

ENGG2020 DIGITAL LOGIC AND SYSTEMS

CHAPTER 2: BOOLEAN ALGEBRA AND LOGIC GATES

By Dr. Anthony Sum

Department of Computer Science and Engineering

The Chinese University of Hong Kong

1

CONTENTS

- Truth Table
- Common Logic Gates
- Boolean expressions
- Boolean algebra
- Alternative Logic Gate Representations
- Physical Characteristics of Logic IC

2

DIGITAL LOGIC

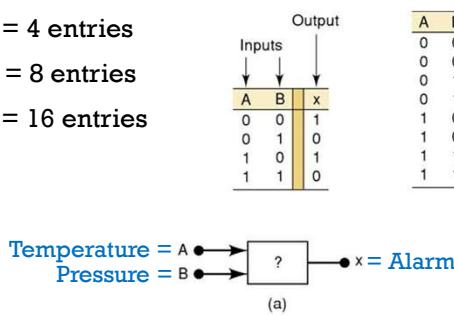
- Digital logic allows only two values, 0 and 1
 - Logic 0 can be false, off, low, no, open switch
 - Logic 1 can be true, on, high, yes, closed switch

Logic 0	Logic 1
False	True
Off	On
LOW	HIGH
No	Yes
Open switch	Closed switch

3

TRUTH TABLE

- A truth table describes the **relationship** between the **input** and **output** of a logic circuit
 - (a) A 2-input table has $2^2 = 4$ entries
 - (b) A 3-input table has $2^3 = 8$ entries
 - (c) A 4-input table has $2^4 = 16$ entries



A	B	C	x
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

(b)

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

(c)

4

COMMON LOGIC GATES

- Three common logical operations:
 - AND
 - OR
 - NOT (i.e. inverter)
- They form the basic elements of digital electronic circuits as switches, also known as **logic gates**

5

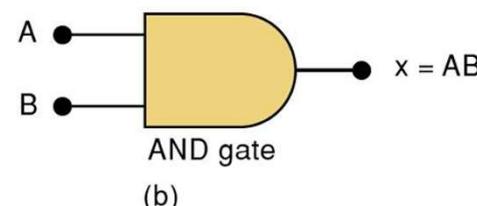
AND GATE

- AND logical operation is similar to **multiplication**, $X = A \cdot B$
- (a) The truth table, and (b) the symbol of AND

AND

A	B	$x = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

(a)

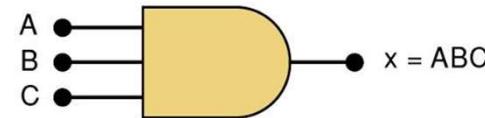


6

AND GATE

- Three inputs AND gate

A	B	C	x = ABC
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



7

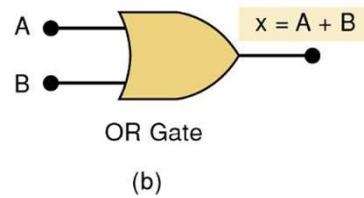
OR GATE

- OR logical operation is similar to **summation**, $X = A + B$
- (a) The truth table, and (b) the symbol of OR

OR

A	B	x = A + B
0	0	0
0	1	1
1	0	1
1	1	1

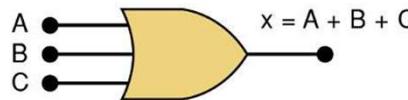
(a)



8

OR GATE

- Three inputs OR gate



A	B	C	x = A + B + C
0	0	0	0
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	1

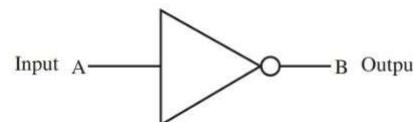
9

NOT GATE

- NOT logical operation is the **inverse/complement** of the input
- It is commonly written as either $X = \bar{A}$ or $X = A'$
- The truth table, and the symbol of NOT

NOT

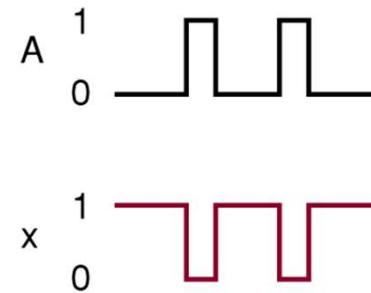
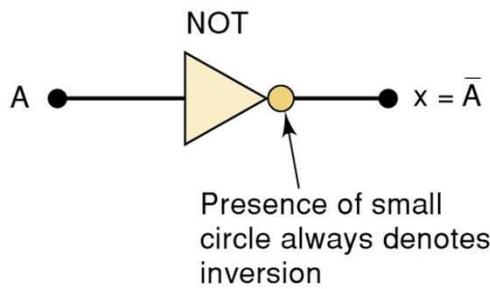
A	x = \bar{A}
0	1
1	0



