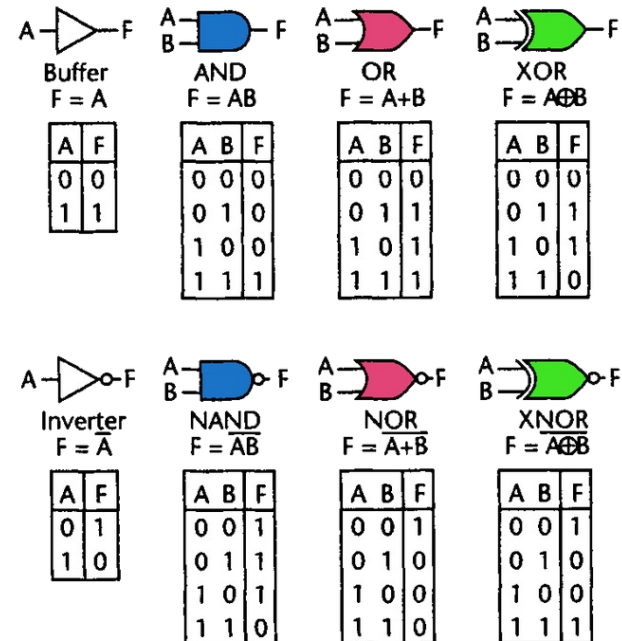# EE2000 Logic Circuit Design
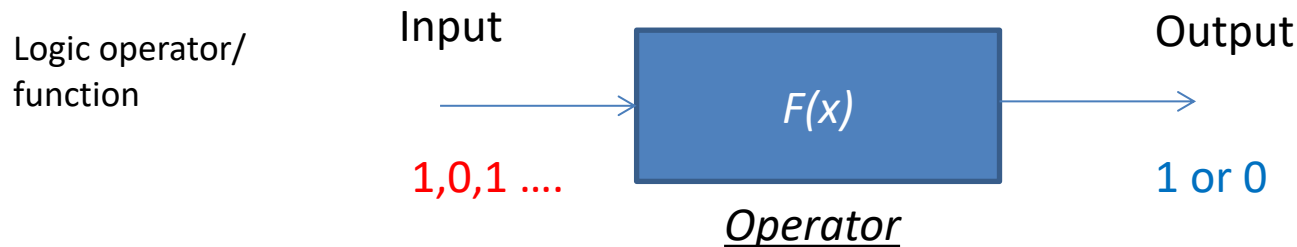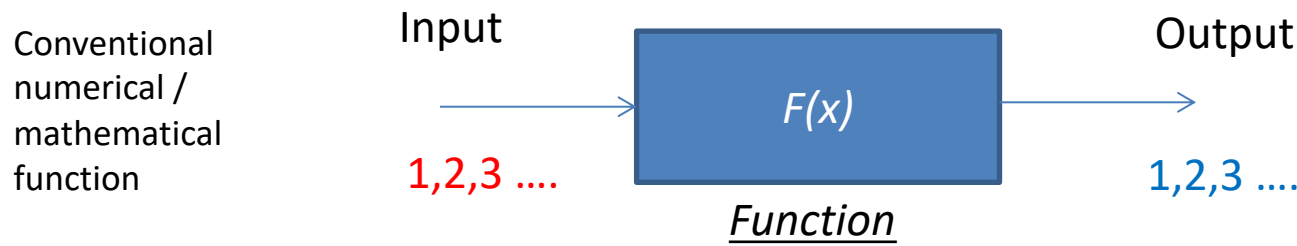
Lecture 1 – Logic Function and Boolean Algebra

Outline

1.1   Basic Logic Gates

1.2   Logic Circuit and Boolean Expression

1.3   Sum of Products vs Product of Sums and Canonical Form

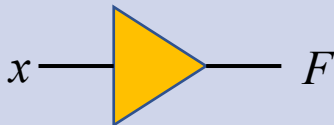1.4   Simplification using Boolean Algebra

1.1 Basic Logic Gates

Gate describes a circuit that performs a basic logic operation

Conventional numerical / mathematical function

Input

Output

F(x)

1,2,3 ....

1,2,3 ....

Function

Logic operator/ function
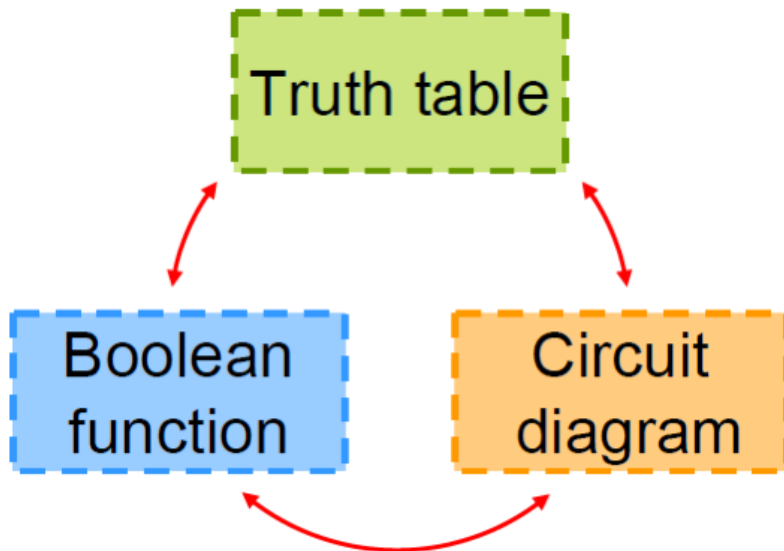
Input

Output

F(x)

1,0,1 ....

1 or 0

Operator

> ➢ Basic building blocks
> ➢ Perform certain function or logic
> ➢ One or more inputs
> ➢ One output
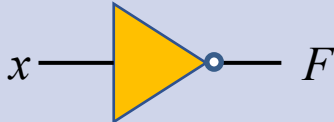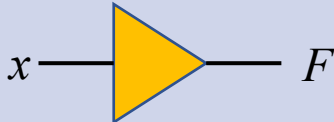
**Binary decision output
e.g, Yes/No ; True/False and 1/0**

Logic Gate

Logic Name	Circuit Diagram	Truth Table	Boolean Function
Buffer: Output (F) follows the same logic state as the Input (x)	x —▷— F	<table><tr><td>x</td><td>F</td></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td></tr></table>	$F(x) = x$ $F = x$

$$F(x, y) = x \cdot y + \overline{x \cdot y}$$

Truth table

Boolean function

Circuit diagram

> Function (F): Operation

> Variable: Inputs $(x, \ y)$

> Complement: Inversion ($\bar{x}, \ \bar{y}$)

