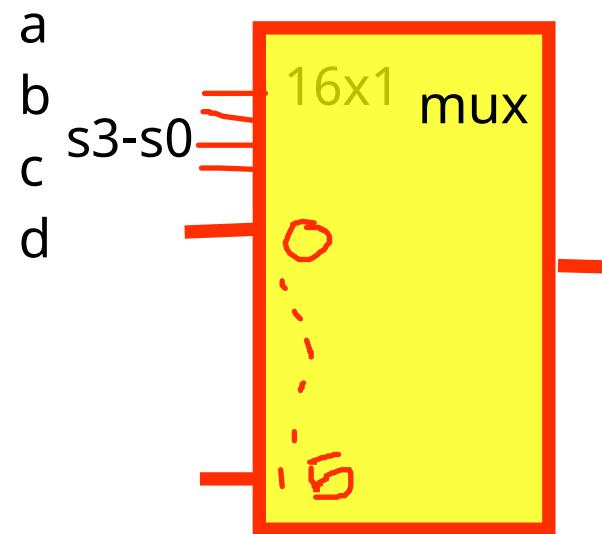
Discussion #11

Yeonsik Noh, PhD

* Implement the following Boolean function with a multiplexer.

$$F(A, B, C, D) = \pi(2, 3, 6, 7, 10, 12, 13, 14)$$

Input	F
A B C D	
0 0 0 0	
0 0 0 1	1
0 0 1 0	0
0 0 1 1	0
0 1 0 0	
0 1 0 1	1
0 1 1 0	0
0 1 1 1	0
1 0 0 0	
1 0 0 1	
1 0 1 0	1
1 0 1 1	
1 1 0 0	0
1 1 0 1	1
1 1 1 0	0
1 1 1 1	1



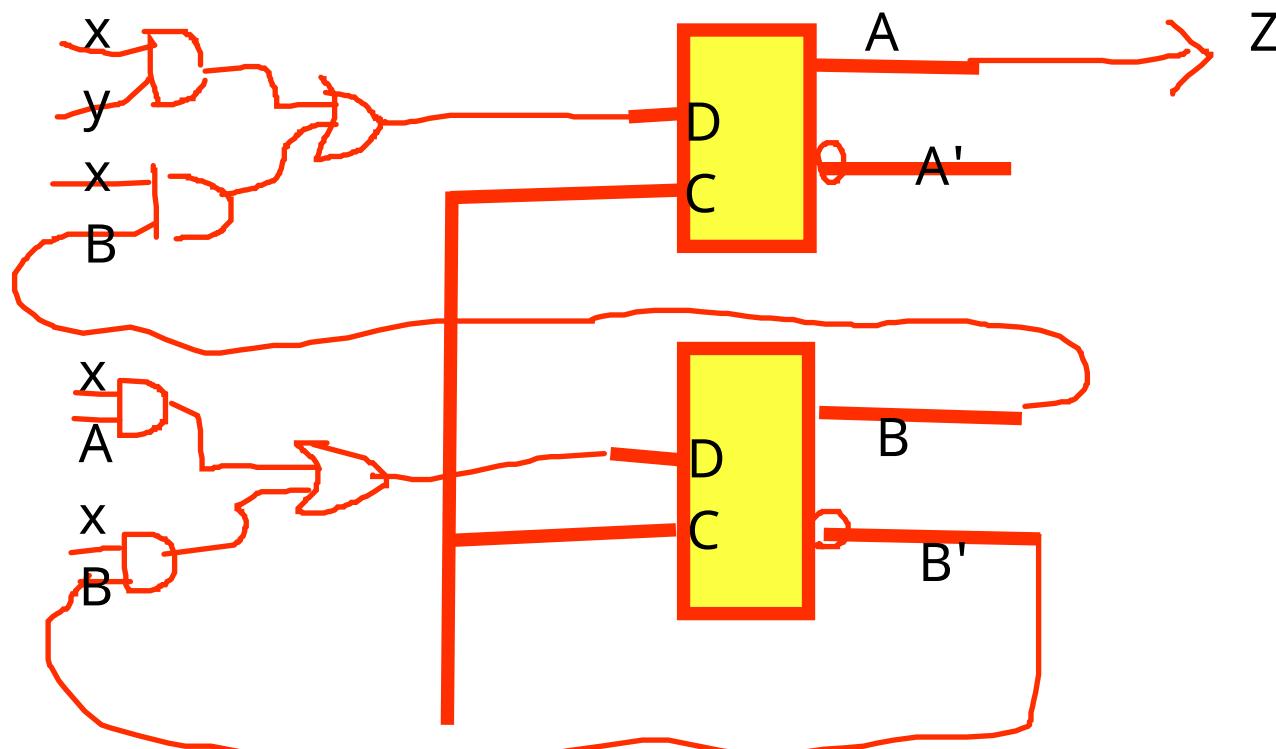
5.6 A sequential circuit with two D flip-flops A and B, two inputs, x and y; and one output z is specified by the following next-state and output equations.

$$A(t+1) = xy' + xB;$$

$$B(t+1) = xA + xB';$$

$$z = A$$

(a) Draw the logic diagram of the circuit



5.6 A sequential circuit with two D flip-flops A and B, two inputs, x and y; and one output z is specified by the following next-state and output equations.

$$A(t+1) = xy' + xB; \quad B(t+1) = xA + xB'; \quad z = A$$

(b) List the state table for the sequential circuit

Present state		Inputs		A(t+1)		B(t+1)		Next state		output
A	B	x	y	xy'	xB	xA	xB'	A	B	z
0	0	0	0	0	0			0		
0	0	0	1							
0	0	1	0					1		
0	0	1	1							
0	1	0	0							
0	1	0	1							
0	1	1	0					1		
0	1	1	1							
1	0	0	0							
1	0	0	1							
1	0	1	0					1		
1	0	1	1							
1	1	0	0							
1	1	0	1							
1	1	1	0					1		
1	1	1	1							

$A(t+1) = xy' + xB$
 $ABxy' + ABxy$
 $0010m^2_0 0110 6$
 $0110m^6_0 0111 7$
 $1010m^{10}_0 1110 14$
 $1110m^{14}_4 1111 15$

5.6 A sequential circuit with two D flip-flops A and B, two inputs, x and y; and one output z is specified by the following next-state and output equations.

$$A(t+1) = xy' + xB;$$

$$B(t+1) = xA + xB';$$

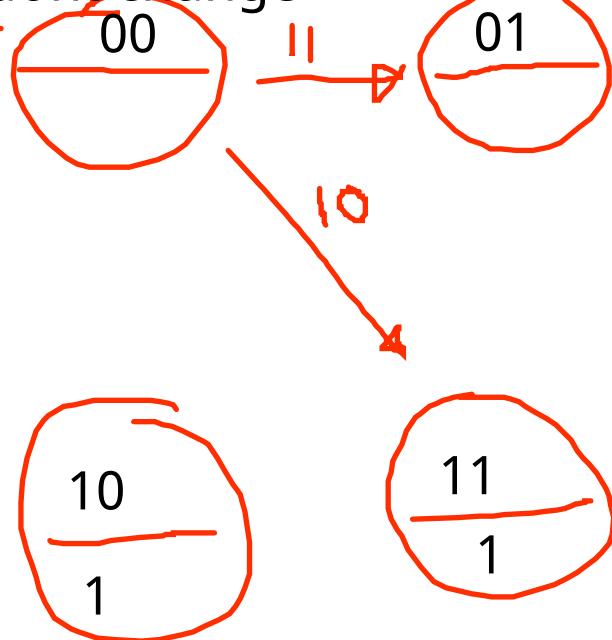
$$z = A$$

mealy
moore

(c) Draw the corresponding state diagram

Present state		Inputs		A(t+1)		B(t+1)		Next state		output
A	B	x	y	xy'	xB	xA	xB'	A	B	z
0	0	0	0							
0	0	0	1							
0	0	1	0							
0	0	1	1							
0	1	0	0							
0	1	0	1							
0	1	1	0							
0	1	1	1							
1	0	0	0							
1	0	0	1							
1	0	1	0							
1	0	1	1							
1	1	0	0							
1	1	0	1							
1	1	1	0							
1	1	1	1							

when xy 00, 01
dont change

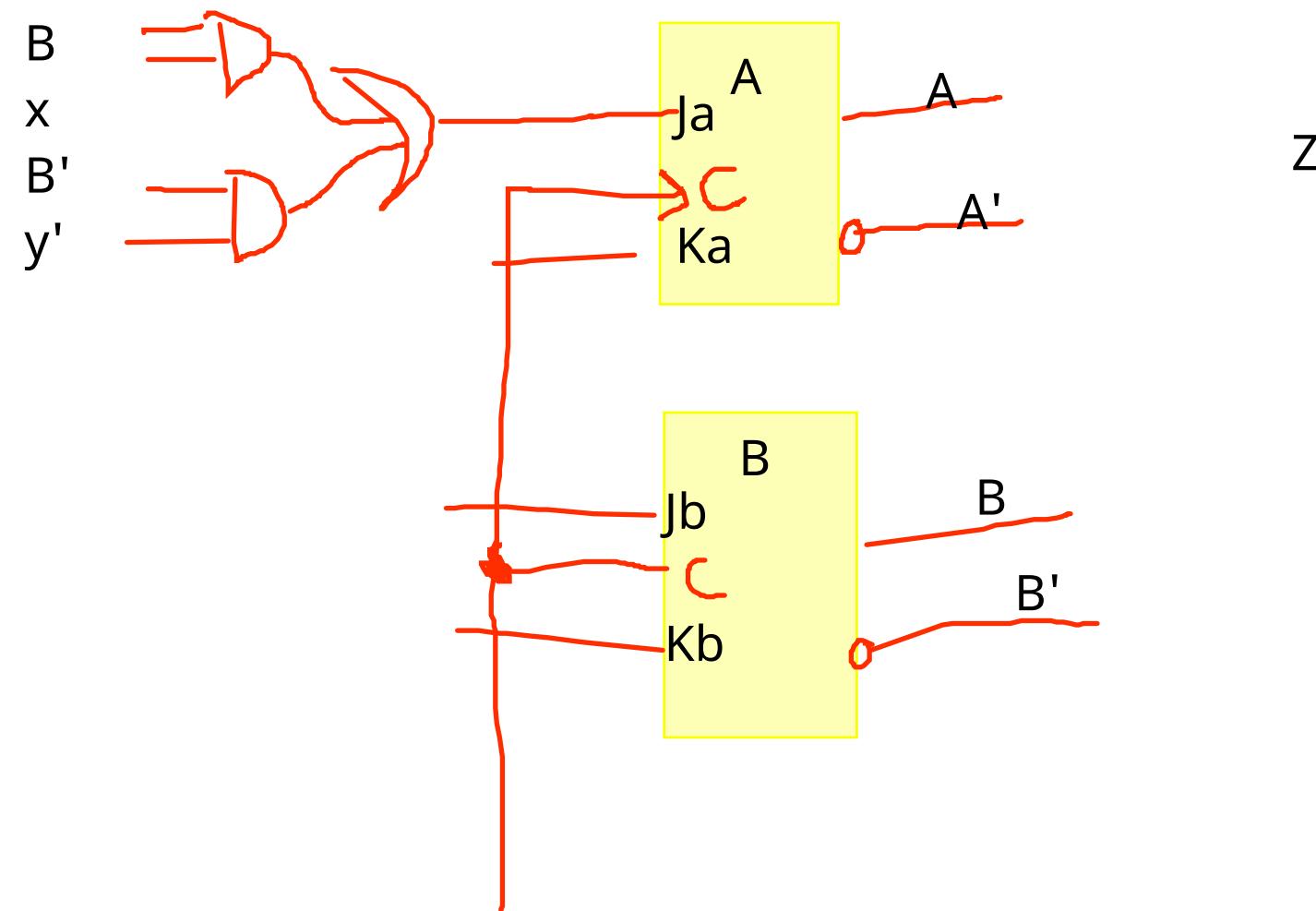


5.10 A sequential circuit has two JK flip-flops A and B, two inputs, x and y; and one output z. The flip-flop input equations and circuit equation are

$$J_A = Bx + B'y', \quad J_B = A'x$$

$$K_A = B'xy', \quad K_B = A + xy', \quad Z = Ax'y' + Bx'y'$$

(a) Draw the logic diagram of the circuit



5.10 A sequential circuit has two JK flip-flops A and B, two inputs, x and y; and one output z. The flip-flop input equations and circuit equation are

$$J_A = Bx + B'y, \quad J_B = A'x$$

$$K_A = B'xy', \quad K_B = A + xy', \quad Z = Ax'y' + Bx'y'$$

(b) Tabulate the state table

5.10 A sequential circuit has two JK flip-flops A and B, two inputs, x and y; and one output z. The flip-flop input equations and circuit equation are

$$\begin{aligned} J_A &= Bx + B'y', \quad J_B = A'x \\ K_A &= B'xy', \quad K_B = A + xy', \quad Z = Ax'y' + Bx'y' \end{aligned}$$