10

NOT GATE

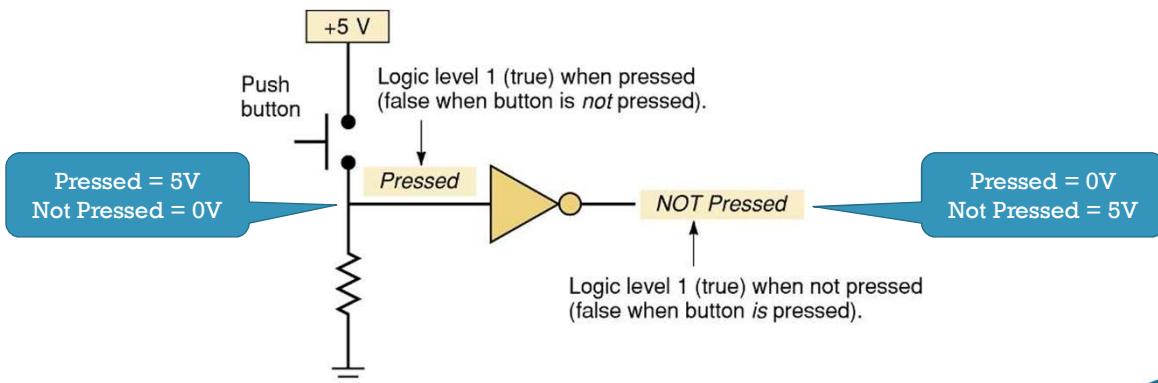
- A NOT gate is also called an inverter



11

NOT GATE

- Application of a NOT gate



12

SUMMARY OF COMMON LOGIC GATES

OR

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 1$$

AND

$$0 \cdot 0 = 0$$

$$0 \cdot 1 = 0$$

$$1 \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

NOT

$$\bar{0} = 1$$

$$\bar{1} = 0$$

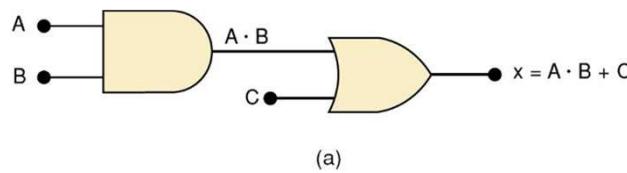
13

14

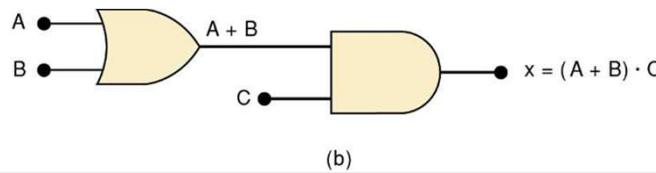
BOOLEAN EXPRESSIONS

LOGIC CIRCUITS AND BOOLEAN EXPRESSION

- If an expression contains both AND and OR gates, the **AND operation will be performed first**



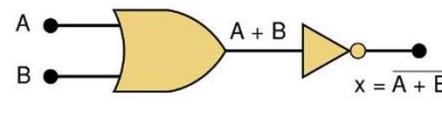
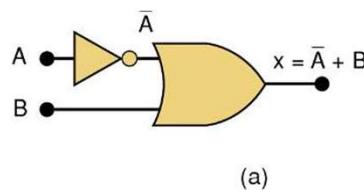
- Unless there is a parenthesis in the expression



15

LOGIC CIRCUITS AND BOOLEAN EXPRESSION

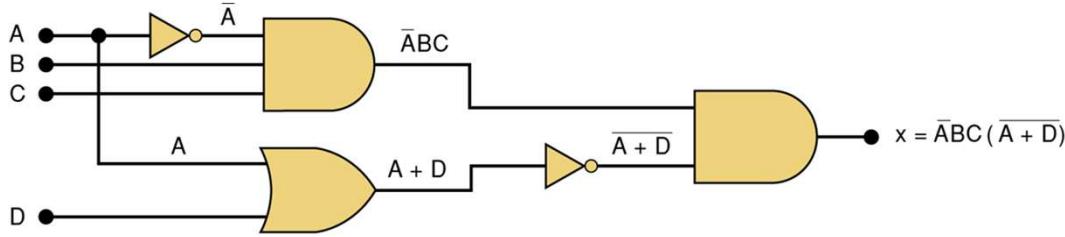
- Logic circuits with NOT gate, or inverter



16

LOGIC CIRCUITS AND BOOLEAN EXPRESSION

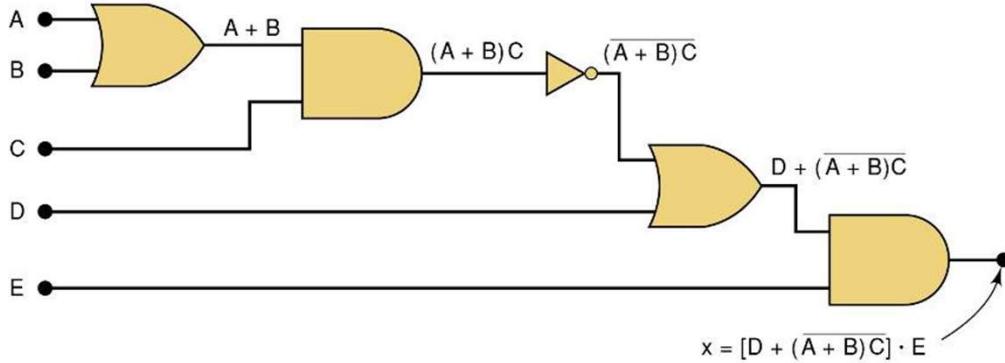
- $X = A'BC (A+D)'$



17

LOGIC CIRCUITS AND BOOLEAN EXPRESSION

- $X = (D + ((A+B)C)')E$



18

RULES ON BOOLEAN EXPRESSION EVALUATION

- As in the previous examples, logic circuits can be expressed by Boolean expressions
- Rules of evaluating a Boolean expression:
 - Perform all inversions of single terms
 - Perform all operations with parenthesis
 - Perform AND operation before an OR operation unless parenthesis indicate otherwise
 - If an expression has a bar over it, perform operations inside the expression, and then invert the result

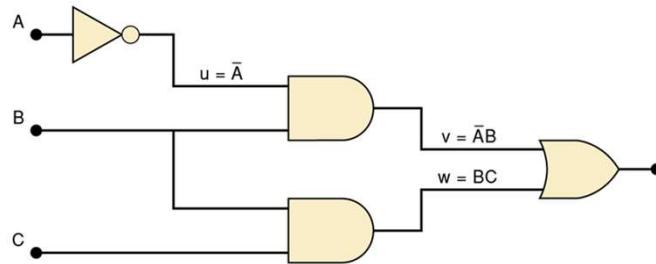
19

20

TRUTH TABLE & LOGIC CIRCUIT & BOOLEAN EXPRESSION

TRUTH TABLE OF LOGIC CIRCUIT

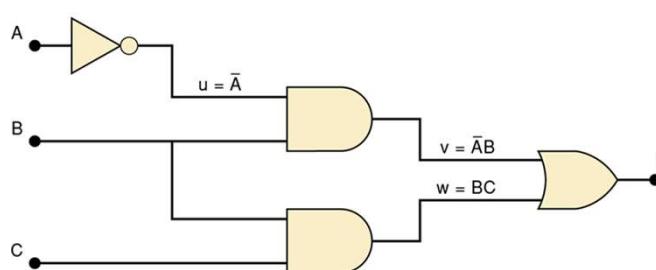
- One of the best way to analyze a logic circuit is to use a truth table
- Let's generate a truth table for the following logic circuit



