Midterm [Part 2]

2019.10.20.

Hao Zhiyu

Outline

1. Logic circuit
2. Optimization

Theorems

(a) $x + 0 = x$;	(b) $x \cdot 0 = 0$;	(theorem 1)
(a) $x + x' = 1$;	(b) $x \cdot x' = 0$;	(theorem 2)
(a) $x + x = x$;	(b) $x \cdot x = x$;	(theorem 3)
(a) $x + 1 = 1$;	(b) $x \cdot 1 = x$;	(theorem 4)
$(x')' = x$;		(involution)

(a) $x + y = y + x$;	(b) $xy = yx$;	(commutative)
(a) $x + (y + z) = (x + y) + z$;	(b) $x(yz) = (xy)z$;	(associative)
(a) $x(y + z) = xy + xz$;	(b) $x + yz = (x+y)(x+z)$;	(distributive)
(a) $x + xy = x$;	(b) $x(x + y) = x$;	(absorption)
(a) $xy + xy' = x$;	(b) $(x + y)(x + y') = x$	(theorem 5)
(a) $x + x'y = x + y$	(b) $x(x' + y) = xy$	(theorem 6)

Hw2

9. Problem 6.4, using both algebraic methods and K-map. (10 points)

6.4 Perform two-level logic size optimization for F(a,b,c) = a + a'b'c + a'c using a K-map.

F(a,b,c) = a+a'c = a + ac + a'c = a + (a+a') c = a + c.

Read through the question carefully!!!

Boolean Algebra

- De Morgan's Law
- Minterm and Maxterm

			Minterms		Maxterms	
x	y	z	Term	Designation	Term	Designation
0	0	0	x'y'z'	m_0	x+y+z	M_0
0	0	1	x'y'z	m_1	x+y+z'	M_1
0	1	0	x'yz'	m_2	x+y'+z	M_2
0	1	1	x'yz	m_3	x+y'+z'	M_3
1	0	0	xy'z'	m_4	x'+y+z	M_4
1	0	1	xy'z	m_5	x'+y+z'	M_5
1	1	0	xyz'	m_6	x'+y'+z	M_6
1	1	1	xyz	m_7	x'+y'+z'	M_7

(a) $(x + y)' = x'y'$

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

For the final result, it is better to simplify the answer.

Find Expression

- First to minterms
- Know how to transfer to maxterms
- Don’t Care Situation

Truth Table

x	y	z	F1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Sum-of-minterms

$$F1 = x'y'z + xy'z' + xy'z + xyz' + xyz$$

$$F1 = m_1+m_4+m_5+m_6+m_7$$

$$F1 = \Sigma\,(1, 4, 5, 6, 7)$$

Product-of-maxterms

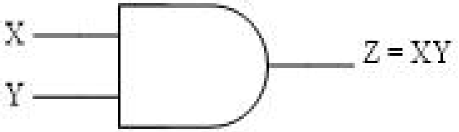
$$F1 = (x+y+z) \bullet (x+y'+z) \bullet (x+y'+z')$$

$$F1 = M_0 \bullet M_2 \bullet M_3$$

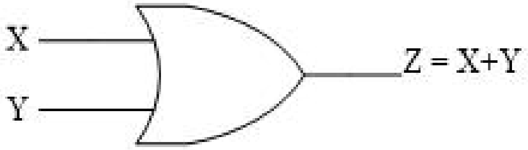
$$F1 = \Pi\,(0, 2, 3)$$

Logic gates

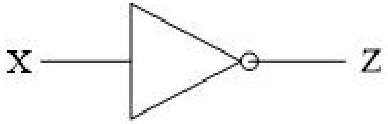
- 1 gate 1 delay (invertor ignored, XOR treated as usual)
- 2 transistors (size) for each gate input