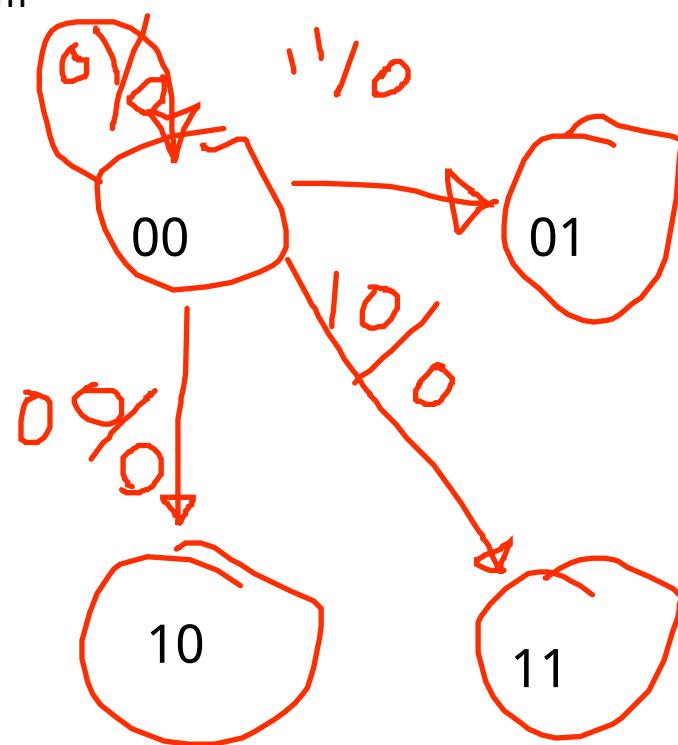
(c) Derive the state equations for A and B

$$Q(t+1) = JQ' + K'Q$$

$$\begin{aligned} A(t+1) &= (Bx + B'y')A' + (B'xy')'A \\ &= A'Bx + A'B'y' + AB + Ax' + Ay \end{aligned}$$

$$B(t+1) = (A'x)B' + (A + xy')'B$$

(d) Draw the corresponding state diagram



5.16 Design a sequential circuit with two D flip-flops A and B, and on input x_in.

(b) When $x_{in} = 0$, the state of the circuit remains the same. When $x_{in} = 1$, the circuit goes through the state transitions from 00 to 11, to 01, to 10, back to 00, and repeat.

Current State		Input	Next State	
A	B	x	A	B
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

ECE124: Discussion

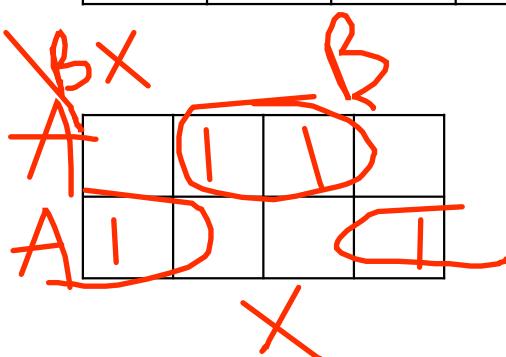
Discussion #12

Yeonsik Noh, PhD

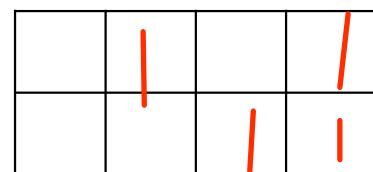
5.16 Design a sequential circuit with two D flip-flops A and B, and on input x_{in} .

(b) When $x_{in} = 0$, the state of the circuit remains the same. When $x_{in} = 1$, the circuit goes through the state transitions from 00 to 11, to 01, to 10, back to 00, and repeat.

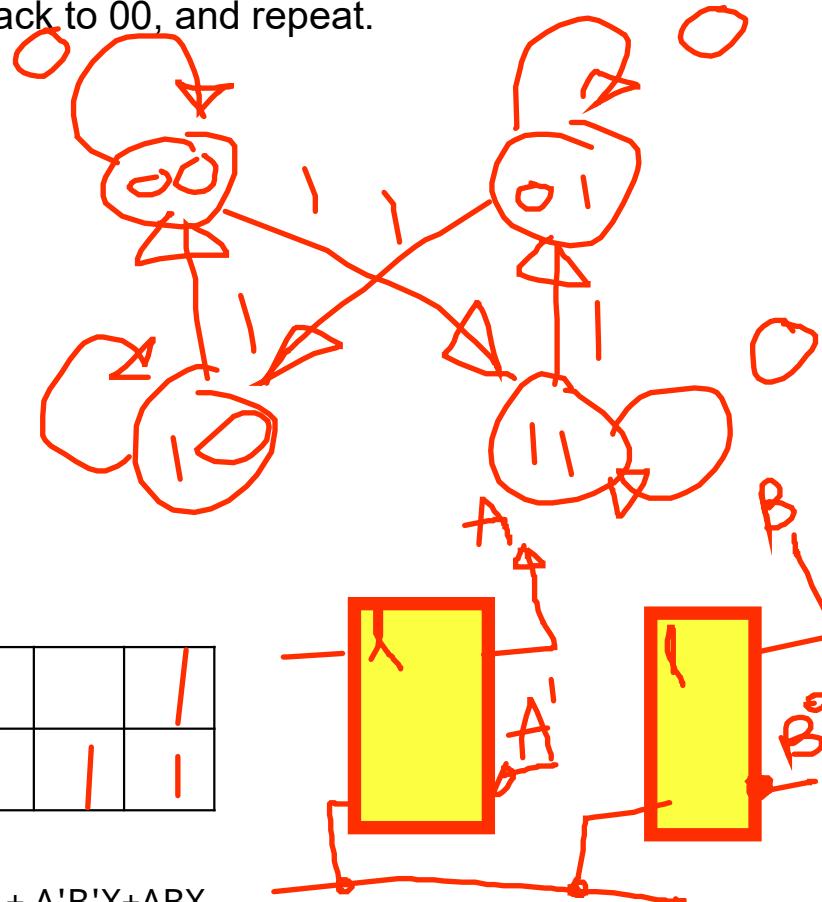
Current State		Input	Next State	
A	B	x	A	B
0	0	0	00	
0	0	1	11	
0	1	0	01	
0	1	1	10	
1	0	0	10	
1	0	1	00	
1	1	0	11	
1	1	1	01	