21

TRUTH TABLE OF LOGIC CIRCUIT

- Step 1: For a **N**-input logic circuit, create a 2^N -row truth table (excluding the header row), and list all combinations of the inputs in order
- Step 2: Create one column for each intermediate node

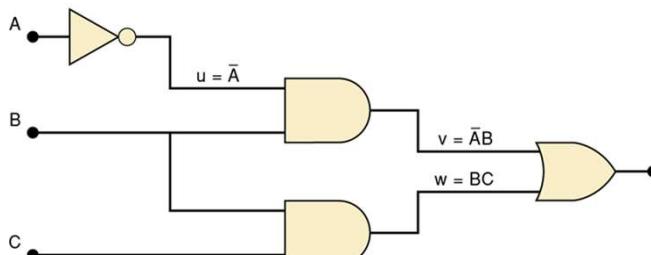


A	B	C	U = \bar{A}	V = $\bar{A}B$	W = BC	X
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

22

TRUTH TABLE OF LOGIC CIRCUIT

- Step 3: Fill in the values of each column accordingly

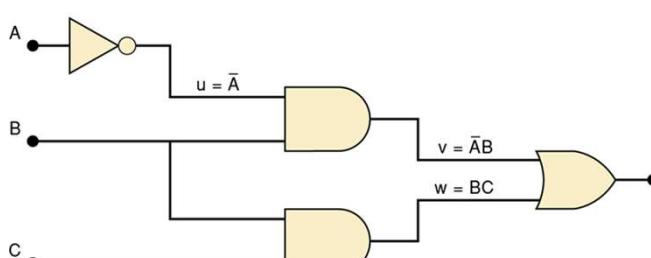


A	B	C	U = \bar{A}'	V = $\bar{A}'B$	W = BC	X
0	0	0	1			
0	0	1	1			
0	1	0	1			
0	1	1	1			
1	0	0	0			
1	0	1	0			
1	1	0	0			
1	1	1	0			

23

TRUTH TABLE OF LOGIC CIRCUIT

- Step 3: Fill in the values of each column accordingly

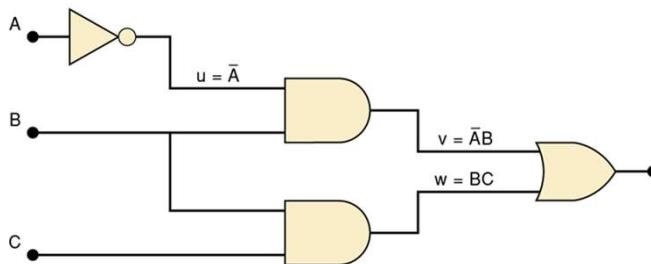


A	B	C	U = \bar{A}'	V = $\bar{A}'B$	W = BC	X
0	0	0	1	0		
0	0	1	1	0		
0	1	0	1	1		
0	1	1	1	1		
1	0	0	0	0		
1	0	1	0	0		
1	1	0	0	0		
1	1	1	0	0		

24

TRUTH TABLE OF LOGIC CIRCUIT

- Step 3: Fill in the values of each column accordingly

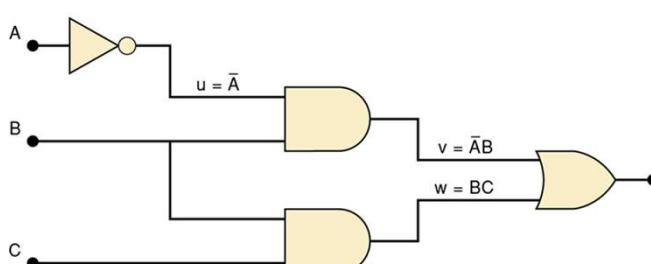


A	B	C	U = A'	V = A'B	W = BC	X
0	0	0	1	0	0	0
0	0	1	1	0	0	0
0	1	0	1	1	0	0
0	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	0	0	0	0
1	1	1	0	0	1	1

25

TRUTH TABLE OF LOGIC CIRCUIT

- Step 4: Until the output values are determined

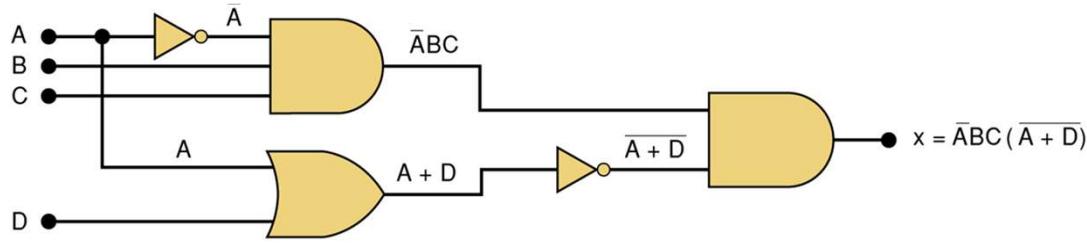


A	B	C	U = A'	V = A'B	W = BC	X
0	0	0	1	0	0	0
0	0	1	1	0	0	0
0	1	0	1	1	0	1
0	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	0	0	0	0
1	1	1	0	0	1	1

26

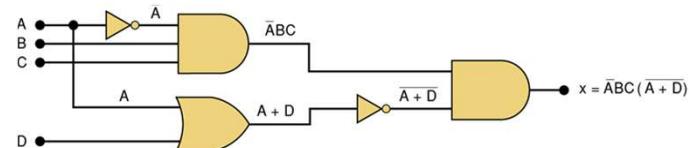
EXAMPLE

- In order to generate the truth table of the following logic circuit...
- How many combinations of input pattern? **16**
- How many intermediate nodes? **4**