> Literal: Each appearance of a variable or its complement (4)

> Product term: One or more literals connected by $\cdot$ operator (2)
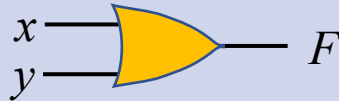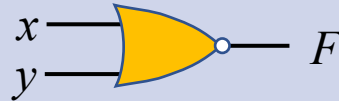
NOT Gate

Logic Name	Circuit Diagram	Truth Table	Boolean Function
NOT gate (Inverter): Output (F) has opposite logic state of the Input (x)	x —▷o— F	x | F 0 | 1 1 | 0	$F = \bar{x}$ $F = x'$

Logic Name	Circuit Diagram	Truth Table	Boolean Function
Buffer: Output (F) follows the same logic state as the Input (x)	x —▷— F	x | F 0 | 0 1 | 1	$F(x) = x$ $F = x$
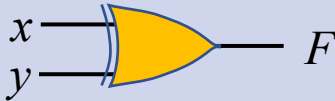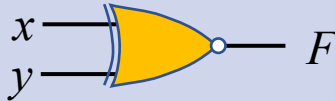
AND & NAND Gates

Logic Name	Circuit Diagram	Truth Table	Boolean Function
AND gate: Output (F) is 1 only when all inputs are 1	x —— y —— AND —— F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	$F = x \cdot y$ $F = xy$
NAND gate: Output (F) is 0 only when all inputs are 1	x —— y —— NAND —— F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	$F = \overline{x \cdot y}$ $F = \overline{xy}$

OR & NOR Gates

Logic Name	Circuit Diagram	Truth Table	Boolean Function
OR gate: Output (F) is 1 when either or all inputs are 1	x —— y —— F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	$F = x + y$
NOR gate: Output (F) is 1 only when all inputs are 0	x —— y —— F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	$F = \overline{x + y}$

XOR & XNOR Gates

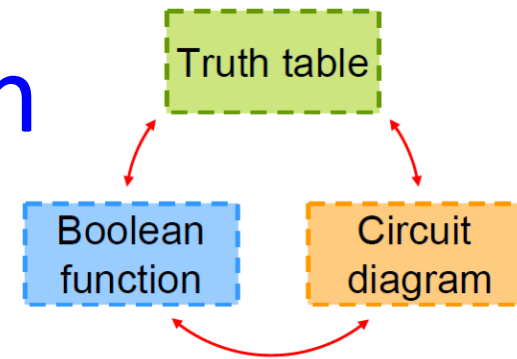| Logic Name	Circuit Diagram	Truth Table	Boolean Function
XOR gate: Output (F) is 1 only when one of the input is 1	x, y → F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	$F = x \oplus y$
XNOR gate: Output (F) is 1 only when all inputs are 0 or 1	x, y → F	<table><tr><td>x</td><td>y</td><td>F</td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	$F = \overline{x \oplus y}$ $F = x \otimes y$

1.2 Logic Circuit and Boolean Expression

Boolean function

Circuit diagram

To derive the Boolean expression of a given logic circuit, begin at the left-most inputs and work towards the final output, writing the expression for each gate.

1st Level

2nd Level

3rd Level
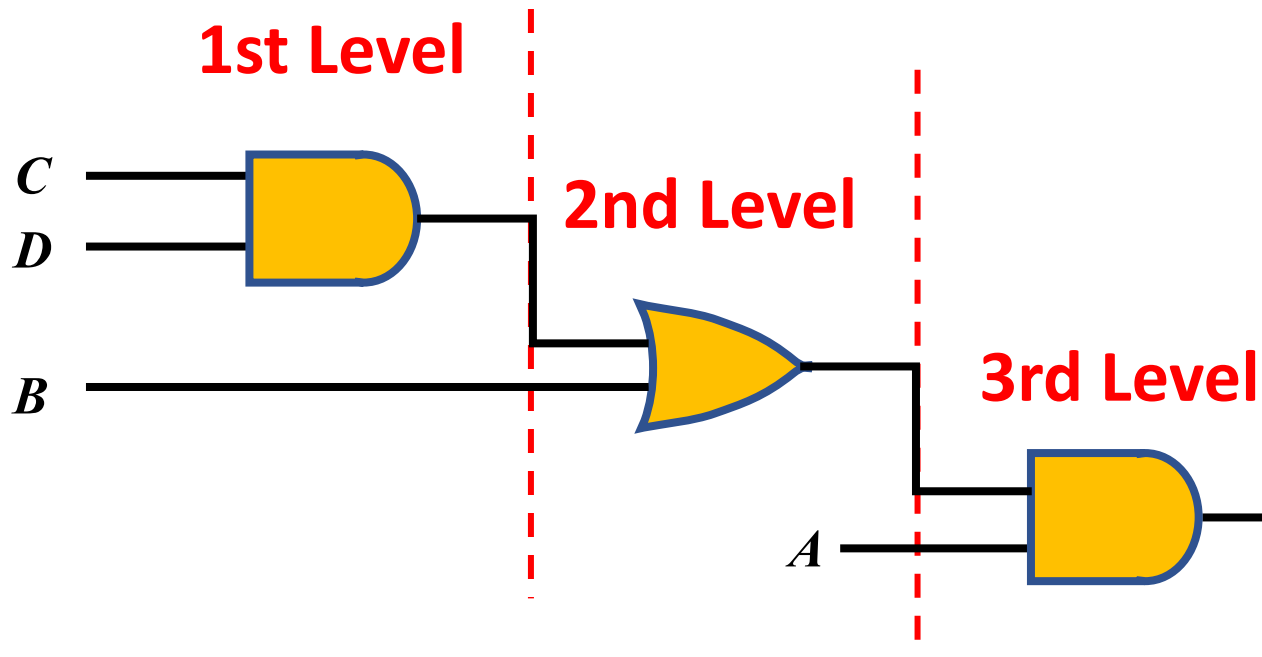
Boolean Expression
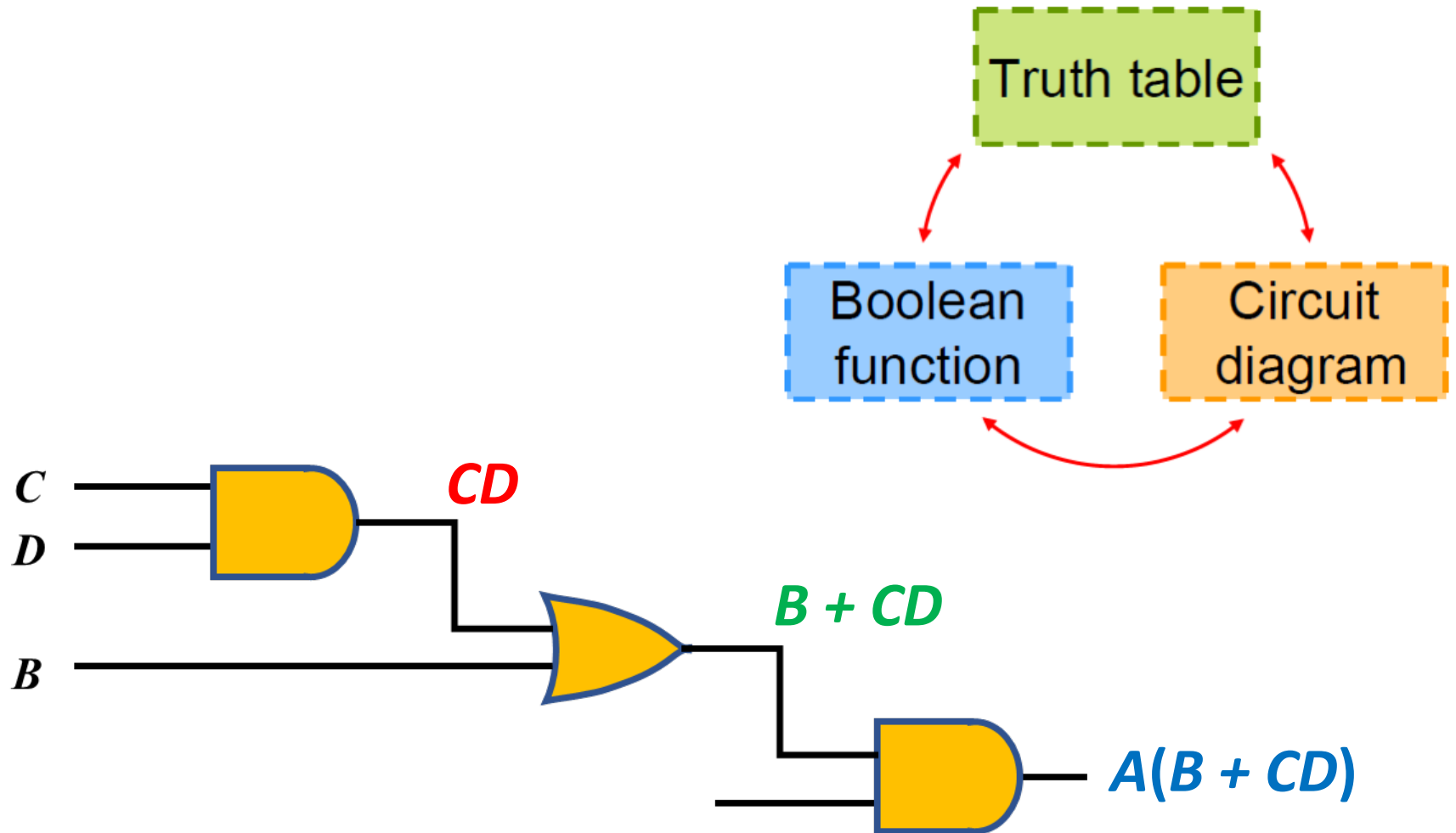
$$F = A(B + CD)$$