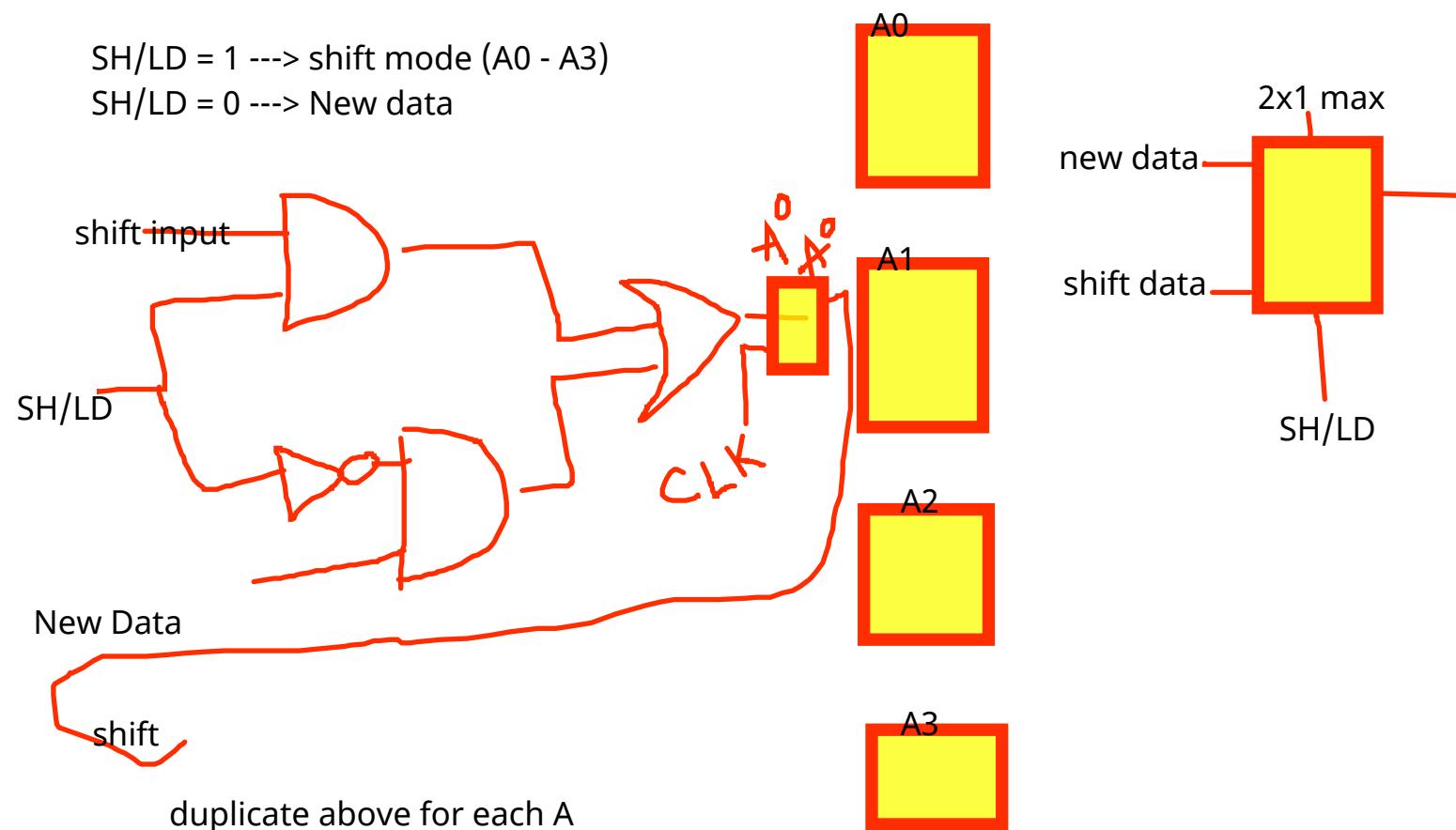
$$A(t+1) = A'X + AX'$$



$$B(t+1) = AB + A'B'X + ABX$$

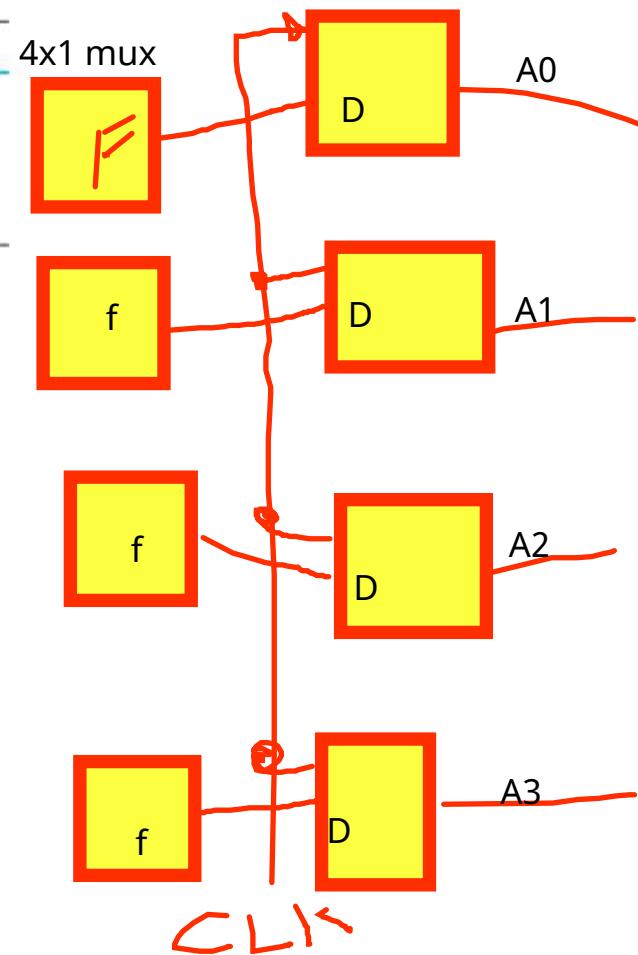
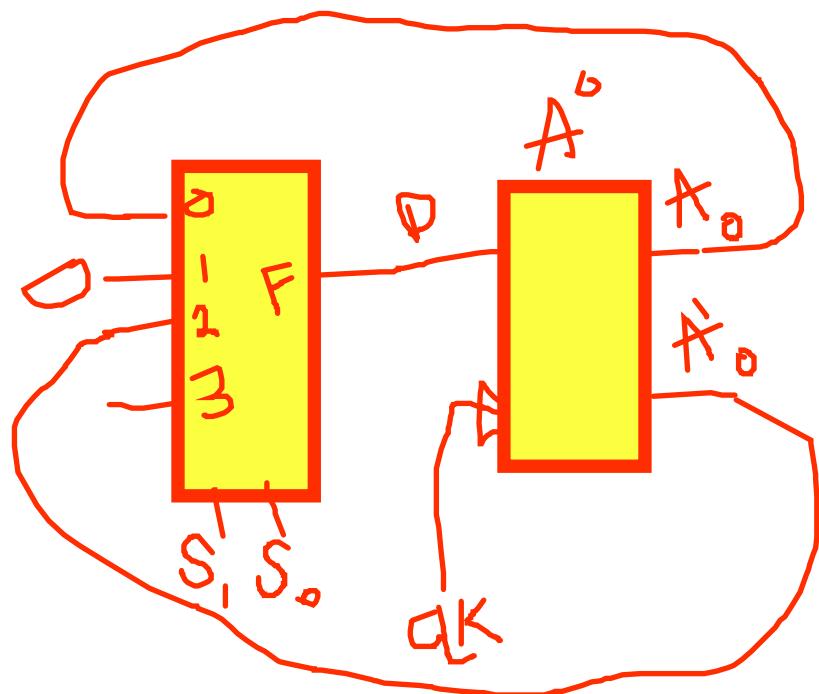


Modified 6.6 Design a four-bit shift register with parallel load using D flip-flops. There is one control input: shift/Load (SH/LD). When $SH/LD = 1$, the content of the register is shifted toward A_3 by one position. New data are transferred into the register when $SH/LD = 0$.

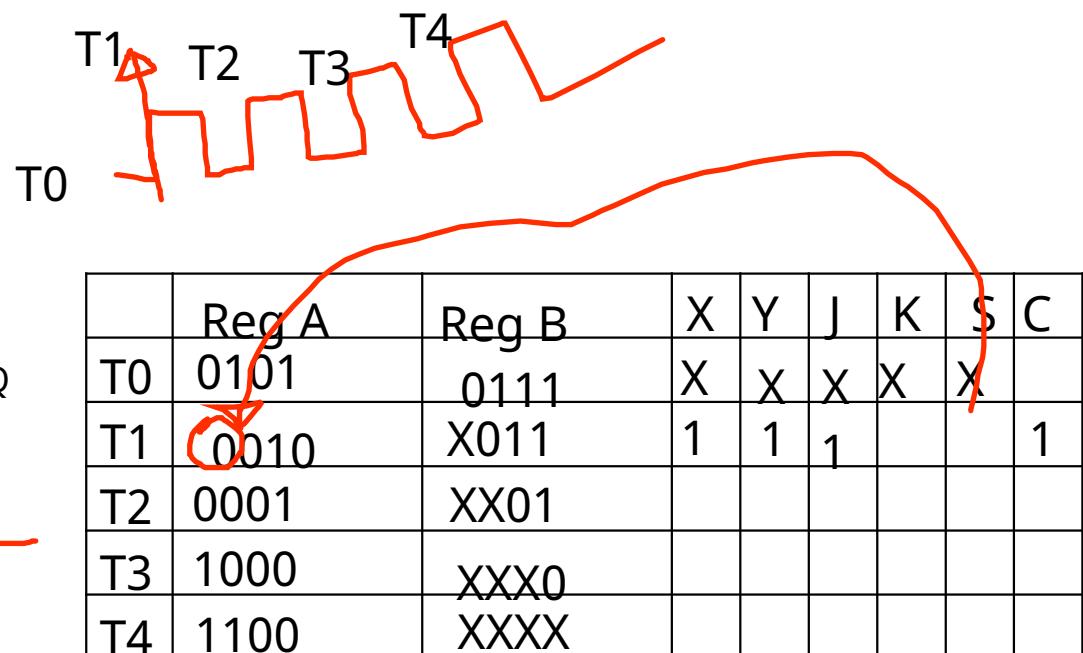
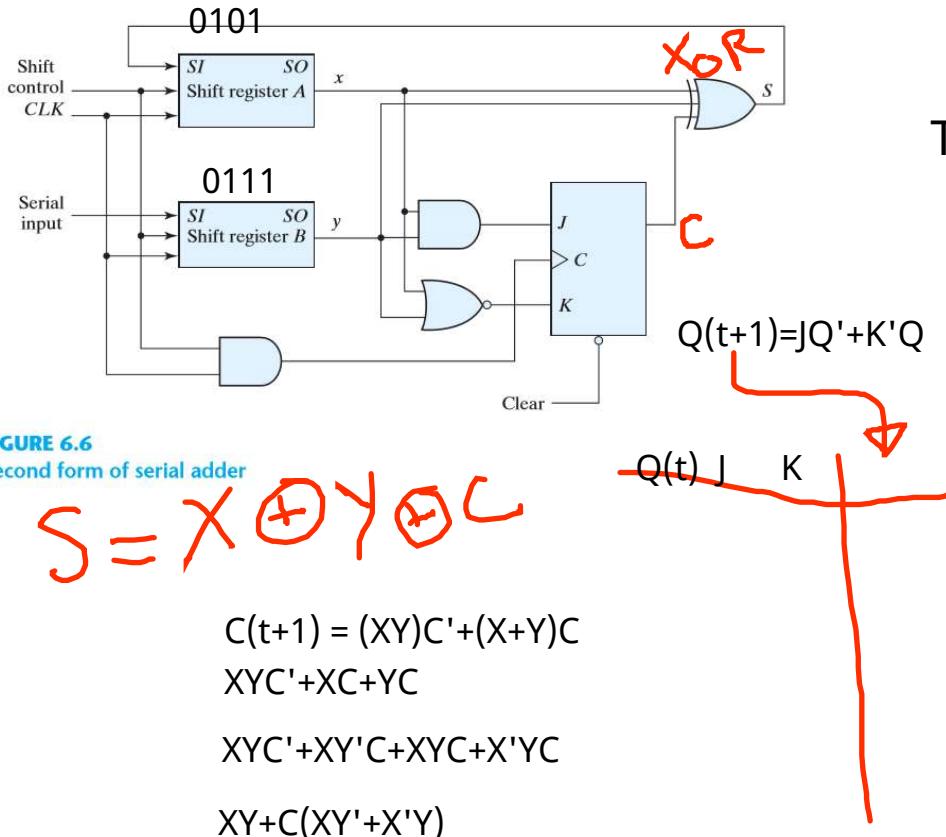


6.7 Draw the logic diagram of a four-bit register with four D flip-flops and four 4×1 multiplexers with mode selection inputs s_1 and s_0 . The register operates according to the following function table.

s_1	s_0	Register Operation
0	0	No change
1	0	Complement the four outputs
0	1	Clear register to 0 (synchronous with the clock)
1	1	Load parallel data

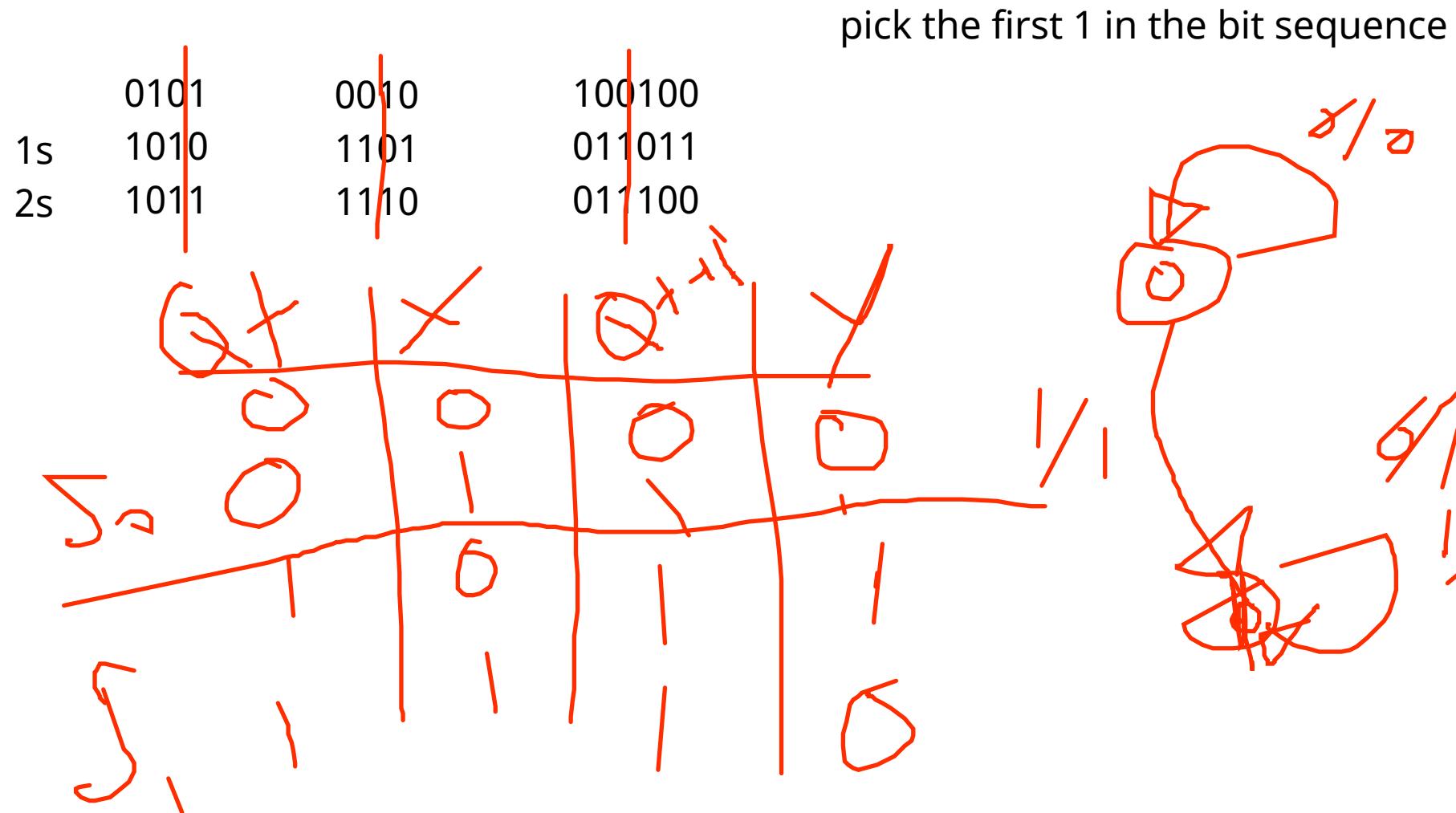


6.8 The serial adder of Fig. 6.6 uses two four-bit registers. Register A holds the binary number 0101 and register B holds 0111. The carry flip-flop is initially reset to 0. List the binary values in register A and the carry flip-flop after each shift.