27

EXAMPLE



A	B	C	D	\bar{A}	\bar{ABC}	$A + D$	$(\bar{A} + D)'$	X
0	0	0	0	1	0	0	1	0
0	0	0	1	1	0	1	0	0
0	0	1	0	1	0	0	1	0
0	0	1	1	1	0	1	0	0
0	1	0	0	0	1	0	1	0
0	1	0	1	1	1	1	0	0
0	1	1	0	0	1	0	1	1
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	1	0	0	0	0	0	1	0
1	1	0	1	0	0	1	0	0
1	1	1	0	0	0	1	0	0
1	1	1	1	0	0	1	0	0

28

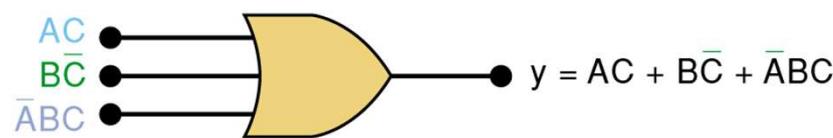
BOOLEAN EXPRESSION TO LOGIC CIRCUIT

- From the Boolean expression, we can also construct the logic circuit directly
- For example, for expression $X=ABC$
 - Use three-input AND gate
- For example, for expression $X=A+B$
 - Use two-input OR gate

29

BOOLEAN EXPRESSION TO LOGIC CIRCUIT

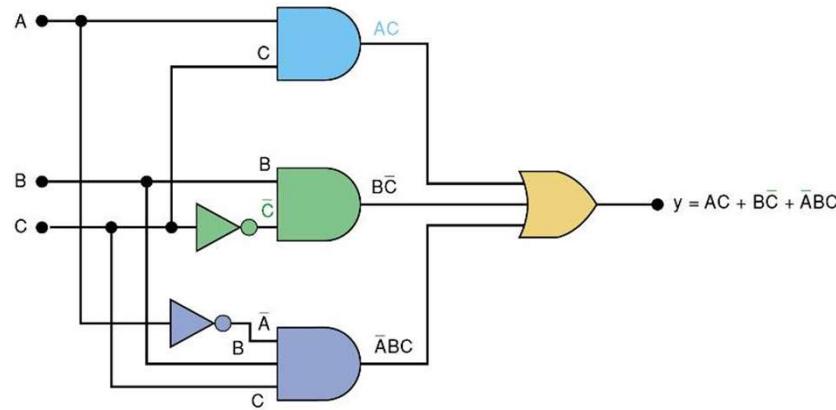
- For expression, $Y=AC+BC'+A'BC$
- Use 3-input OR gate as below



30

BOOLEAN EXPRESSION TO LOGIC CIRCUIT

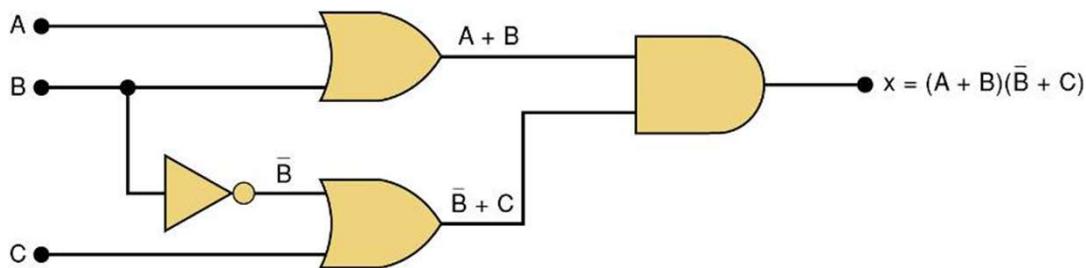
- Further implement the 3 inputs by using 2-input and 3-input AND gates accordingly



31

EXAMPLE

- $X = (A+B)(B'+C)$



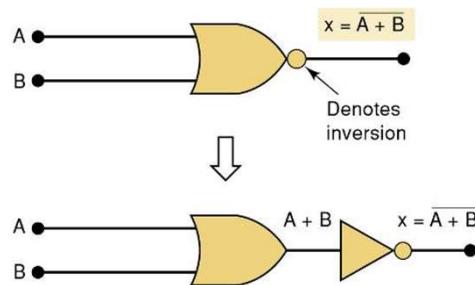
32

33

MORE LOGIC GATES

NOR GATE

- NOR gate is an inverted OR gate
- An inversion “bubble” is placed at the output of the OR gate, $X = (\bar{A} + \bar{B})'$

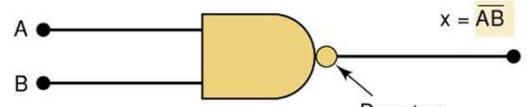


		OR	NOR
A	B	$A + B$	$\bar{A} + \bar{B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

34

NAND GATE

- NAND gate is an inverted AND gate
- $X = (AB)'$



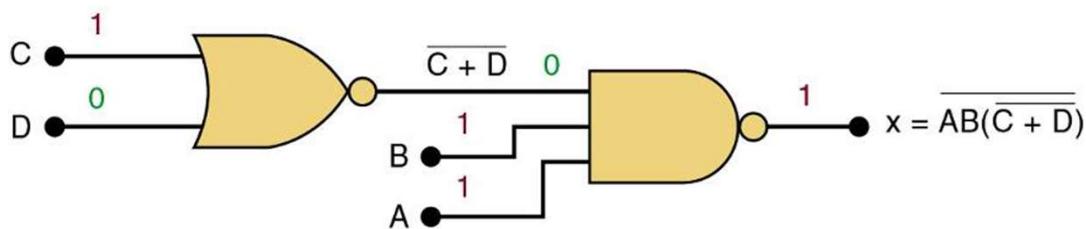
		AND	NAND
A	B	AB	\overline{AB}
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

(c)

35

USING NOR & NAND GATES

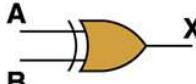
- Implementation of $X=(AB(C+D))'$ by using NOR and NAND gates