NEXT: Construct a truth table for above logic circuit

Truth Table
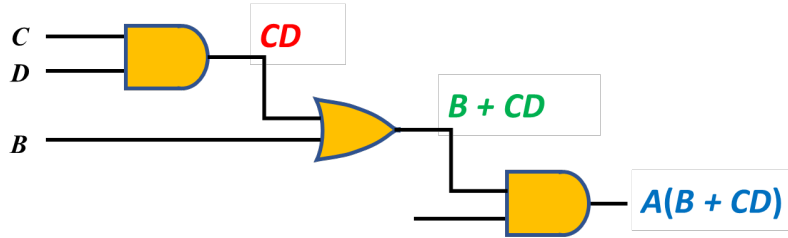
$$F = A(B + CD)$$

For the truth table, find the output following the steps:
1. Write down all input possibility
2. Write down the stage output (i.e. *CD*, *B + CD*)
3. Write down the final stage output (i.e. *A(B + CD)*)

Inputs				Operation		Output
A	B	C	D	CD	$B+CD$	$A(B+CD)$
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			

Complete Solution
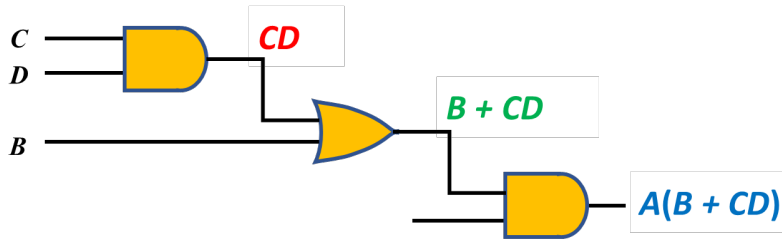
$$F = A(B + CD)$$

Examples of numbering systems		Inputs				Output
Decimal	Hexadecimal	A	B	C	D	A(B+CD)
0	0	0	0	0	0	0
1	1	0	0	0	1	0
2	2	0	0	1	0	0
3	3	0	0	1	1	0
4	4	0	1	0	0	0
5	5	0	1	0	1	0
6	6	0	1	1	0	0
7	7	0	1	1	1	0
8	8	1	0	0	0	0
9	9	1	0	0	1	0
10	A	1	0	1	0	0
11	B	1	0	1	1	1
12	C	1	1	0	0	1
13	D	1	1	0	1	1
14	E	1	1	1	0	1
15	F	1	1	1	1	1

From Truth Table

Inputs			Output
x	y	z	$f(x, y, z)$
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	0
1	1	1	0

$f(x, y, z) = 1$
if $(x = 0$ AND $y = 0$ AND $z = 1)$
OR
$(x = 1$ AND $y = 0$ AND $z = 0)$
OR
$(x = 1$ AND $y = 0$ AND $z = 1)$

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

Questions:
1. How many literals?
2. How many product terms?

12
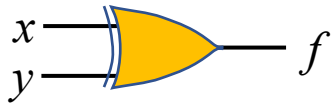
Exercises

x	y	f
0	0	0
0	1	1
1	0	1
1	1	0
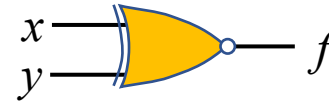
x	y	f
0	0	1
0	1	0
1	0	0
1	1	1

Given the Truth Tables of XOR and XNOR gates, write down the Boolean Function (using sum and product) and draw the Logic Circuit using NOT, AND and OR gates.