6.10 Design a serial 2's completer with a shift register and a D flip-flop. The binary number is shifted out from one side and its 2's complement shifted into the other side of the shift register.

- Using the characteristic of 2's complement



6.10 Design a serial 2's completer with a shift register and a D flip-flop. The binary number is shifted out from one side and its 2's complement shifted into the other side of the shift register.

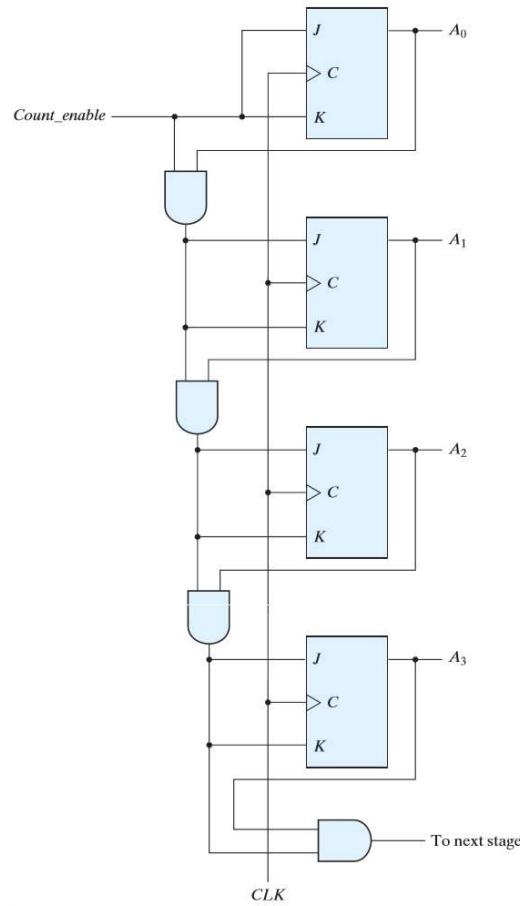
- Using the characteristic of 2's complement

$$Q(t+1) = Q(t) + x$$

$$y = Q(t) \oplus x$$



6.17 Design a four-bit binary synchronous counter with D flip-flops.



$$\begin{aligned} T_0 &= E \\ T_1 &= A_0 E \\ T_2 &= A_1 A_0 E \\ T_3 &= A_2 A_1 A_0 E \end{aligned}$$

$$\begin{aligned} t_0(t+1) &= A_0 \text{ XOR } E \\ t_1(t+1) &= A_1 \text{ XOR } A_0 E \\ t_2(t+1) &= A_2 \text{ XOR } A_1 A_0 E \\ t_3(t+1) &= A_3 \text{ XOR } A_2 A_1 A_0 E \end{aligned}$$

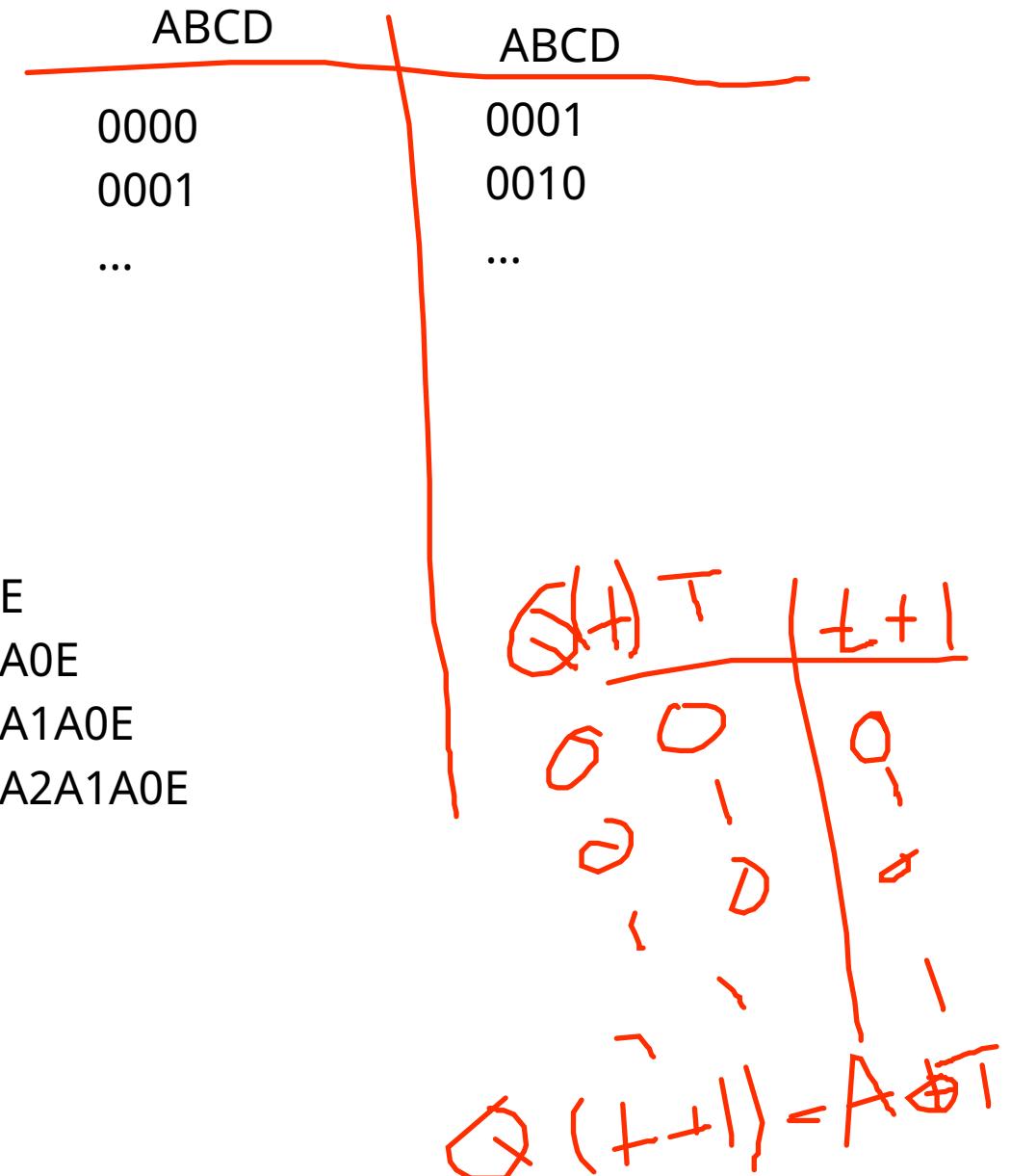
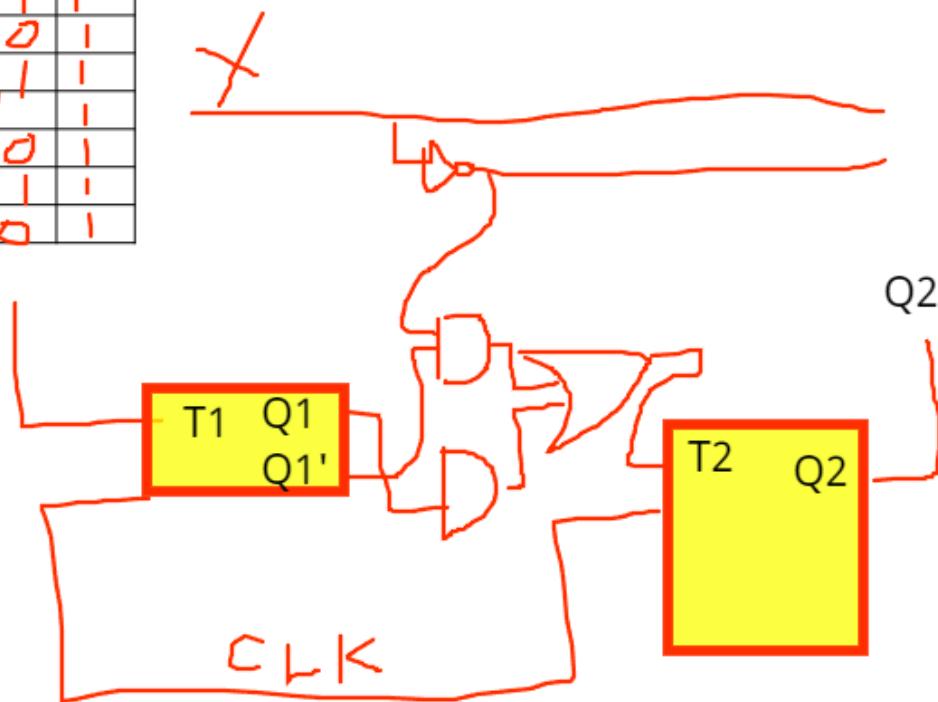
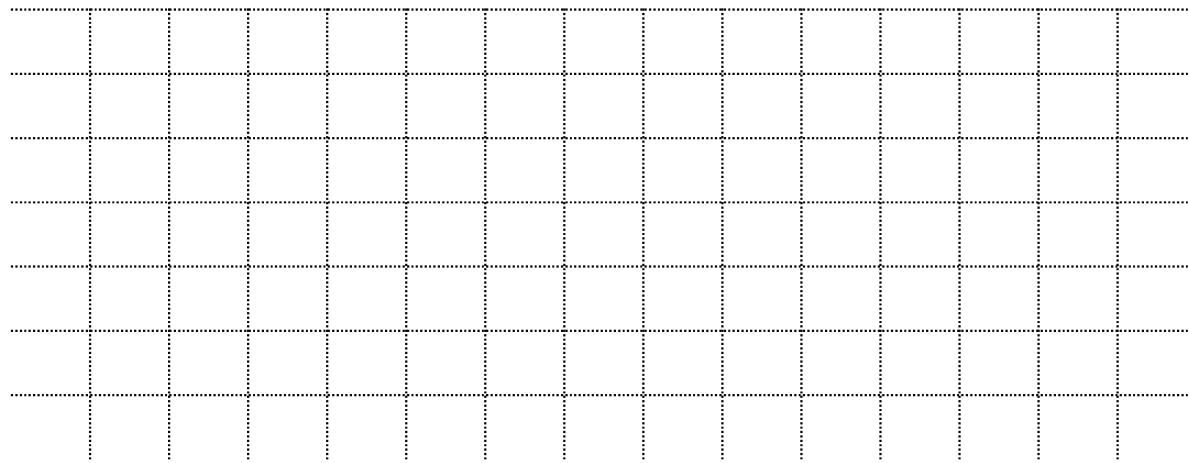


FIGURE 6.12
Four-bit synchronous binary counter

***Up-Down Counter** Using T flip-flops, design a 2-bit up-down counter with external input x to determine whether Up or Down.