36

XOR GATE

- Similar to OR gate, except that when all inputs are 1s, the output will be 0

Boolean Expression	Logic Diagram Symbol	Truth Table															
$X = A \oplus B$		<table border="1"> <thead> <tr> <th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <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> </tbody> </table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	0
A	B	X															
0	0	0															
0	1	1															
1	0	1															
1	1	0															

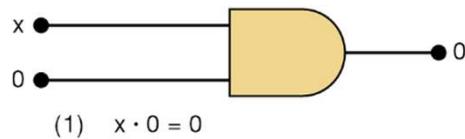
37

38

BOOLEAN ALGEBRA

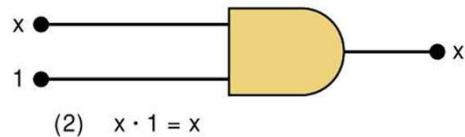
BOOLEAN THEOREMS

- Theorem 1



Theorem (1) states that if any variable is ANDed with 0, the result must be 0.

- Theorem 2

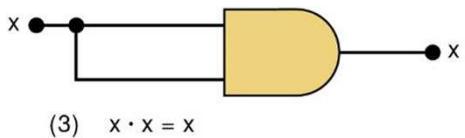


Theorem (2) is also obvious by comparison with ordinary multiplication.

39

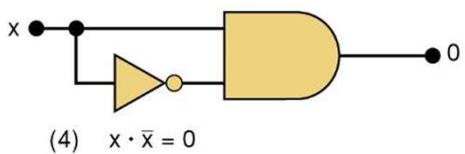
BOOLEAN THEOREMS

- Theorem 3



Prove Theorem (3) by trying each case.
 If $x = 0$, then $0 \cdot 0 = 0$
 If $x = 1$, then $1 \cdot 1 = 1$
 Thus, $x \cdot x = x$

- Theorem 4

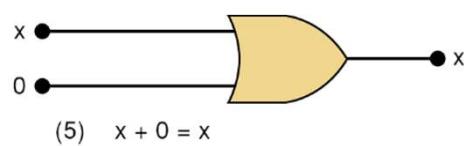


Theorem (4) can be proved in the same manner.

40

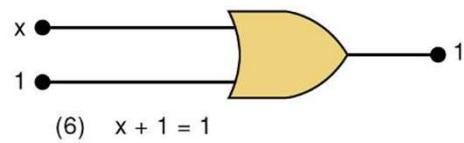
BOOLEAN THEOREMS

- Theorem 5



Theorem (5) is straightforward, as 0 added to anything does not affect value, either in regular addition or in OR addition.

- Theorem 6

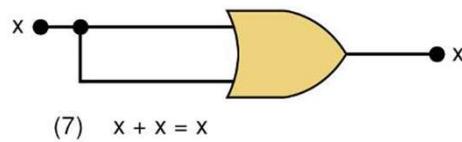


Theorem (6) states that if any variable is ORed with 1, the result is always 1. Check values: $0 + 1 = 1$ and $1 + 1 = 1$.

41

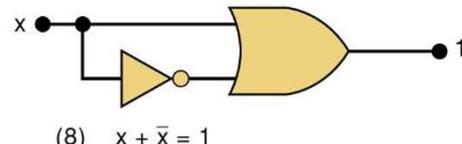
BOOLEAN THEOREMS

- Theorem 7



Theorem (7) can be proved by checking for both values of x:
 $0 + 0 = 0$ and $1 + 1 = 1$.

- Theorem 8



Theorem (8) can be proved similarly.

42

BOOLEAN THEOREMS - MULTIVARIABLE

- Theorem 9 and 10 – Commutative Laws

$$(9) \quad x + y = y + x$$

$$(10) \quad x \cdot y = y \cdot x$$

- Theorem 11 and 12 – Associative Laws

$$(11) \quad x + (y + z) = (x + y) + z = x + y + z$$

$$(12) \quad x(yz) = (xy)z = xyz$$

- Theorem 13a and 13b – Distributive Law

$$(13a) \quad x(y + z) = xy + xz$$

$$(13b) \quad (w + x)(y + z) = wy + xy + wz + xz$$

43

BOOLEAN THEOREMS - MULTIVARIABLE

- Theorem 14 and 15 – do not have counterparts in ordinary algebra

15a proof

$$\begin{aligned} x+x' &= x(1+y)+x' \\ y &= x+xy+x' \\ x+y(x+x') &= x+y \end{aligned}$$

$$(14) \quad x + \underline{xy} = x$$

$$(15a) \quad \underline{x} + \underline{xy} = \underline{x} + y$$

$$(15b) \quad \underline{x} + xy = \underline{x} + y$$

- Proof of Theorem 14:

$$\begin{aligned} x + xy &= x(1 + y) \\ &= x \cdot 1 \quad [\text{using theorem (6)}] \\ &= x \quad [\text{using theorem (2)}] \end{aligned}$$

x	y	xy	x + xy
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

44

BOOLEAN THEOREMS – DE MORGAN'S THEOREMS

- Theorem 16 and 17 – DeMorgan's theorems

$$(16) \quad (\overline{x + y}) = \overline{\overline{x} \cdot \overline{y}}$$

$$(17) \quad (\overline{x \cdot y}) = \overline{x} + \overline{y}$$

Theorem (16) says inverting the OR sum of two variables is the same as inverting each variable individually, then ANDing the inverted variables.

Theorem (17) says inverting the AND product of two variables is the same as inverting each variable individually and then ORing them.