Exercises

Given the Boolean function $f(x, y, z) = xy' + x'z'$, draw the Logic Circuit and work out the truth table.

Inputs			xy'	$x'z'$	Output
x	y	z			$f(x, y, z)$
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

1.3 Sum of Products vs Product of Sums, Canonical Form

> Product terms: One or more literals connected by AND operator.

$$x, xy', xyz', y'z'$$

> Standard product terms: A product term that includes each variable of the problem, either uncomplemented or complemented.

$$xyz, xy'z', x'yz', x'y'z'$$

product terms

sum term
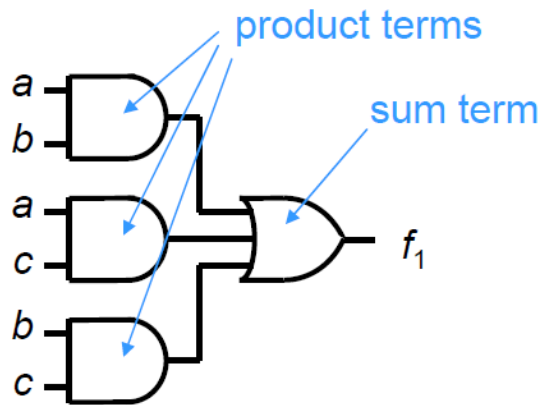
$f_1(a, b, c) = ab + ac + bc$

> Sum of Products (SOP): A group of AND gates followed by a single OR gate.

$xy + yz + x'z + z'$	4 product terms
$xy + yz + x'z$	3 product terms
$xy + yz$	2 product terms
xy	1 product term

Canonical (Standard) Form in SOP

Inputs			Minterms		Output
x	y	z	Term	Designation	$f(x, y, z)$
0	0	0	$x'y'z'$	m_0	1
0	0	1	$x'y'z$	m_1	0
0	1	0	$x'yz'$	m_2	1
0	1	1	$x'yz$	m_3	0
1	0	0	$xy'z'$	m_4	1
1	0	1	$xy'z$	m_5	1
1	1	0	xyz'	m_6	0
1	1	1	xyz	m_7	0

➢ Minterms: Standard product terms. Uncomplement = 1; Complement = 0.

➢ Canonical Sum (Sum of standard product terms): Sum of products expression with minterms only when output is 1

$$f(x, y, z) = x'y'z' + x'yz'$$
$$+ xy'z' + xy'z$$
$$= m_0 + m_2 + m_4 + m_5$$
$$= \sum m(0, 2, 4, 5)$$

Question

Which function(s) is formed by minterm?

A) $f = a + b + c$

B) $f = ab + bc + ac$

C) $f = abc + ab'c'$

D) $f = ab + ab'c'$

DeMorgan's Theorem

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

Complement of Sum = Product of Complement

x	y	$\overline{x + y}$	$\bar{x} \cdot \bar{y}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

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

Complement of Product = Sum of Complement
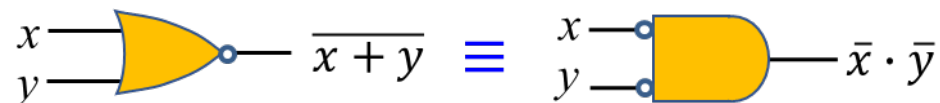
x	y	$\overline{xy}$	$\bar{x} + \bar{y}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

Common mistake! $(xy)' \neq x'y'$ **and** $(x + y)' \neq x' + y'$

Example

$$(a + b)' = a'b'$$

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

$$f = wx'y + xy' + wxz$$

$$f' = (wx'y + xy' + wxz)'$$
$$= (wx'y)'(xy')'(wxz)'$$
$$= (w' + x + y')(x' + y)(w' + x' + z')$$

Product of Sums

> Sum terms: One or more literals connected by OR operator.

$$x, x + y', y + z', x + y + z'$$

> Standard sum terms: A sum term that includes each variable of the problem, either uncomplemented or complemented.

$$x + y + z, x + y' + z', x' + y + z', x' + y' + z'$$

sum terms

product term

f_2
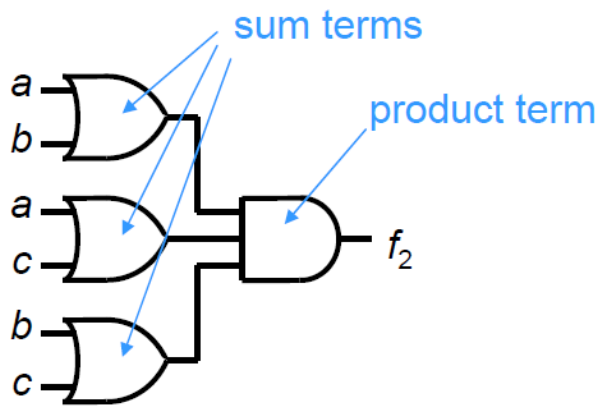
$f_2(a, b, c) = (a + b)(a + c)(b + c)$

> Product of Sums (POS): A group of OR gates followed by a single AND gate.

$(w + z)(w' + y')xy$	4 sum terms
$(w + z)(w' + y')x$	3 sum terms
$(w + z)(w' + y')$	2 sum terms
$(w + z)$	1 sum term

SOP to POS

$$f(x, y, z)$$
$$= x'y'z' + x'y'z + x'yz$$
$$+ xy'z + xyz'$$

$$= m_0 + m_1 + m_3 + m_5$$
$$+ m_6$$

$$= \sum m(0,1,3,5,6)$$

Inputs			Minterms		Output	
x	y	z	Term	Designation	f	f'
0	0	0	$x'y'z'$	m_0	1	0
0	0	1	$x'y'z$	m_1	1	0
0	1	0	$x'yz'$	m_2	0	1
0	1	1	$x'yz$	m_3	1	0
1	0	0	$xy'z'$	m_4	0	1
1	0	1	$xy'z$	m_5	1	0
1	1	0	xyz'	m_6	1	0
1	1	1	xyz	m_7	0	1

$$f'(x, y, z) = x'yz' + xy'z'$$
$$+xyz$$

$$f(x, y, z)$$
$$= (x'yz' + xy'z' + xyz)'$$
$$= (x'yz')'(xy'z')'(xyz)'$$
$$= (x + y' + z)(x' + y + z)$$
$$(x' + y' + z')$$

$$(a + b)' = a'b' \qquad (ab)' = a' + b'$$

Canonical (Standard) Form in POS

Inputs			Maxterms		Output
x	y	z	*Term*	*Designation*	$f(x, y, z)$
0	0	0	$x + y + z$	M_0	1
0	0	1	$x + y + z'$	M_1	1
0	1	0	$x + y' + z$	M_2	0
0	1	1	$x + y' + z'$	M_3	1
1	0	0	$x' + y + z$	M_4	0
1	0	1	$x' + y + z'$	M_5	1
1	1	0	$x' + y' + z$	M_6	1
1	1	1	$x' + y' + z'$	M_7	0

➢ Maxterms: Standard sum terms. Uncomplement = 0; Complement = 1.