Present state		Input	Next state		FF inputs	
B	A	x	B(t+1)	A(t+1)	T _B	T _A
0	0	0	0	1	0	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	0	1	1	1
1	0	0	1	1	1	1
1	0	1	0	0	0	1
1	1	0	0	1	1	1
1	1	1	1	0	0	1





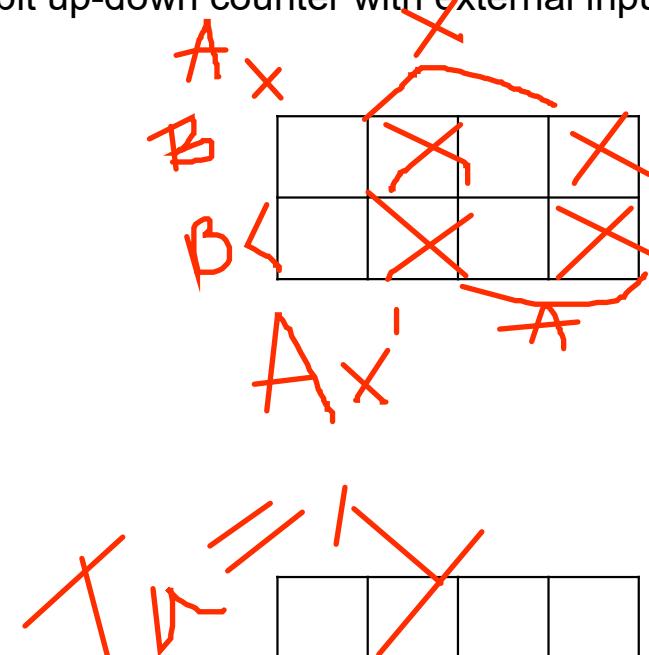
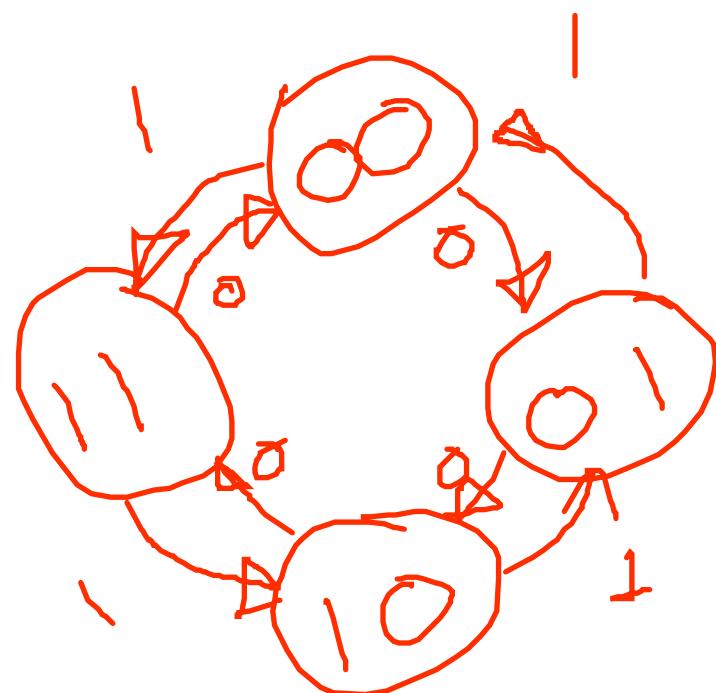
ECE124: Discussion

Discussion #13

Yeonsik Noh, PhD

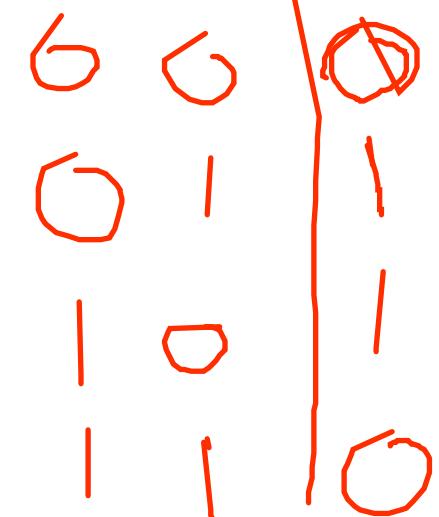
*Up-Down Counter Using T flip-flops, design a 2-bit up-down counter with external input x to determine whether Up or Down.

Present state		Input	Next state		FF inputs	
B	A	x	B(t+1)	A(t+1)	T _B	T _A
0	0	0	0	1	0	1
0	0	1	1	0	1	1
0	1	0	0	0	1	1
0	1	1	1	0	0	1
1	0	0	1	0	0	1
1	0	1	0	1	1	1
1	1	0	0	0	1	1
1	1	1	1	0	0	1



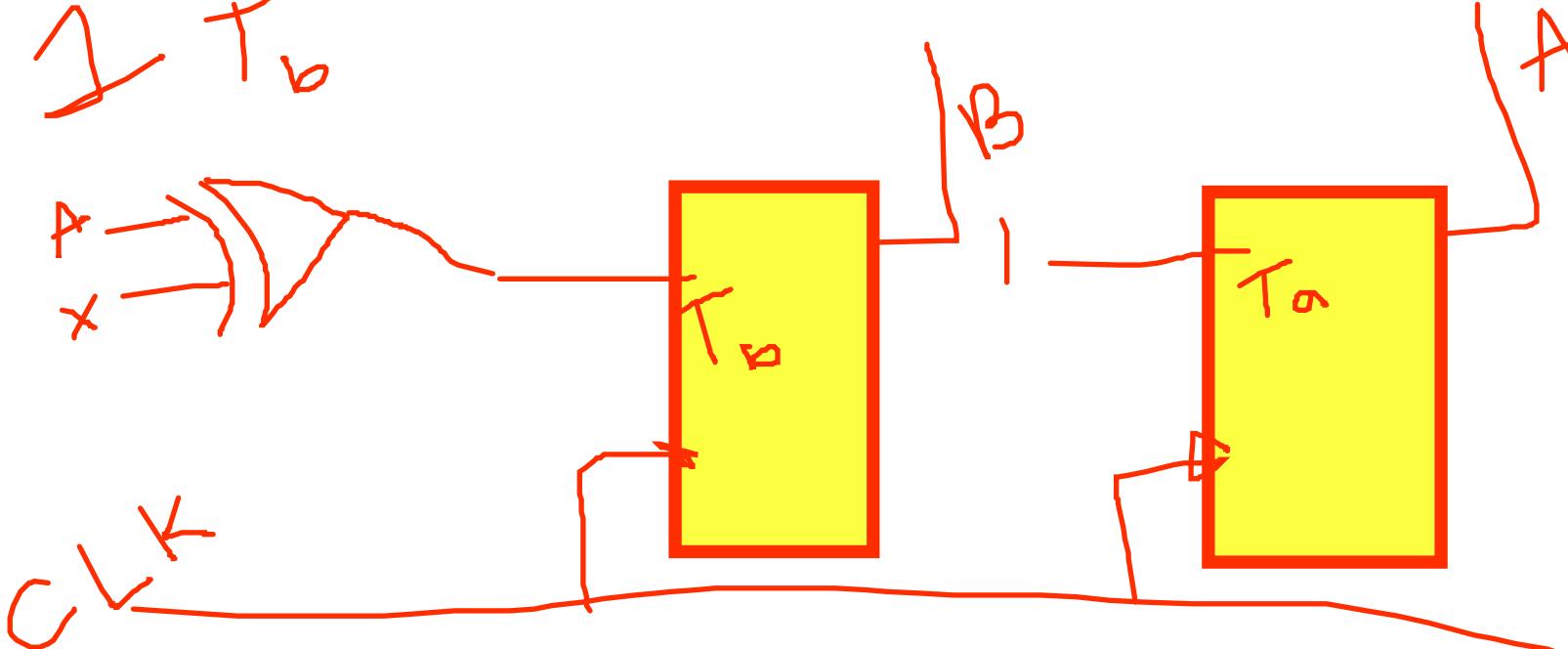
AX'+A'X
A XOR B
T flipflop table

Q(t) Q(t+1) T

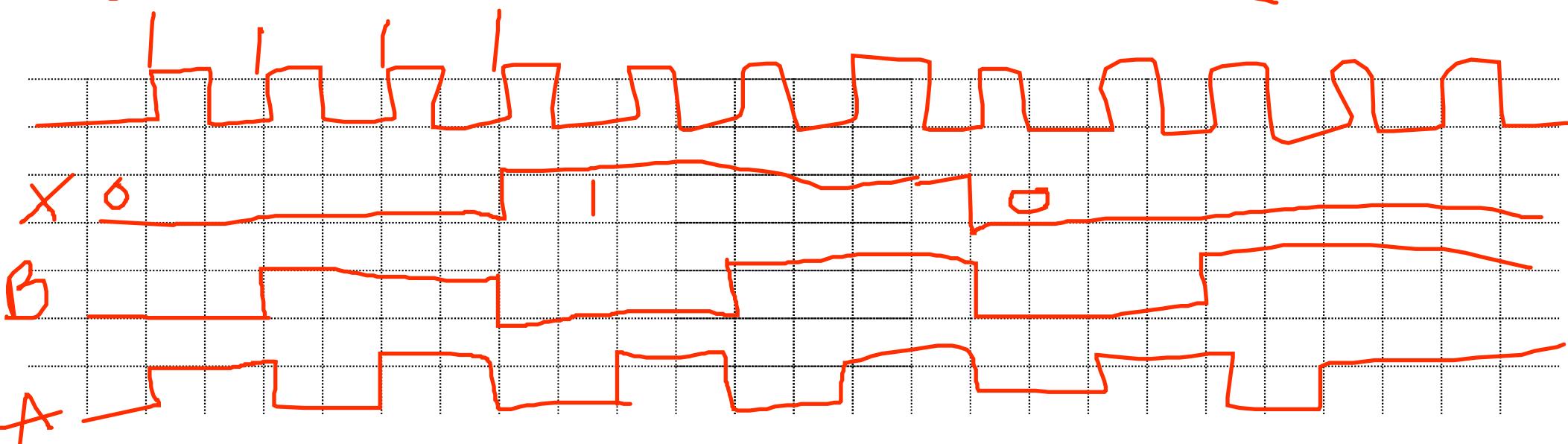


$A \oplus X T_a$

$\bar{A} T_b$

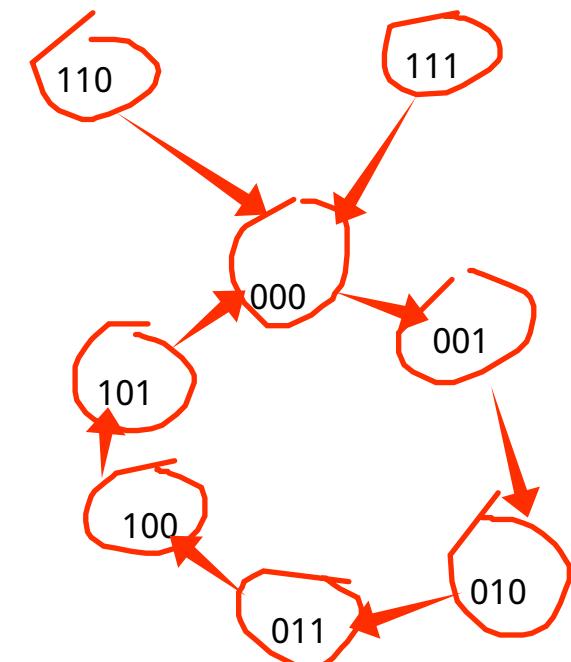
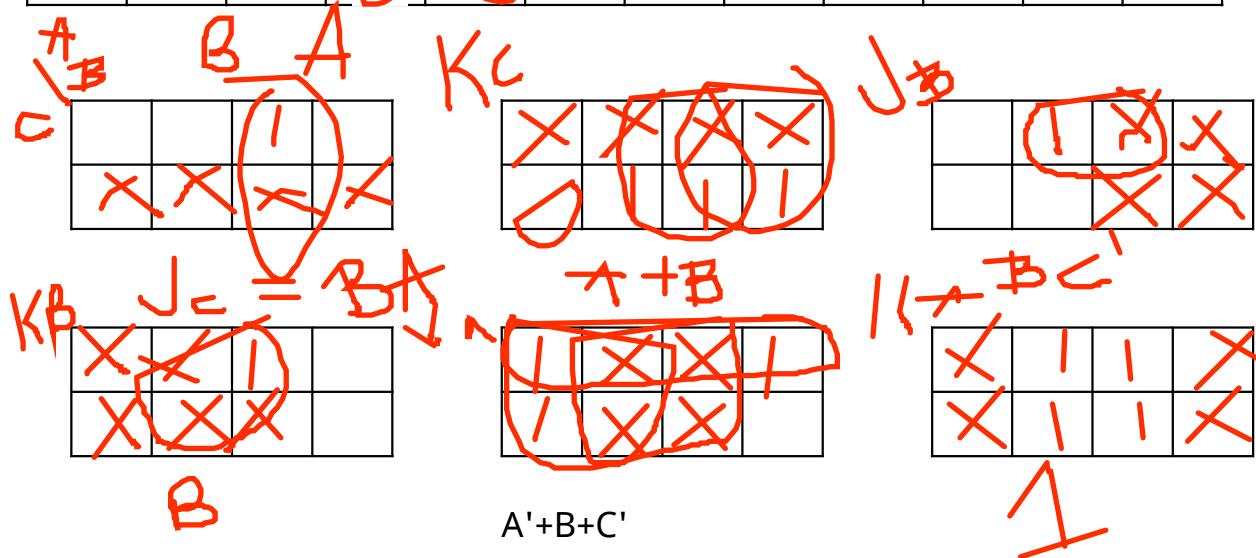