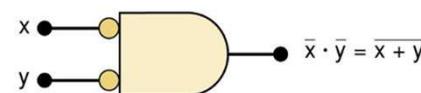
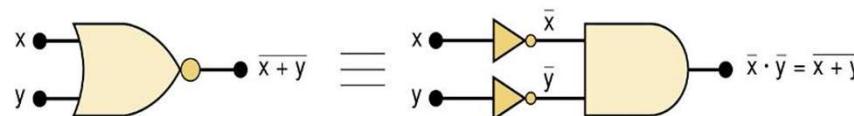
***Exam appear DeMorgan thm proof

45

BOOLEAN THEOREMS – DE MORGAN'S THEOREMS

- Theorem 16

$$(16) \quad (\overline{x + y}) = \overline{\overline{x} \cdot \overline{y}}$$

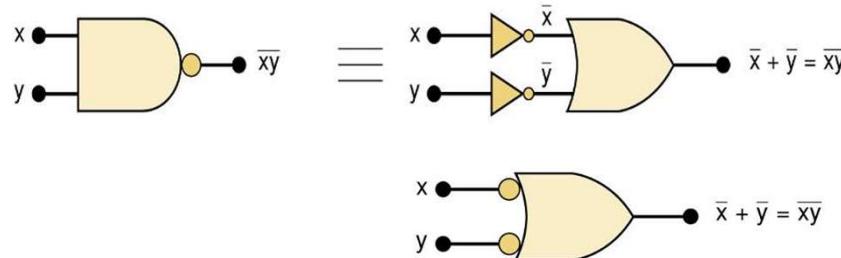


46

BOOLEAN THEOREMS – DE MORGAN'S THEOREMS

- Theorem 17

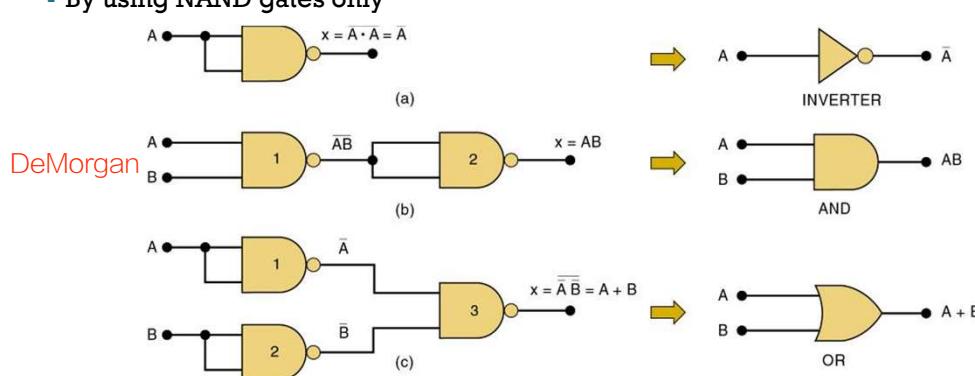
$$(17) \quad (\overline{x \cdot y}) = \overline{x} + \overline{y}$$



47

UNIVERSALITY OF NAND & NOR GATES

- NAND or NOR gates can be used to create the three basic logic gates, AND, OR and NOT gates
- By using NAND gates only

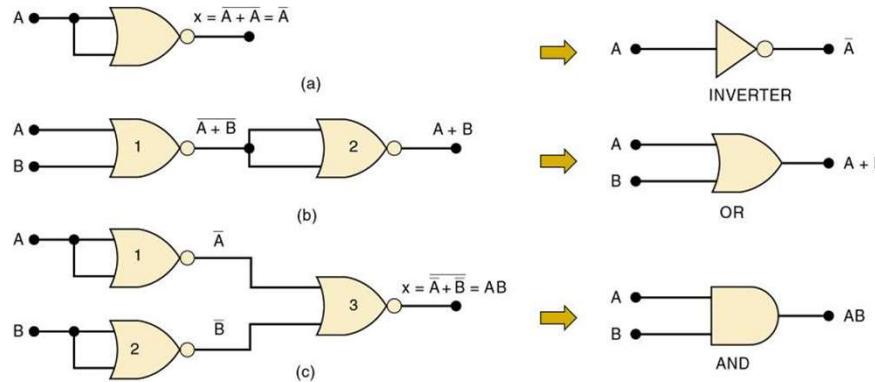


48

Many IC apply NAND NOR gate only to form
coz The combination of them can do every function

UNIVERSALITY OF NAND & NOR GATES

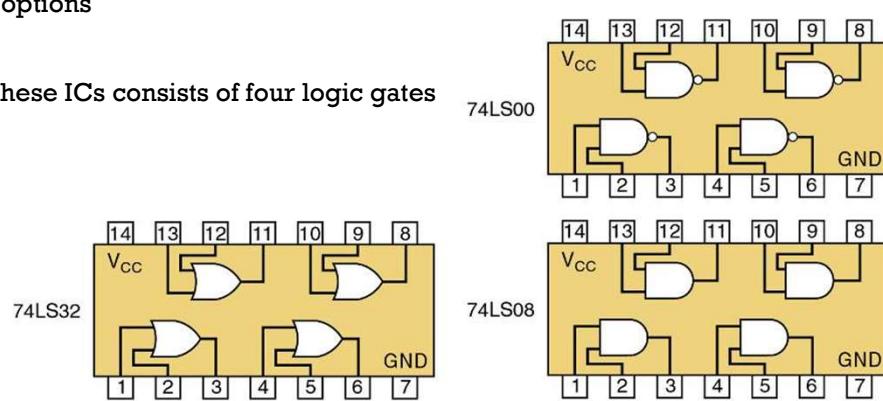
- By using NOR gates only



49

UNIVERSALITY OF NAND & NOR GATES

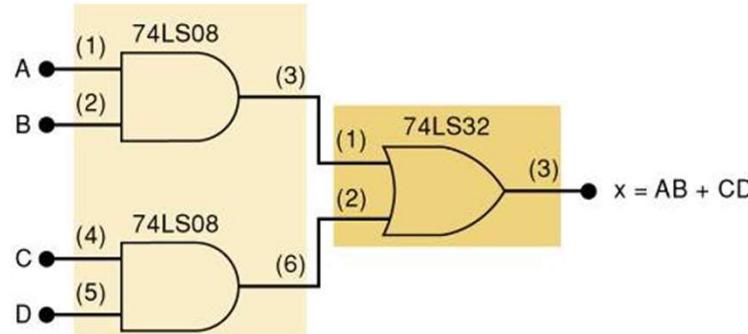
- Suppose we have to implement a logic function, $X = AB + CD$, with the following chips as options
- Each of these ICs consists of four logic gates