➢ Canonical Product (Product of standard sum terms): Product of sums expression with maxterms only when output is 0

$$f(x, y, z)$$
$$= (x + y' + z)(x' + y + z)$$
$$\quad (x' + y' + z')$$
$$= M_2 M_4 M_7 = \prod M(2,4,7)$$

Thought (SOP or POS?)

Inputs			Output
x	y	z	$f(x, y, z)$
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	0

SOP & POS

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

$$f(x,y,z) = \sum m(3,5) = \prod M(0,1,2,4,6,7)$$

$$f'(x,y,z) = \prod M(3,5) = \sum m(0,1,2,4,6,7)$$

Summary

➤ $\overline{m_i} = M_i$ and $\overline{M_i} = m_i$

$\overline{m_0} = (x'y'z')' = x + y + z = M_0$

➤ If a f is in SOP form, its complement is in POS form (*vice versa*).

$f = xyz + xy'z$

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

➤ Canonical SOP (all minterms with output 1)

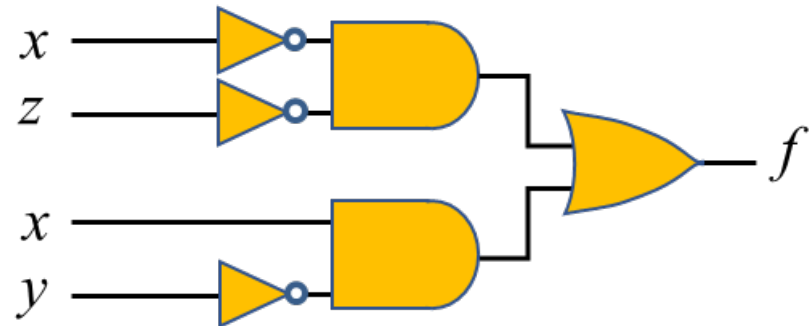$f = xyz + xy'z = \sum m(5,7)$ and $f' = \sum m(0,1,2,3,4,6)$

➤ Canonical POS (all maxterm with output 0)

$f = \prod M(0,1,2,3,4,6)$ and $f' = \prod M(5,7)$

Recap

Given the Boolean function $f(x, y, z) = xy' + x'z'$, draw the Logic Circuit and work out the truth table.

Inputs			xy'	$x'z'$	Output
x	y	z			$f(x, y, z)$
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	1	0	1
1	0	1	1	0	1
1	1	0	0	0	0
1	1	1	0	0	0

$$f(x, y, z) = x'y'z' + x'yz'$$
$$+xy'z' + xy'z$$

26

1.4 Simplification using Boolean Algebra

- Obtain a simple (or the simplest) logic circuit

- Reduce the cost of circuit

    Cost in logic circuit

    - Gate cost (number of gates in the implementation)

    - Gate-input cost (number of inputs to the gates)

    - Total cost = Gate cost + Gate-input cost

Gate-input cost

- The number of gate-input is proportional to the number of transistors in the logic circuit
- The cost can be determined by checking logic diagram / schematic and the Boolean function
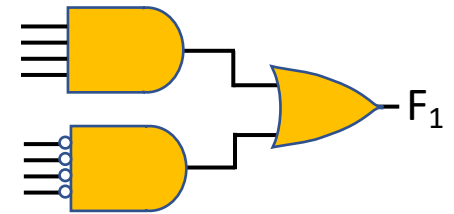
$$F_1 = abcd + a'b'c'd'$$

$$F_2 = (a' + b)(b' + c)(c' + d)(d' + a)$$

F_1 has **3** gates and **10** gate-inputs

Total cost = 13

F_2 has **5** gates and **12** gate-inputs

Total cost = 17
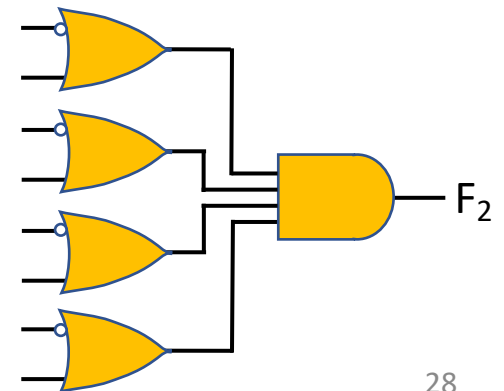
How to reduce cost?

The key of simplifying logic functions is to reduce the no. of terms and no. of literals

$$\downarrow \text{no. of literals} = \downarrow \text{no. of gate inputs}$$
$$\downarrow \text{no. of terms} = \downarrow \text{no. of gates}$$

In addition, costs in space and power consumption can be reduced

Simplification

$f(a, b, c) = a'bc' + a'bc + ab'c' + ab'c + abc$

5 product terms, 15 literals
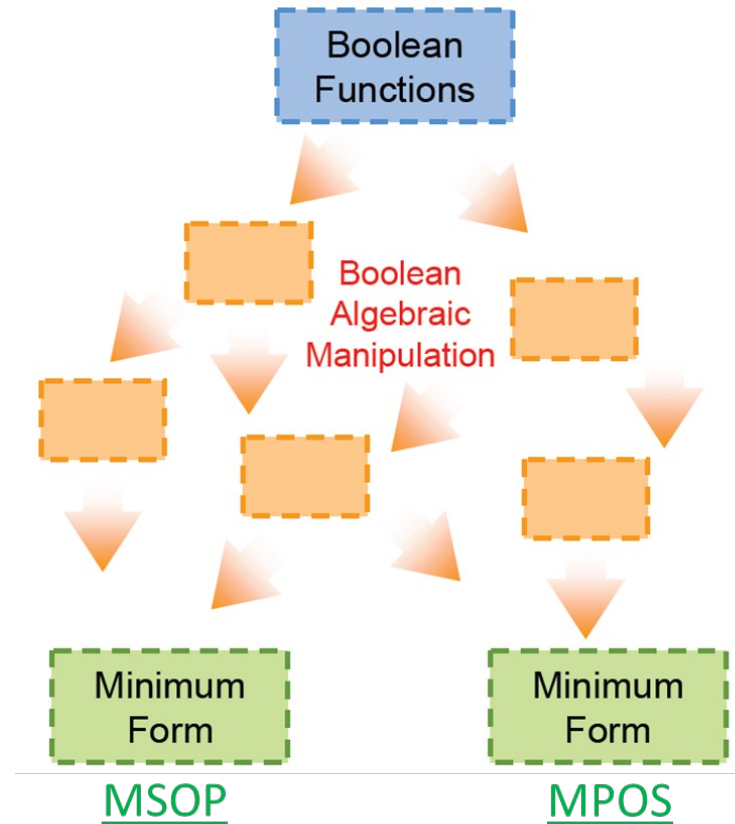
$f(a, b, c) = a'b + ab' + abc$