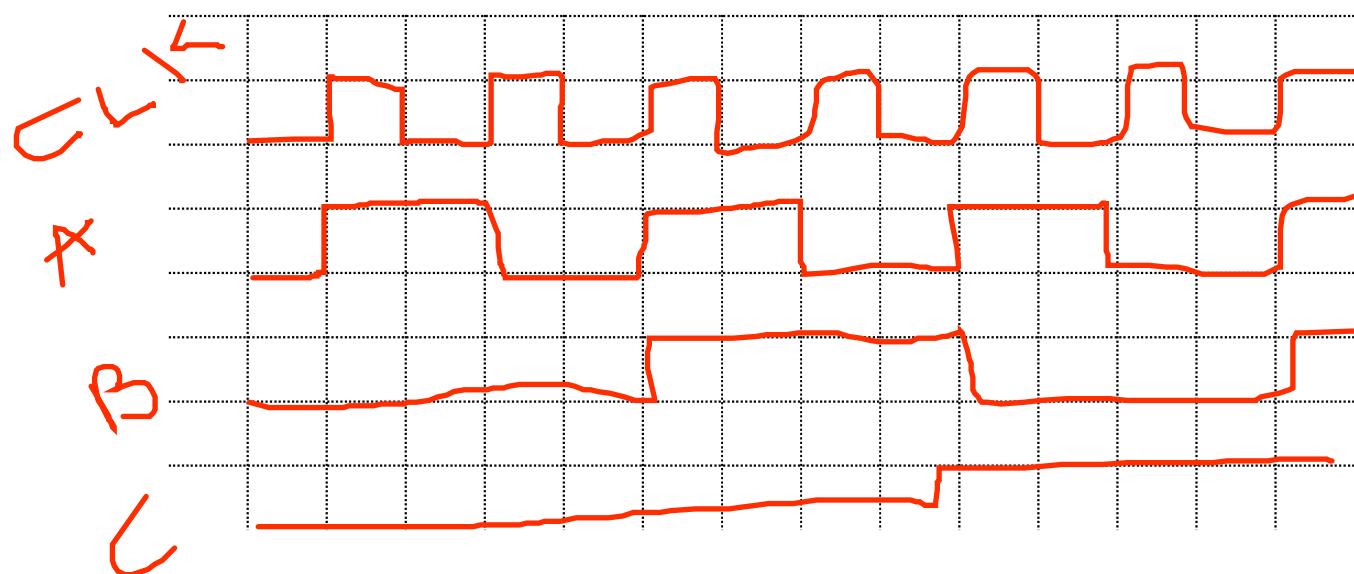
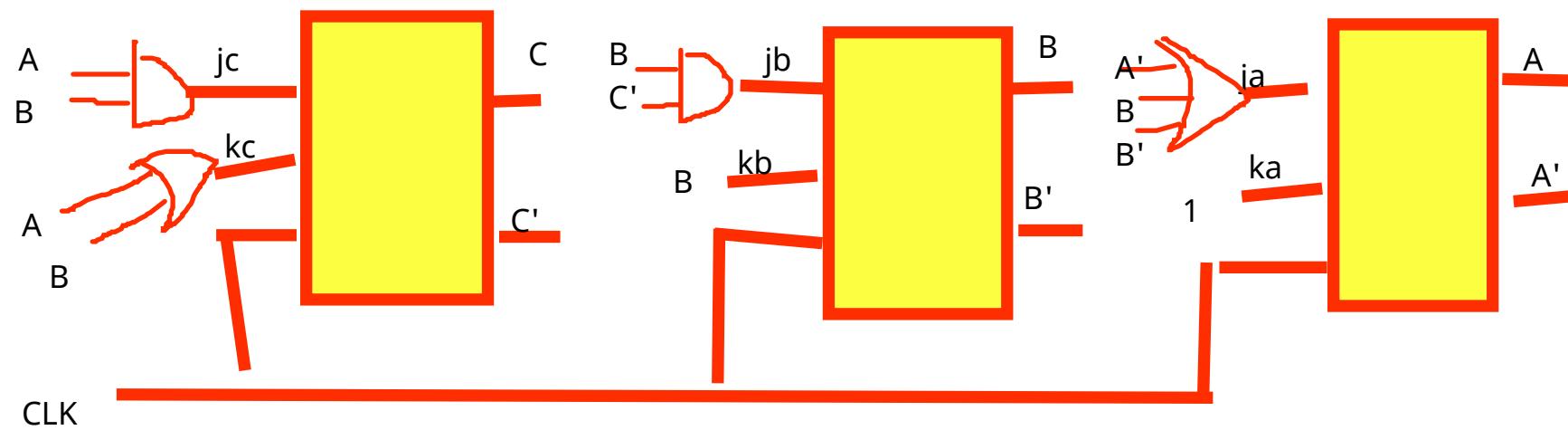
C



*Mod-6 Counter Using JK flip-flops, design a counter with the following repeated binary sequence: 0, 1, 2, 3, 4, 5. In case of 6 and 7, include the self correcting state to 0.

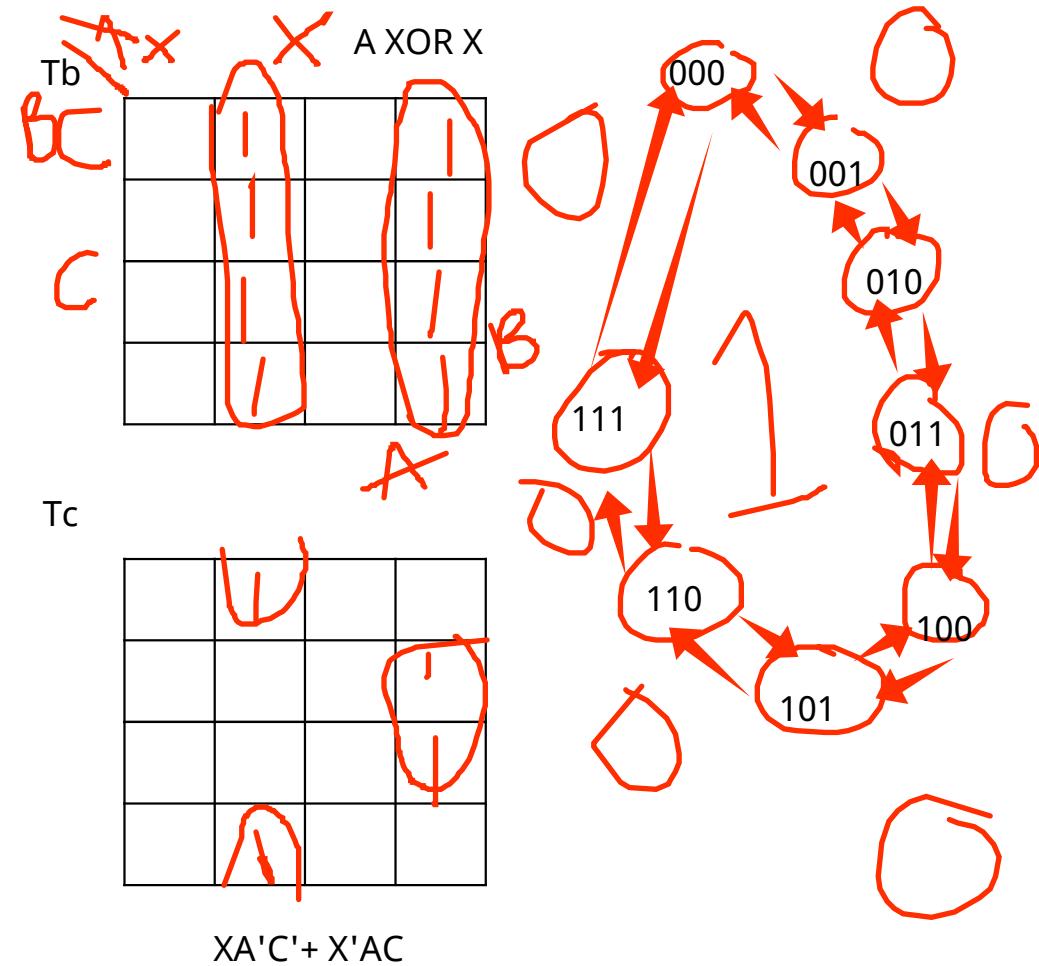
Present State			Next State			Inputs of JK Flip-flops					
C	B	A	C(t+1)	B(t+1)	A(t+1)	J _C	K _C	J _B	K _B	J _A	K _A
0	0	0	0	0	1	X		X	1	X	
0	0	1	0	1	0	X	1	X	X	1	
0	1	0	0	1	1	X	X		1	X	
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X		X	1	X	
1	0	1	0	0	0	X	1		X	1	
1	1	0	0	0	0	X	1	X	1		X
1	1	1	0	0	0	X	1	X	X	X	1



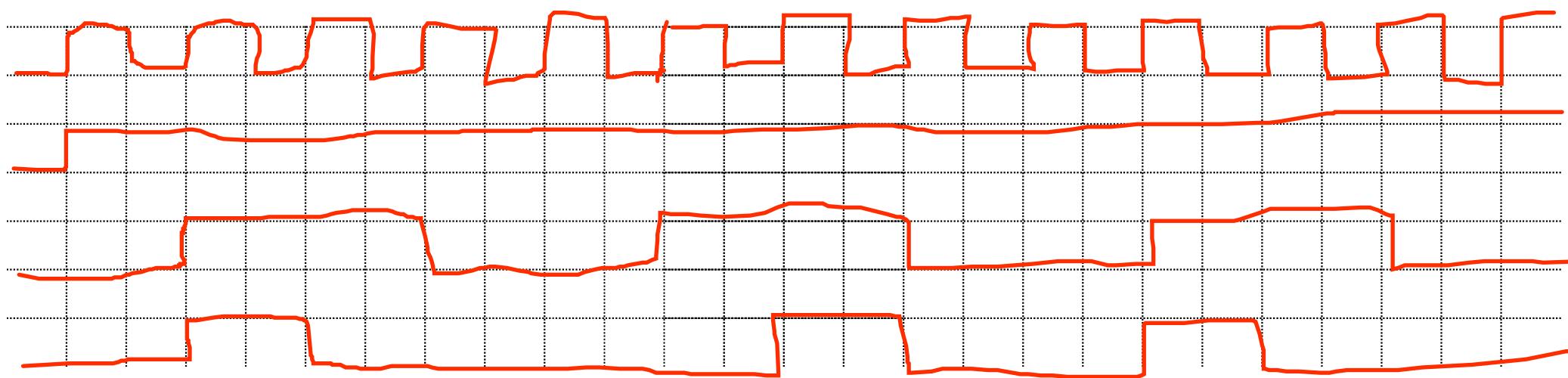
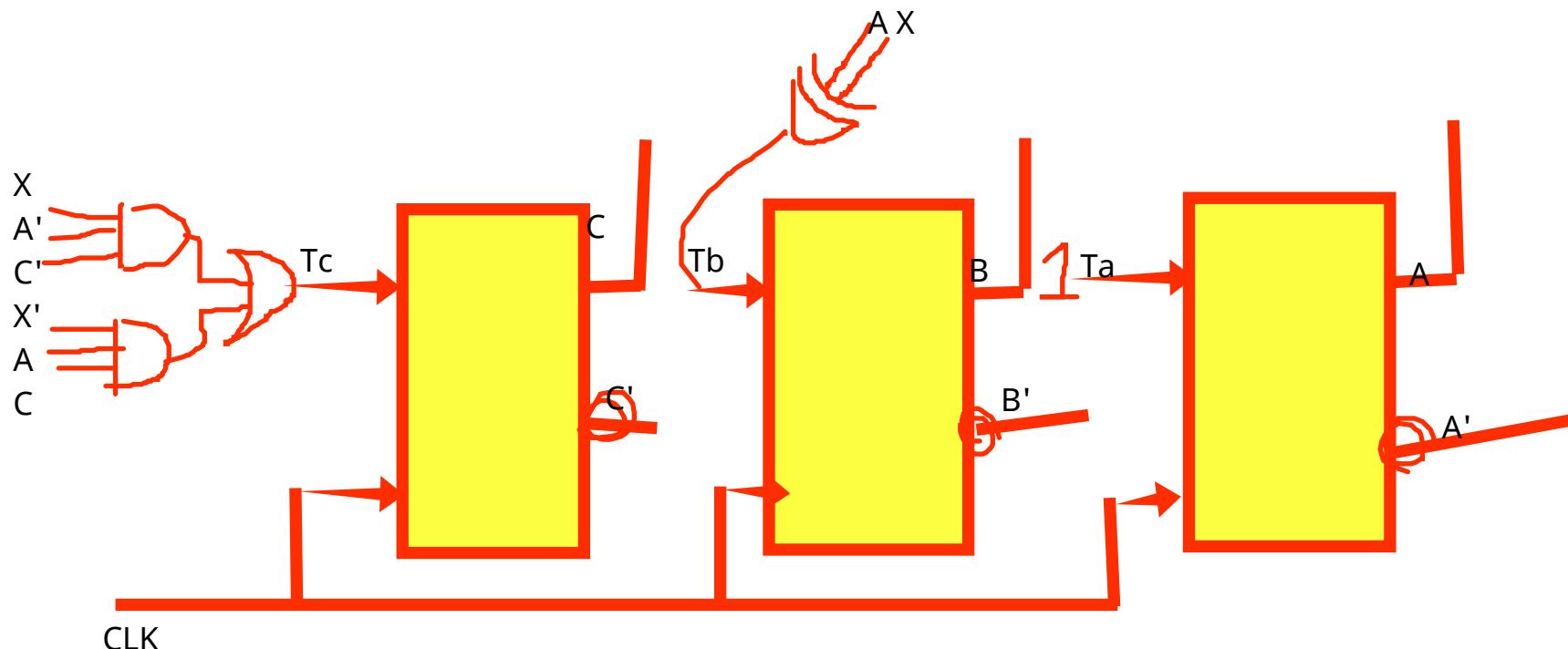