50

UNIVERSALITY OF NAND & NOR GATES

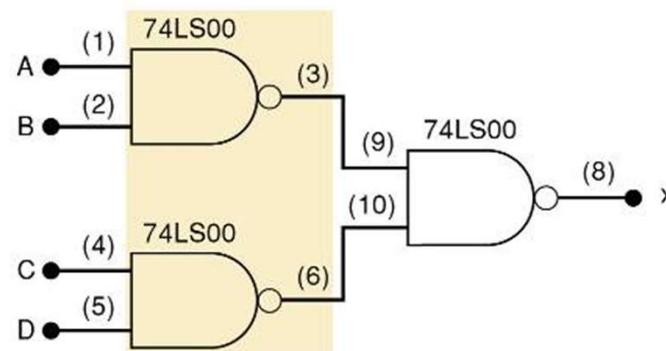
- Implementation 1 – using one 74LS08 and one 74LS32



51

UNIVERSALITY OF NAND & NOR GATES

- Implementation 2 – using only one 74LS00



52

53

ALTERNATIVE LOGIC GATE REPRESENTATION



ACTIVE-HIGH AND ACTIVE-LOW

- An input/output can be **active-HIGH** or **active-LOW**
 - Active-HIGH is indicated by an input/output has no inversion bubble
 - Active-LOW is indicated by an input/output has an inversion bubble
- In the circuit diagram, the state-HIGH or active-LOW pins are commonly **labelled** without bar or with bar respectively
- For example, the READ (RD) signal of a memory

A bar over a signal
means
ACTIVE LOW.

\overline{RD}

Absence of a bar
means asserted
(active) HIGH

RD

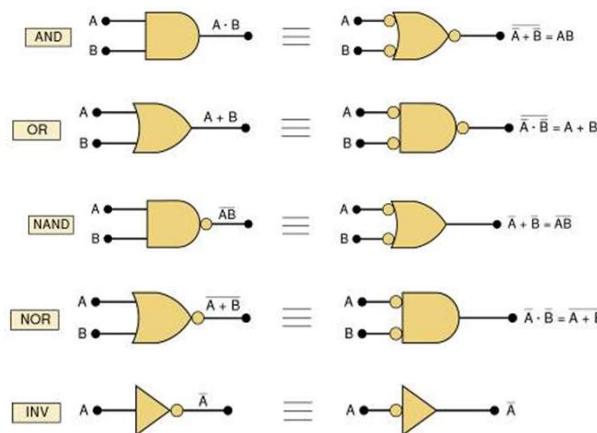
54

ACTIVE-HIGH AND ACTIVE-LOW

- In order to implement different conditions with the same logic, alternate logic gate representation is useful
- To convert a standard symbol to an alternate for AND, OR, NAND, NOR gates
 - Change the basic gate from AND to OR, or OR to AND
 - Add an inversion bubble if there is no inversion
 - Remove inversion bubble if there is inversion

55

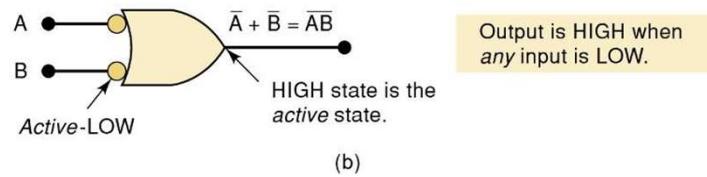
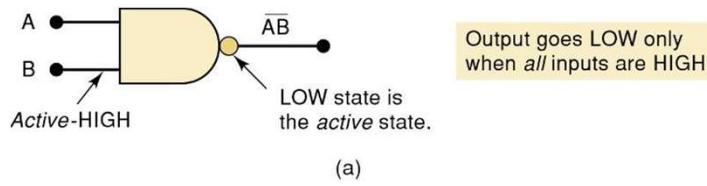
ALTERNATE REPRESENTATIONS



56

ALTERNATE REPRESENTATIONS OF NAND

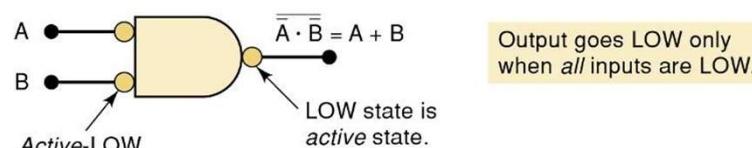
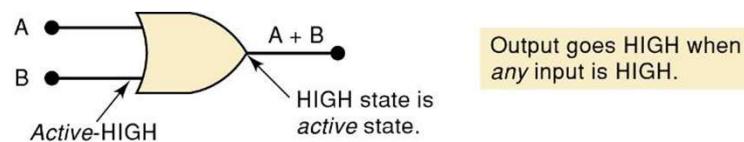
- Implementations of NAND gates in either active-HIGH or active-LOW



57

ALTERNATE REPRESENTATIONS OF OR

- Different implementations of OR gates in either active-HIGH or active-LOW

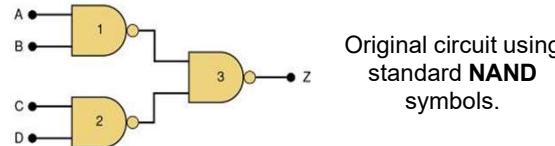


58

ALTERNATE REPRESENTATIONS

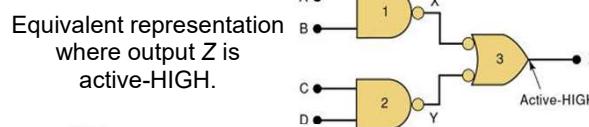
- Proper use of alternate gate symbols in the circuit diagram can make it clearer

- For example,

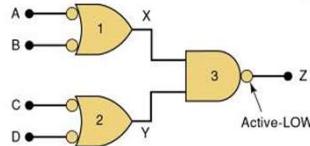


Original circuit using standard **NAND** symbols.

Equivalent representation where output Z is active-HIGH.



Equivalent representation where output Z is active-LOW.