3 product terms, 7 literals

$f(a, b, c) = a'b + ab' + ac$

3 product terms, 6 literals

$f(a, b, c) = a'b + ab' + bc$

3 product terms, 6 literals

Boolean Functions

Boolean Algebraic Manipulation

Minimum Form — MSOP

Minimum Form — MPOS

Boolean or Switching Algebra

➢ A system of mathematical logic to perform different operations in binary system with elements of S = {0, 1}

➢ {0 ,1} represents a light off or on, a switch up or down, a low or high voltage, *etc.*

➢ The formulation is referred as Switching or Boolean Function

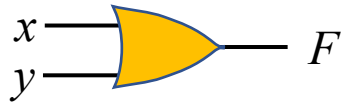*e.g.* If $x, y \in S$,

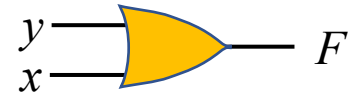$$f(x, y) = x + y$$

Basic Postulates

If $x, y \in S$,

Commutative	$x + y = y + x$	$xy = yx$
Associative	$x + (y + z) = (x + y) + z$	$x(yz) = (xy)z$
Identity	$x + 0 = x$	$x \cdot 1 = x$
Null	$x + 1 = 1$	$x \cdot 0 = 0$
Complement	$x + x' = 1$	$x \cdot x' = 0$
Idempotency	$x + x = x$	$x \cdot x = x$
Involution	$(x')' = x$	

Commutative

$$x + y = y + x$$

 is equivalent to
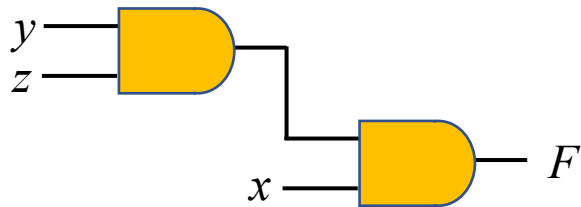
$$xy = yx$$

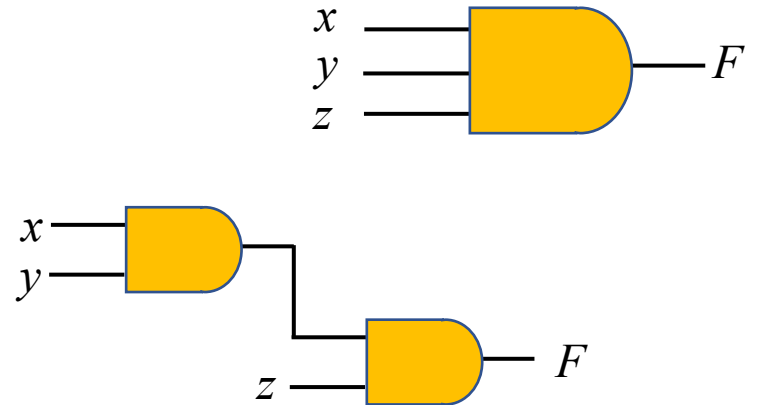 is equivalent to

Associative

$$x + (y + z) = (x + y) + z$$
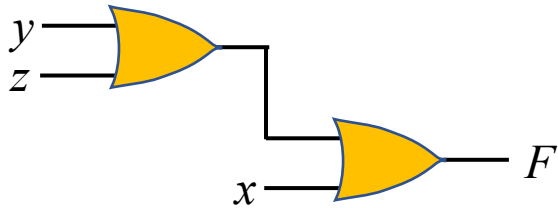
is equivalent to
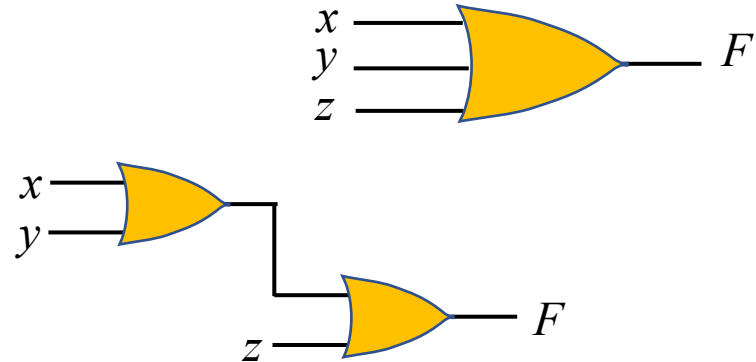
$$x(yz) = (xy)z$$

is equivalent to

Duality Principle

Commutative	$x + y = y + x$	$xy = yx$
Associative	$x + (y + z)$ $= (x + y) + z$	$x(yz) = (xy)z$
Identity	$x + 0 = x$	$x \cdot 1 = x$
Null	$x + 1 = 1$	$x \cdot 0 = 0$
Complement	$x + x' = 1$	$x \cdot x' = 0$
Idempotency	$x + x = x$	$x \cdot x = x$

Boolean algebra remains true when the operators OR and AND is interchanged and the identity elements 0 and 1 are interchanged.
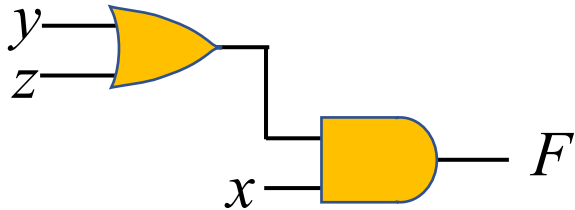
Distributive
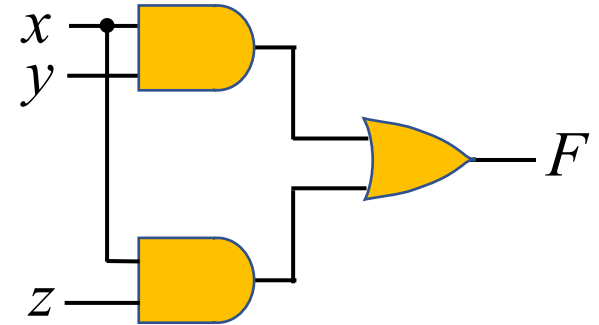
$$x(y + z) = xy + xz$$