*Up-Down Counter Using T flip-flops, design a 3-bit up-down counter with external input x to determine whether Up or Down.

Present state			Input	Next state			FF inputs		
C	B	A	x	C(t+1)	B(t+1)	A(t+1)	T _C	T _B	T _A
0	0	0	0			1			1
0	0	0	1	1	1	1	1	1	1
0	0	1	0		1		1	1	1
0	0	1	1						1
0	1	0	0		1	1			1
0	1	0	1			1	1	1	1
0	1	1	0	1			1	1	1
0	1	1	1		1				1
1	0	0	0	1		1			1
1	0	0	1		1	1	1	1	1
1	0	1	0	1	1		1	1	1
1	0	1	1						1
1	1	0	0	1	1	1			1
1	1	0	1	1		1	1	1	1
1	1	1	0				1	1	1
1	1	1	1	1	1				1



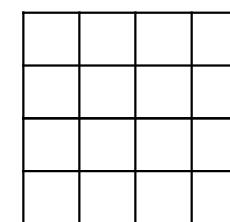
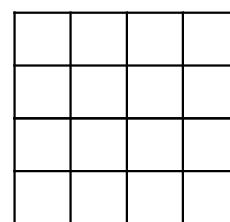
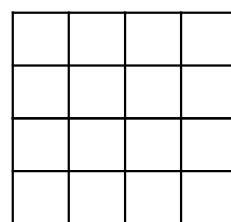
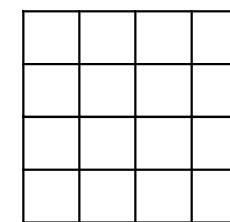
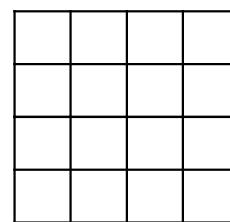
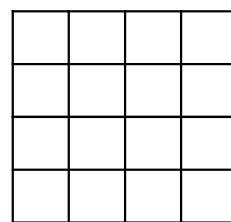
$$T_a = 1$$



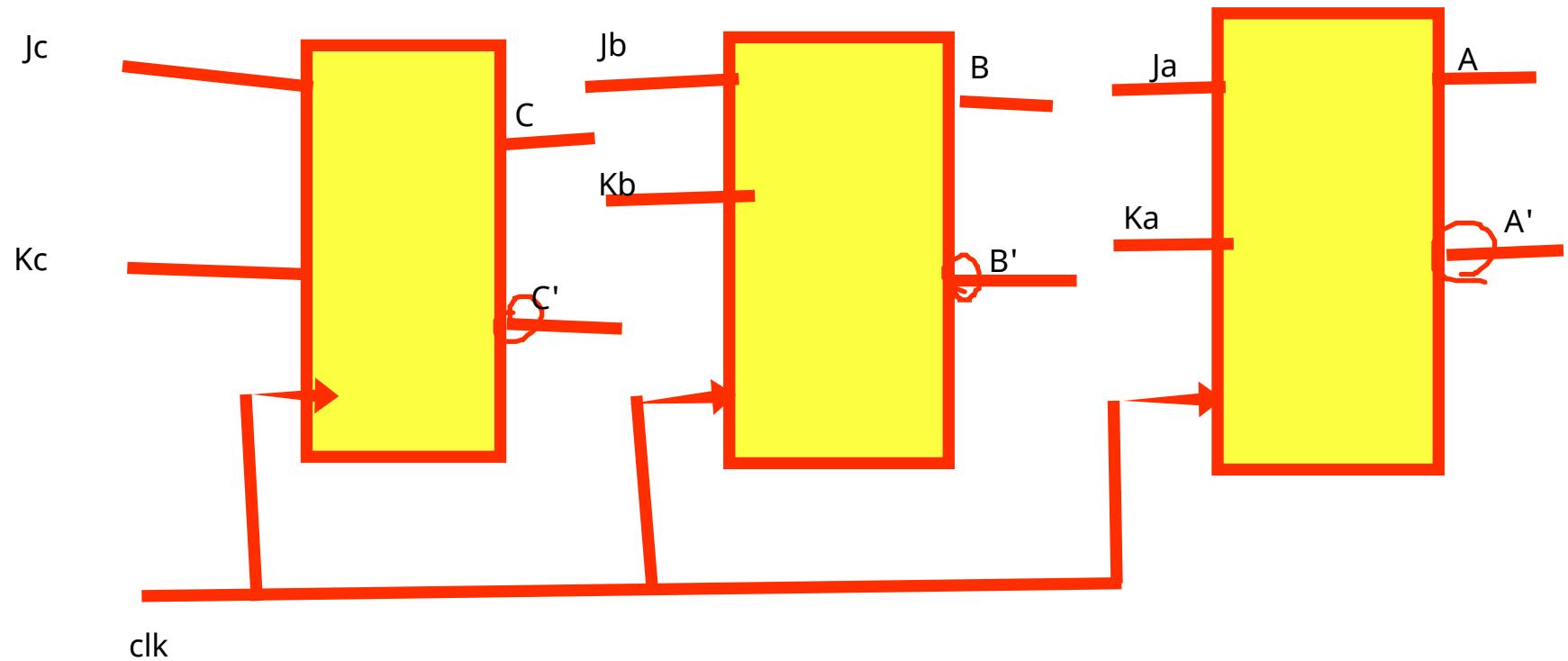
- Design of 3-bit binary up/down counter

Present State	External Input	Next State	Inputs of FFs					
			J _C	K _C	J _B	K _B	J _A	K _A
0 0 0	0	0 0 1	0	X	0	X	1	X
0 0 0	1	1 1 1	1	X	1	X	1	X
0 0 1	0	0 1 0	0	X	1	X	X	1
0 0 1	1	0 0 0	0	X	0	X	X	1
0 1 0	0	0 1 1	0	X	X	0	1	X
0 1 0	1	0 0 1	0	X	X	1	1	X
0 1 1	0	1 0 0	1	X	X	1	X	1
0 1 1	1	0 1 0	0	X	X	0	X	1
1 0 0	0	1 0 1	X	0	0	X	1	X
1 0 0	1	0 1 1	X	1	1	X	1	X
1 0 1	0	1 1 0	X	0	1	X	X	1
1 0 1	1	1 0 0	X	0	0	X	X	1
1 1 0	0	1 1 1	X	0	X	0	1	X
1 1 0	1	1 0 1	X	0	X	1	1	X
1 1 1	0	0 0 0	X	1	X	1	X	1
1 1 1	1	1 1 0	X	0	X	0	X	1

- Design of 3-bit binary up/down counter



- Design of 3-bit binary up/down counter



ECE124: Discussion

Discussion #2

Yeonsik Noh, PhD

Number Systems

<i>Decimal</i>	<i>Binary</i>	<i>Hex</i>	<i>Octal</i>
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	08	10
09	01001	09	11
10	01010	0A	12
11	01011	0B	13
12	01100	0C	14
13	01101	0D	15
14	01110	0E	16
15	01111	0F	17

Image from:

<http://pirun.ku.ac.th/~fscinrn/Nuclear%20Instrument/node119.html>

Conversions between number systems

- Conversion from base r to decimal

3 digits out#

(a₂a₁a₀)₈

$$=(a_0*8^0)+(a_0*8^1)+(a_0*8^2)=(?)_{10}$$

- Conversion from decimal to base r
 - Divide number and successive quotients by r
 - Sequence of remainders is base r number

$$\sum_{i=0}^{P*n} a_i \times r^{i P*n}$$

1.9 Express the following numbers in different number systems.

(a) $(10110.0101)_2$

1.) Decimal

$$(1 \times 2^4) + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^{-2} + 1 \times 2^{-4} = (?)_{10} \quad \text{-- dont have to find on the test,}$$

just use formula

2.) octal

010 110 . 010 100

2 6 . 2 4

3.) Hex

0001 0110 . 0101

1 6 . 5

Decimal	Binary	Hex	Octal
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	08	10
09	01001	09	11
10	01010	0A	12
11	01011	0B	13
12	01100	0C	14
13	01101	0D	15
14	01110	0E	16
15	01111	0F	17

1.9 Express the following numbers in different number systems.

(c) $(26.24)_8$

Binary

26.24

010 110 . 010 100

what if $(26.24)_{16}$

0010 0110 . 0010 0100

Decimal	Binary	Hex	Octal
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	08	10
09	01001	09	11
10	01010	0A	12
11	01011	0B	13
12	01100	0C	14
13	01101	0D	15
14	01110	0E	16
15	01111	0F	17

1.9 Express the following numbers in different number systems.

(d) $(DABA.B)_{16}$

Binary

1101 1010 1011 1010 . 1011

octal

15 12 13 12 . 13

$(25.6875)_{10}$

to binary

10011.1011

$$25/2 = 12 \text{ R} 1 \quad 1 \quad <- \text{ LSB}$$

$$.6875 * 2 = 1 + .375 \quad <- \text{MSB}$$

$$12/2 = 6$$

$$.375 * 2 = 0 + .75$$

$$6/2 = 3$$

$$.75 * 2 = 1 + .5$$

$$3/2 = 1 \text{ R} 1 \quad 1$$

$$.5 * 2 = 1 + .0$$

$$1/2 = 0 \text{ R} 1 \quad 1$$

Decimal	Binary	Hex	Octal
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	08	10
09	01001	09	11
10	01010	0A	12
11	01011	0B	13
12	01100	0C	14
13	01101	0D	15
14	01110	0E	16
15	01111	0F	17

$$\text{signed}(101)_2 = (-1)_{10}$$

- **Computer arithmetic of signed numbers**

signed magnitude --> left most (MSB)

- **N's complement**

- Given a number \mathbf{N} in base r having n digits, the $(r-1)$'s complement of N is defined as:

$$(r^n - 1) - \mathbf{N}$$

- Given a number \mathbf{N} in base r having n digits, the (r) 's complement of N is defined as:

$$(r^n) - \mathbf{N}$$

$$-(1*2^2)+(1*2^1)+(1*2^0)$$

Signed Numbers with Complements

- 3-bit representations of positive and negative numbers:

$$(111)_2 = -(001)_2$$

$$3 = 011$$

$$-3 = 101$$

$$2 = 010$$

$$-2 = 110$$

$$101 = (-1)_{10}$$

Decimal	2's Complement	1's Complement	Signed Magnitude
0/11	+3	0/11	0/11
0/10	+2	0/10	0/10
0/01	+1	0/01	0/01
0/00	0	0/00	0/00
-0	X	1/11	1/00
-1	1/11	1/10	1/01
-2	1/10	1/01	1/10
-3	1/01	1/00	1/11
-4	1/00	X	X

- Bit sequence does not reveal coding—context matters!
- Left most bit is sign bit in 2's complement—automatically!