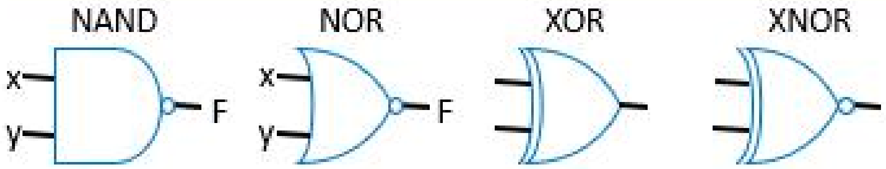
2-input AND gate



2-input OR gate



NOT gate/Inverter



x	y	F
0	0	1
0	1	1
1	0	1
1	1	0

x	y	F
0	0	1
0	1	0
1	0	0
1	1	0

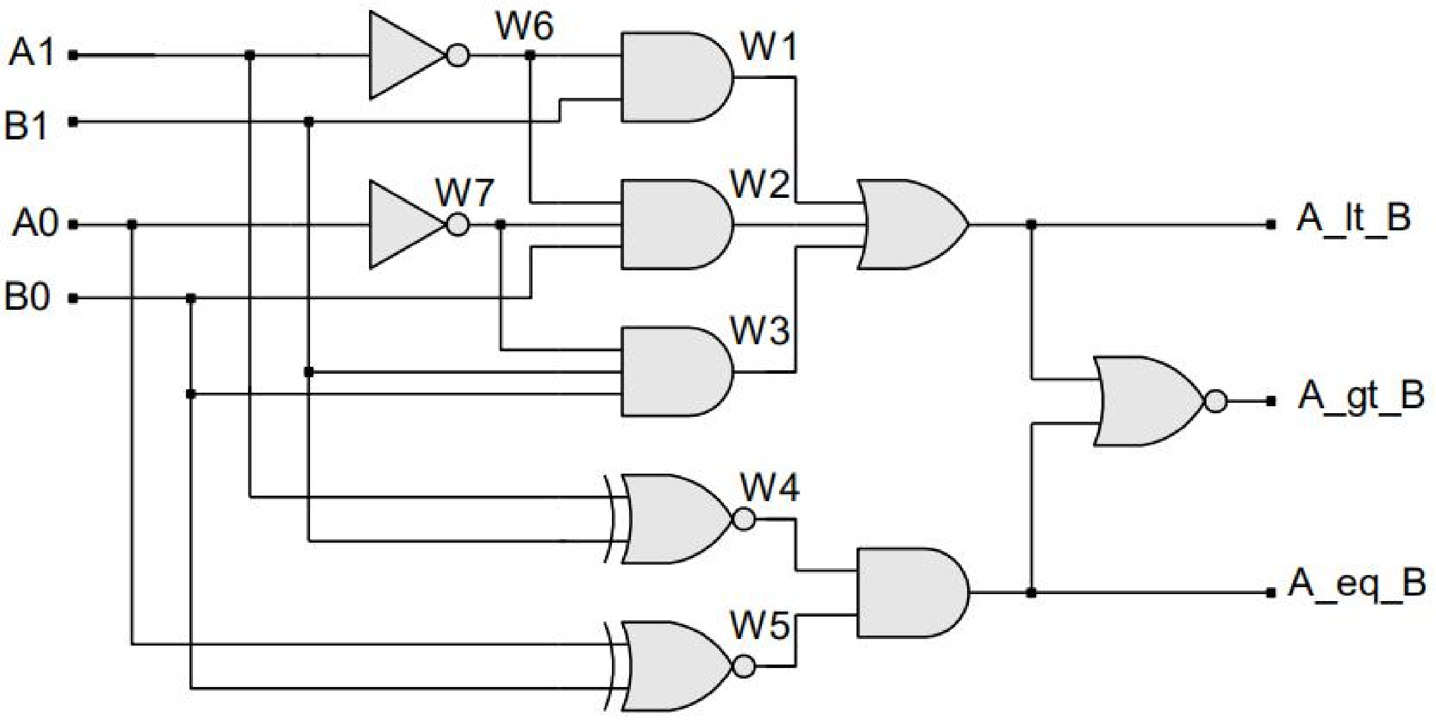
x	y	F
0	0	0
0	1	1
1	0	1
1	1	0

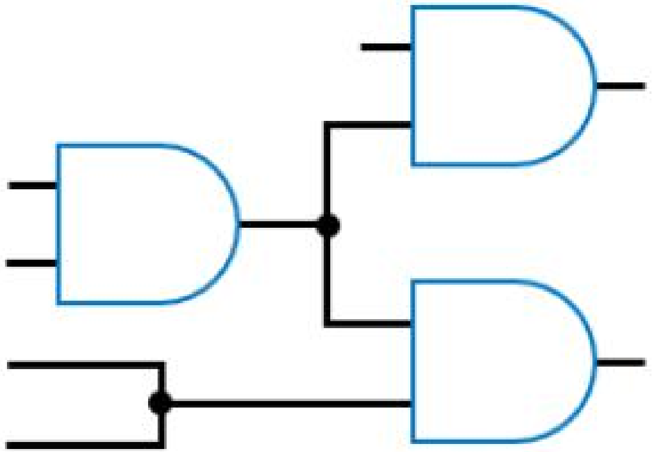
x	y	F
0	0	1
0	1	0
1	0	0
1	1	1

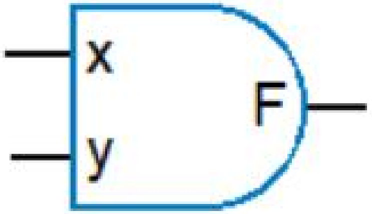
- NAND: Opposite of AND (“NOT AND”)
- NOR: Opposite of OR (“NOT OR”)
- XOR: outputs 1 when inputs have odd number of 1’s
- XNOR: Opposite of XOR (“NOT XOR”)

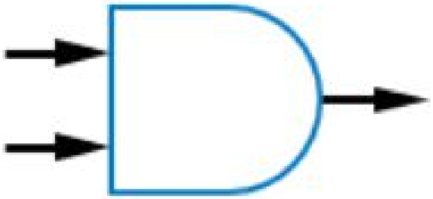
Hw3

10. Highlight the critical paths of the following circuit. Assume that each gate (including the individual inverters and XOR gates) has a delay of 1 ns and each wire has a delay of 0.5 ns. (5 points)






x
y
F



K-map

- Building from truth table
- Grouping
- Find PIs and EPIs
- Practice makes perfect!!!