$$x + yz = (x + y)(x + z)$$

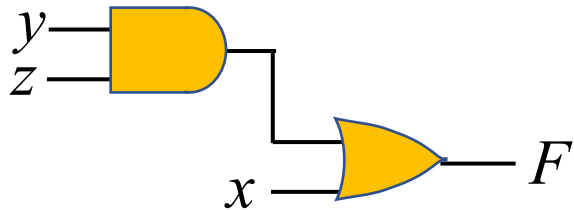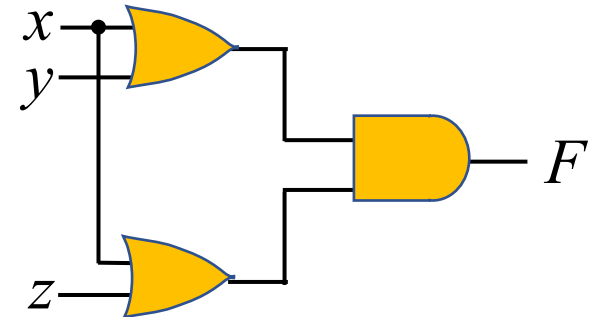| Inputs			$x + yz$			$(x + y)(x + z)$		
x	y	z	x	yz	f	$x + y$	$x + z$	f
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0
0	1	0	0	0	0	1	0	0
0	1	1	0	1	1	1	1	1
1	0	0	1	0	1	1	1	1
1	0	1	1	0	1	1	1	1
1	1	0	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1

Distributive

$$x(y + z) = xy + xz$$

is equivalent to

$$x + yz = (x + y)(x + z)$$

is equivalent to

Question

- Which of the following has the same function as $x + x'y$?
    a) $x + xy'$
    b) $x + y$
    c) $x' + y$
    d) y

Simplification

$$x + x'y = (x + x')(x + y) \quad \text{distributive}$$

$$= 1 \cdot (x + y) \quad \text{complement}$$

$$= x + y \quad \text{identity}$$

$$x(x' + y) = xx' + xy \quad \text{distributive}$$

$$= 0 + xy \quad \text{complement}$$

$$= xy \quad \text{identity}$$

Adjacency

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

$$= x \cdot 1$$ complement

$$= x$$ identity

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

$$= xx + xy + xy' + yy'$$

$$= x + x(y + y') + 0$$ complement

$$= x + x(1) = x$$ Identity and idempotency

Exercise

- Simplify the following expression.

$$xyz' + xyz + xy'z + x'yz + x'y'z + x'y'z'$$

Exercise

- Simplify the following expression.

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

Adsorption

$$x + xy = x \cdot 1 + xy \qquad \text{identity}$$

$$= x(1 + y) \qquad \text{distributive}$$

$$= x \qquad \text{Null}$$

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

Exercise

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

$$B+AC$$

(a)

(b)
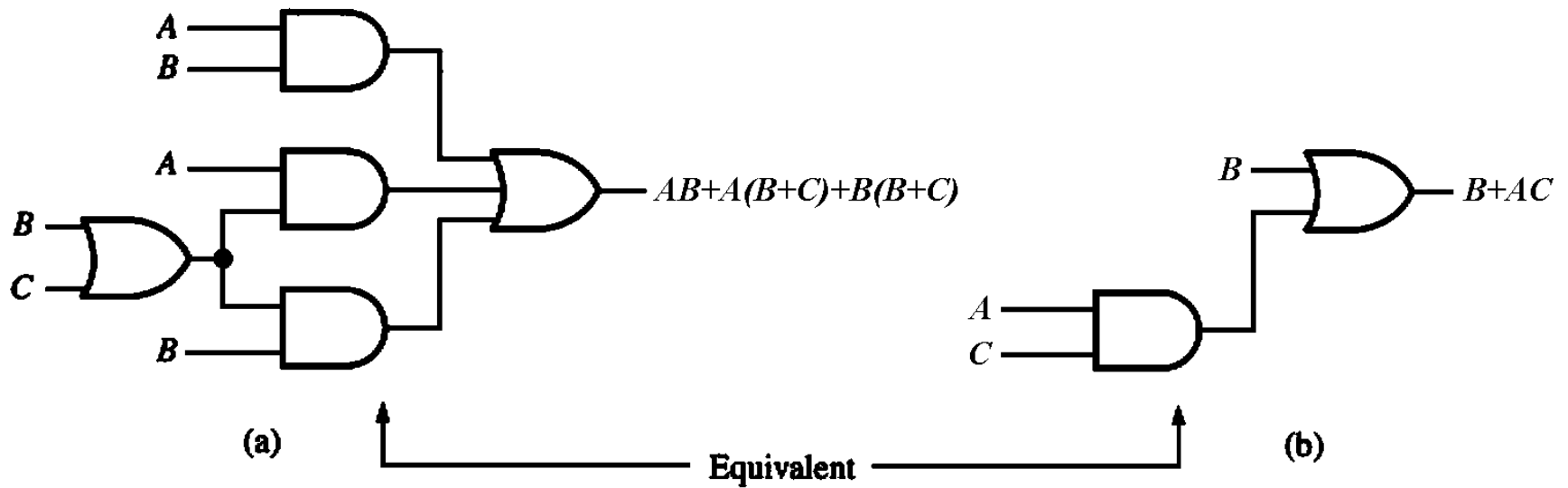
Equivalent

Prove that the above Circuit (a) is equivalent to Circuit (b).

Summary

Distributive	$x(y + z) = xy + xz$	$x + yz = (x + y)(x + z)$
Simplification	$x + x'y = x + y$	$x(x' + y) = xy$
Adjacency	$xy + xy' = x$	$(x + y)(x + y') = x$
Adsorption	$x + xy = x$	$x(x + y) = x$

Consensus

$$xy + x'z + {\color{red}yz} = xy + x'z + yz(x + x') \;{\color{blue}\text{complement}}$$
$$= xy + xyz + x'z + x'yz \quad {\color{blue}\text{distributive}}$$
$$= xy + x'z \quad {\color{blue}\text{adsorption}}$$

Consensus term: For any two product terms where **exactly** one variable appears uncomplemented in one and complemented in the other, the Consensus term is the product of the remaining literals. **The consensus term could be eliminated.**

Term 1	Term 2	Consensus Term
$x y' z$	$w x'$	$w y' z$
$w x y'$	$x y z'$	$w x z'$
$w x y'$	$x y' z$	$-$
$x y' z$	$w x' y$	$-$

46

Question

Which of the following can be simplified using consensus theorem?