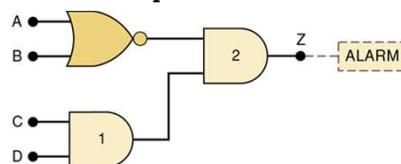


59

ALTERNATE REPRESENTATIONS

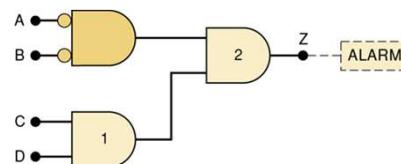
- If possible, choose an alternative gate symbols, so that

- Bubble outputs are connected to bubble inputs
- Non-bubble outputs are connected to non-bubble outputs



Modify the circuit diagram so it represents the circuit operation more effectively.

The NOR gate symbol should be changed to the alternate symbol with a non-bubble (active-HIGH) output to match the non-bubble input of AND gate 2.



60

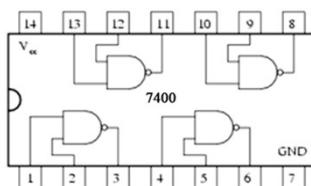
61

PHYSICAL CHARACTERISTICS OF LOGIC IC



DATASHEET

Datasheet of 74LS00



SN74LS00

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
V_{IH}	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
V_{IL}	Input LOW Voltage			0.8	V	Guaranteed Input LOW Voltage for All Inputs
V_{IK}	Input Clamp Diode Voltage	-0.65	-1.5		V	$V_{CC} = \text{MIN}, I_{IN} = -18 \text{ mA}$
V_{OH}	Output HIGH Voltage	2.7	3.5		V	$V_{CC} = \text{MIN}, I_{OL} = \text{MAX}, V_{IN} = V_{IH}$ or V_{IL} per Truth Table
V_{OL}	Output LOW Voltage	0.25	0.4		V	$I_{OL} = 4.0 \text{ mA}, V_{CC} = V_{CC} \text{ MIN}, V_N = V_{IL} \text{ or } V_{IH}$ per Truth Table
		0.35	0.5		V	$I_{OL} = 8.0 \text{ mA}, V_{CC} = \text{MAX}, V_N = 2.7 \text{ V}$
I_{IH}	Input HIGH Current		20	μA		$V_{CC} = \text{MAX}, V_N = 2.7 \text{ V}$
			0.1	mA		$V_{CC} = \text{MAX}, V_N = 7.0 \text{ V}$
I_{IL}	Input LOW Current		-0.4	mA		$V_{CC} = \text{MAX}, V_N = 0.4 \text{ V}$
I_{OS}	Short Circuit Current (Note 1)	-20	-100	mA		$V_{CC} = \text{MAX}$
I_{CC}	Power Supply Current Total, Output HIGH			1.6	mA	$V_{CC} = \text{MAX}$
	Total, Output LOW			4.4		

Note 1: Not more than one output should be shorted at a time, nor for more than 1 second.

AC CHARACTERISTICS ($T_A = 25^\circ\text{C}$)

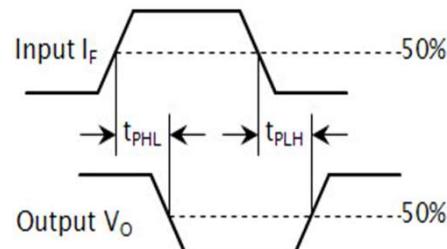
Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
t_{PLH}	Turn-Off Delay, Input to Output	9.0	15	ns		$V_{CC} = 5.0 \text{ V}$ $C_L = 15 \text{ pF}$
t_{PHL}	Turn-On Delay, Input to Output	10	15	ns		

62

PROPAGATION DELAY

- Due to limitations in **transistor switching speeds** caused by undesirable internal **capacitive stored charges**
- A **delay** time for an input wave (at 50% point of input) to propagate through an IC to the output (at 50% point of output)
 - Time propagation from HIGH to LOW (t_{PHL})
 - Time propagation from LOW to HIGH (t_{PLH})

二極體(Diode): current control

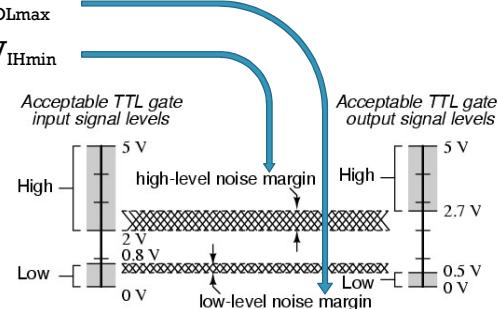


63

NOISE MARGIN

- Noise is all undesirable voltage variations that are superimposed on normal operating voltage levels
- Noise margin is the maximum noise voltage level that a logic signal can tolerate without errors
- Noise margin for LOW output = $V_{ILmax} - V_{OLmin}$
- Noise margin for HIGH output = $V_{OHmin} - V_{IHmax}$

Parameter	Minimum	Typical	Maximum
V_{OL}		0.2 V	0.4 V
V_{IL}			0.8 V
V_{OH}	2.4 V	3.4 V	
V_{IH}	2.0 V		



64

FAN-IN AND FAN-OUT

- Fan-in: the **number of inputs** available on a logic gate
- Fan-out: the **number of logic gates** which the **output can drive**
 - $\text{Fan-out} = \min. \left\{ \frac{I_{OH}}{I_{IH}}, \frac{I_{OL}}{I_{IL}} \right\}$
- For example, $I_{OH}=600\mu\text{A}$, $I_{IH}=40\mu\text{A}$, $I_{OL}=16\mu\text{A}$, $I_{IL}=1.6\mu\text{A}$
 - $\text{Fan-out} = \min. \left\{ \frac{600}{40}, \frac{16}{1.6} \right\} = 10$
 - I_{OH} = Output Current of a gate when its output is HIGH
 - I_{OL} = Output Current of a gate when its output is LOW
 - I_{IH} = Input Current of a gate when its input is HIGH
 - I_{IL} = Input Current of a gate when its input is LOW

65

ANY QUESTIONS ?

66