[1.14](#) Obtain the 1's and 2' complements of the following binary numbers:

(d) 10101010

01010101

01010110

(e) 10100101

01011010

01011011

(f) 11111111

00000000

00000001

- Convert the following decimal numbers to 8-bit signed binary numbers and represent them in 8-bit 2's complement representation.

(a) -27

11100101

(b) 27

00011011

(c) -52

00110100
11001100



$$27/2 = 13R1$$

$$13/2=6R1$$

$$6/2=3$$

$$3/2=1R1$$

$$1/2=0R1$$

$$52/2=26 \quad \text{--LSB}$$

$$26/2=13$$

$$13/2=6R1$$

$$6/2=3$$

$$3/2=1R1$$

$$1/2=0R1$$

1.18 Perform subtraction on the given unsigned binary numbers using the 2's complement of the subtrahend. Where the result should be negative, find its 2's complement and affix a minus sign.

(a) $10011 - 10010$

$$\begin{array}{r} 010011 \\ + 101110 \\ \hline 000001 \end{array}$$

$$19-18 = 1$$

(c) $1001 - 110101$

$$\begin{array}{r} 00001001 \\ 00110101 \end{array}$$

$$\begin{array}{r} 00001001 \\ + 11001011 \\ \hline 11010100 \end{array}$$

-00101100



ECE124: Discussion

Discussion #3

Yeonsik Noh, PhD

1.23 Represent the unsigned decimal numbers 791 and 658 in BCD and then show the steps necessary to form their sum. 8421 code

0111 1001 0001 0110 0101 1000
791 658

$$\begin{array}{r} 0111\ 1001\ 0001 \\ +\ 0110\ 0101\ 1000 \\ \hline \end{array}$$

$$\begin{array}{r} 1110\ 1110\ 1001 \\ +\ 1101\ 0110 \\ \hline \end{array}$$

$$\begin{array}{r} 0110 \\ 0001\ 0101\ 0100\ 1001 \\ \hline \end{array}$$

1.24 Formulate a weighted binary code for the decimal digits, using the following weights. (a) 6,3,1,1 (b) 6,4,2,1

Decimal	6 3 1 1	6 4 2 1
0	0000	0000
1	0001	0001
2	0011	0010
3	0100	0011
4	0101	0100
5	0111	0101
6	1000	0110
7	1001	0111
8	1011	1000
9	1111	1001

Write the expression “G. Boole” in ASCII using 8-bit code (including the period and space). Treat the leftmost bit of each character as a parity bit (odd parity).

G 1100 0111

. 1010 1110

sp 0010 0000

B 1100 0010

o 1110 1111

o 1110 1111

I 1110 1100

e 1110 0101

USASCII code chart

b ₇ b ₆ b ₅				0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1	
b ₄	b ₃	b ₂	b ₁	Column Row	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	8	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	HT	EM)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	:	K	[k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	l
1	1	0	1	13	CR	GS	-	=	M]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	-	o	DEL

1.33 The state of a 12-bit register is **100010010111**. what is its content if it represents:

(d) A binary number?

2199

$$2^{11} + 2^7 + 2^4 + 2^2 + 2^1 + 2^0 = 2199$$

1000 8

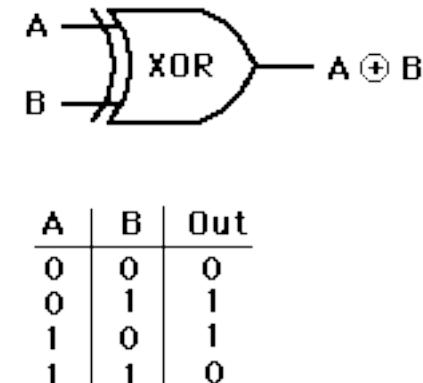
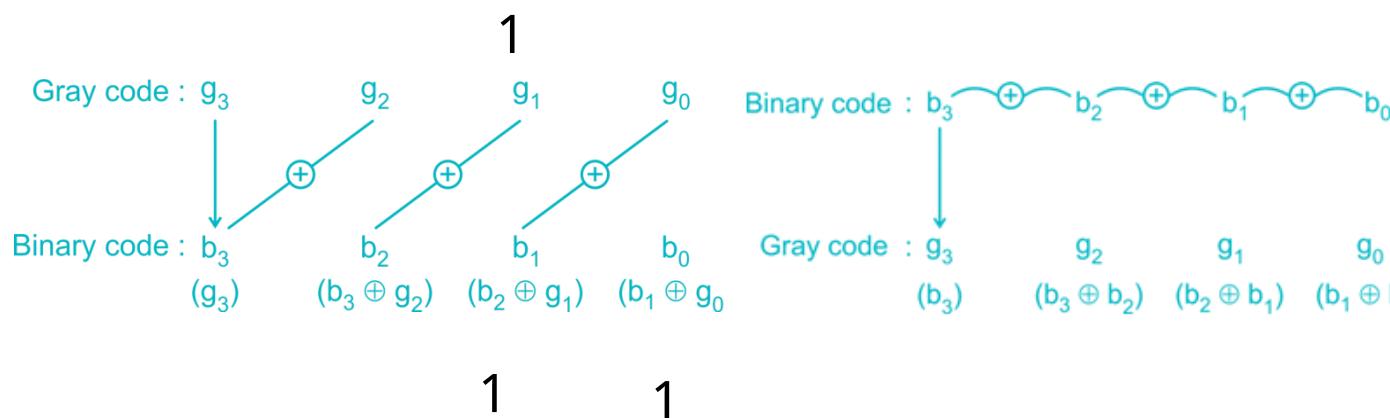
1001 7

871

0111 1

Gray code

Binary				Gray Code			
b ₃	b ₂	b ₁	b ₀	g ₃	g ₂	g ₁	g ₀
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0



- The state of a 12-bit register is **100010010111**. what is its content if it represents a 12-bit Gray code?

100010010111

111

Image from:

<http://hydrogen.physik.uni-wuppertal.de/hyperphysics/hyperphysics/hbase/electronic/xor.html>

<https://www.geeksforgeeks.org/digital-logic-code-converters-binary-gray-code/>

<https://testbook.com/blog/conversion-from-grav-code-to-binary-code-and-vice-versa/>

<https://ncalculators.com/digital-computation/binary-gray-code-converter.htm>

2.1 Demonstrate the validity of the following identities by means of truth tables:

or function

(b) The distributive law: $x + yz = (x + y)(x + z)$

x	y	z	$y \cdot z$	$x + yz$	$x + y$	$x + z$	$(x + y)(x + z)$
0	0	0	0	0	1	1	1
0	0	1	0	1	1	1	1
0	1	0	0	1	1	1	1
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

2.1 Demonstrate the validity of the following identities by means of truth tables:

(c) The distributive law: $x(y + z) = xy + xz$

x	y	z	$y+z$	$x(y+z)$	xy	xz	$xy+xz$
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	0	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

2.1 Demonstrate the validity of the following identities by means of truth tables:

(d) The associative law: $x + (y + z) = (x + y) + z$

x	y	z	$y+z$	$x+(y+z)$	$x+y$	$(x+y)+z$
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	1	1	1
0	1	1	1	1	1	1
1	0	0	0	0	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

2.1 Demonstrate the validity of the following identities by means of truth tables:

(e) The associative law: $x(yz) = (xy)z$

x	y	z	yz	$x(yz)$	xy	$(xy)z$
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	1	1	0	0
1	0	0	0	0	1	1
1	0	1	0	0	1	1
1	1	0	0	0	1	1
1	1	1	1	1	1	1

ECE124: Discussion

Discussion #4

Yeonsik Noh, PhD

- Simplifying Boolean expression:

(a) $xy + (xy)'$

$$xy + (xy)'$$

$$xy + x'y + y'$$

$$(x+x')(y+x') + y'$$

$$y + x' + y'$$

$$1 + x' + 1$$

$$1$$

(b) $xy + x'y'$

$$(xy+x')(xy+y')$$

$$\{(x+x')(y+x')\}\{(x+y')(y+y')\}$$

$$(1(y+x'))((x+y)1)$$

$$(y+x')(x+y')$$

$$(yx+x'x)+(yy'+x'y')$$

$$xy + x'y'$$

2.8 Find the complement of $F = wx + yz$; then show that $FF' = 0$ and $F + F' = 1$.

$$(wx+yz)(wx+yz)'$$

$$(wx+yz)\{(w'+x')(y'+z')\}$$

$$\{wx(w'+x')(y'+z')\} + \{yz(w'+x')(y'+z')\}$$

$$0 + 0$$

$$wx+yz+\{(wx)'(yz)'\}$$

$$wx + \{(yz+(wx)')*(yz+(yz'))\}$$

$$wx+yz+(wx)'$$

$$1+yz$$

$$1$$

2.9 Find the complement of the following expression.

(a) $F(x,y) = xy' + x'y$

$xy' + x'y$

$(xy' + x'y)'$

$(xy')'(x'y)'$

$(x'+y)(x+y')$

$(x'+y)x + (x'+y)y'$

$x'x + yx + y'x'(y'y)$

$xy + x'y'$

2.9 Find the complement of the following expression.

(b) $F(a,b,c) = (a + c)(a + b')(a' + b + c')$

$$\{(a+c)(a+b')(a'+b+c')\}$$

$$(a+c)' + (a'b) + (ab'c)$$

$$a'c' + a'b + ab'c$$

$$a'c'(b+b') + a'b(c+c') + ab'c$$

$$a'bc' + a'b'c' + a'bc + ab'c$$

$$a'bc' + a'b'c' + a'bc + ab'c$$

$$m_2 + m_0 + m_3 + m_5$$

$$f' = 1$$

2.9 Find the complement of the following expression.

(c) $F(x,y,z,v,w) = z + z'(v'w + xy)$

$$f' = \{z + z'(v'w + xy)\}$$

$$z' \{z'(v'w + xy)\}'$$

$$F' = z' (z + v + w')(z + x' + y')$$

$$z' + 0$$

2.10 Given the Boolean functions F_1 and F_2 , show that

(a) The Boolean function $E = F_1 + F_2$ contains the sum of the minterms of F_1 and F_2 .

$$f_1 = \text{or} \Rightarrow f_1(xy) = x+y$$

$$f_2 = \text{and} \Rightarrow f_2(xy) = xy$$

$$E_1 = f_1 + f_2 = x+y+xy$$

$$x(y+y') + y(x+x') + xy$$

$$xy+xy'+x'y$$

$$m_3 + m_2 + m_1$$

$$\text{sum}(123)$$

$$E_2 = f_1 * f_2$$

$$(x+y)xy$$

$$x(xy) + y(xy)$$

$$xxy + xyy$$

$$xy+xy = xy = m_3$$

2.10 Given the Boolean functions F_1 and F_2 , show that

(b) The Boolean function $E = F_1F_2$ contains only the minterms that are common to F_1 and F_2 .

Previous slide