a) $xy'z + wx' + wy'z' + wy$
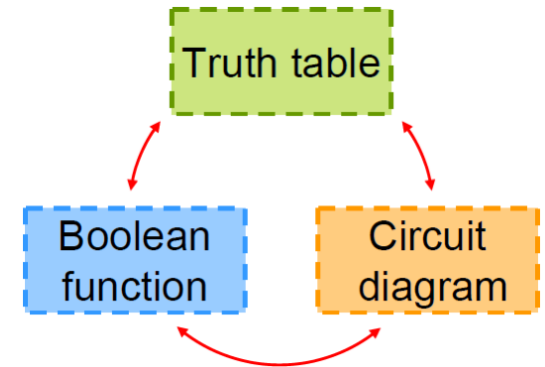
b) $wxy' + xyz' + w'xz' + yz$

c) $wx' + xy'z + w'yz' + x'yz'$

Exercise

Simplify the following function.

$$f(w, x, y, z) = wxy' + w'y'z + wx'y' + xy'z + w'z$$

Conclusion

1.1   Basic Logic Gates

 - Circuit diagram, Boolean function, Truth table

1.2   Logic Circuit and Boolean Expression

- Work out complete solution

1.3   Sum of Products vs Product of Sums and Canonical
        Form

- Express function in Canonical SOP or POS

1.4   Simplification using Boolean Algebra

- Simplify the function for simpler circuit diagram

Summary (Given in Test and Exam)

|		
Commutative	$a + b = b + a$	$ab = ba$
Associative	$a + (b + c) = (a + b) + c$	$a(bc) = (ab)c$
Identity	$a + 0 = a$	$a(1) = a$
Null	$a + 1 = 1$	$a(0) = 0$
Complement	$a + a' = 1$	$a(a') = 0$
Idempotency	$a + a = a$	$a(a) = a$
Involution	$(a')' = a$	
Distributive	$a(b + c) = ab + ac$	$a + bc = (a + b)(a + c)$
Adjacency	$ab + ab' = a$	$(a + b)(a + b') = a$
Simplification	$a + a'b = a + b$	$a(a' + b) = ab$
DeMorgan	$(a + b)' = a'b'$	$(ab)' = a' + b'$
Absorption	$a + ab = a$	$a(a + b) = a$
Consensus	$ab + a'c + bc = ab + a'c$	

Exercise

1. Derive the Boolean functions to describe the operations of the logic circuit as shown.
2. Simplify the functions and draw the circuit.
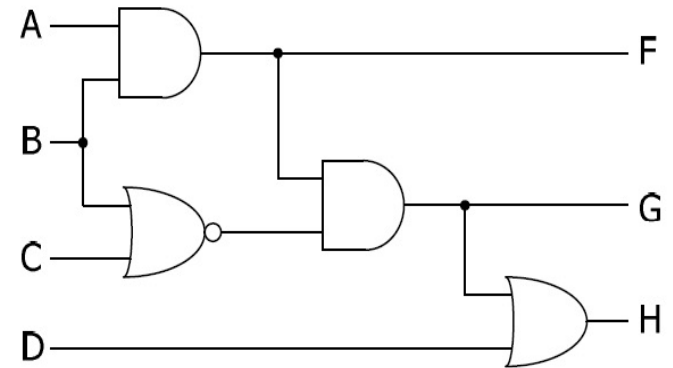
Exercise

Work out the Boolean functions of the following circuits. Which standard logic gate does each of them represent?
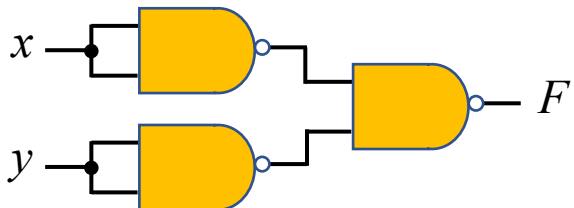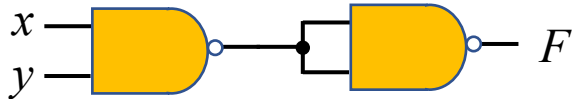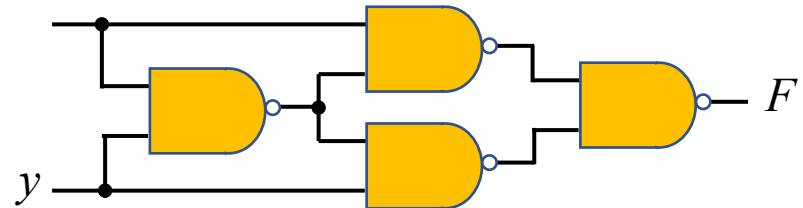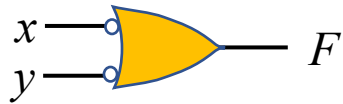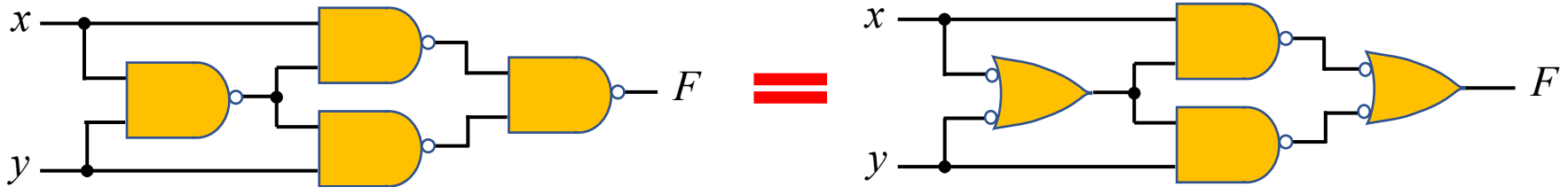
Exercise

Work out the Boolean functions of the following circuits. Which standard logic gate does each of them represent?

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

$$F = x(x' + y') + y(x' + y')$$
$$= xy' + yx' = x \oplus y$$

> ➤ Any Boolean function can be implemented using NAND gates (Functional completeness)
>
> ➤ NAND gates are used for SOP function
>
> ➤ Same for NOR gate but for POS function

Exercise

1. Express the Canonical Sum and Product based on the Truth Table provided.
2. Simplify the Function in SOP form.
3. Design the logic circuit using
   NAND and NOT gates.

Inputs			Output
x	y	z	$f(x, y, z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Next Lecture

Simplify $f(a, b, c, d) = \Sigma m(0, 1, 2, 5, 8, 9, 10)$ in POS form

Fill the 1s and 0s into the map

Group the 0s using the same procedure as grouping the 1s

$f'(a, b, c, d) = ab + cd + bd'$

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