Building

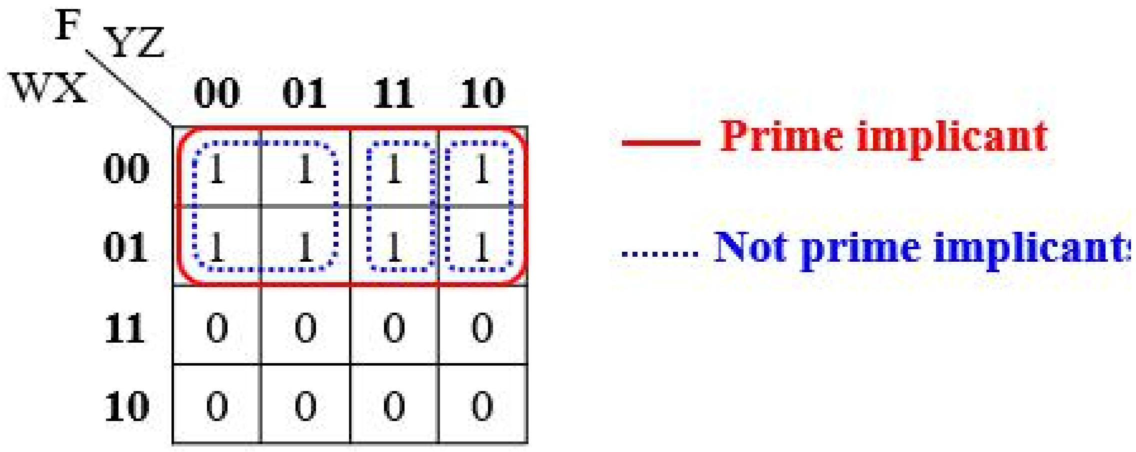
- From truth table
- From current equation (first transfer to truth table)

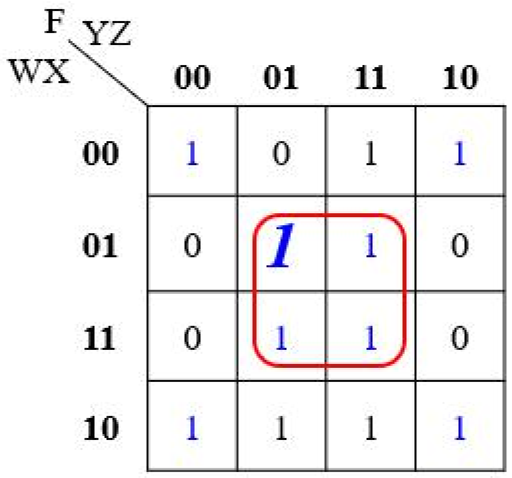
W	X	Y	Z	F		
0	0	0	0	1	m0	W'X'Y'Z'
0	0	0	1	0	m1	W'X'Y'Z
0	0	1	0	1	m2	W'X'YZ'
0	0	1	1	1	m3	W'X'YZ
0	1	0	0	0	m4	W'XY'Z'
0	1	0	1	0	m5	W'XY'Z
0	1	1	0	0	m6	W'XYZ'
0	1	1	1	1	m7	W'XYZ
1	0	0	0	1	M8	WX'Y'Z'
1	0	0	1	1	m9	WX'Y'Z
1	0	1	0	0	M10	WX'YZ'
1	0	1	1	0	m11	WX'YZ
1	1	0	0	1	m12	WXY'Z'
1	1	0	1	0	m13	WXY'Z
1	1	1	0	0	m14	WXYZ'
1	1	1	1	1	m15	WXYZ

WX \ YZ	00 Y'Z'	01 Y'Z	11 YZ	10 YZ'
W'X'	1 m0	0 m1	1 m3	1 m2
W'X	0 m4	0 m5	1 m7	0 m6
WX	1 m12	0 m13	1 m15	0 m14
WX'	1 m8	1 m9	0 m11	0 m10

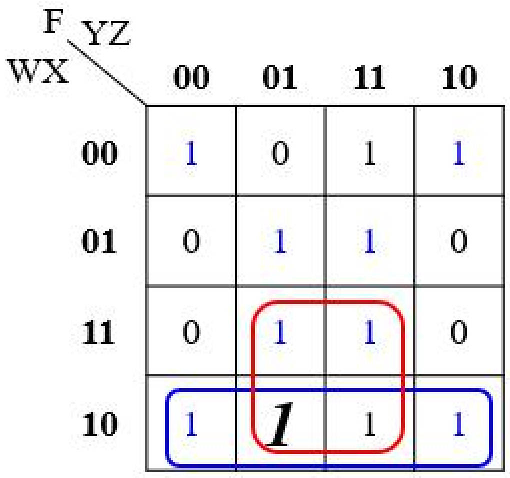
PI and EPI

- A prime implicant (PI) is a group that cannot be entirely contained by another implicant
- A prime implicant (PI) is essential if one of its cells is covered ONLY by that PI





Essential PI: XZ



No essential PIs found