



Computer Fundamentals

Dr. Safdar Nawaz Khan Marwat
DCSE, UET Peshawar

Lecture 6



Binary Logic Operations

- Also called Boolean logic operations
- Boolean variable can have only two possible values i.e. 0 or 1
- Logic operations or functions of Boolean variables
 - ❑ E.g. AND, OR, NOT etc.



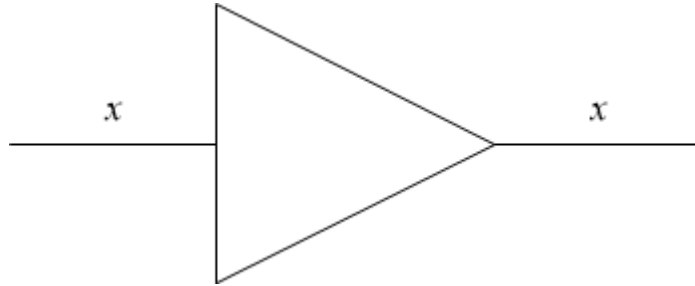
Binary Logic Operations

Name	Example	Symbolically
Buffer	$y = x$	x
NOT	$y = \text{NOT}(x)$	\bar{x}
AND	$z = x \text{ AND } y$	$x.y$
OR	$z = x \text{ OR } y$	$x + y$
XOR	$z = x \text{ XOR } y$	$x \oplus y$
NAND	$z = x \text{ NAND } y$	$\overline{x.y}$
NOR	$z = x \text{ NOR } y$	$\overline{x + y}$
XNOR	$z = x \text{ XNOR } y$	$\overline{x \oplus y}$

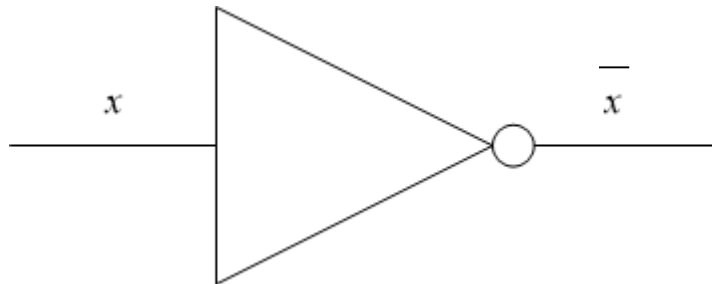


Diagrammatic Representation

➤ Buffer



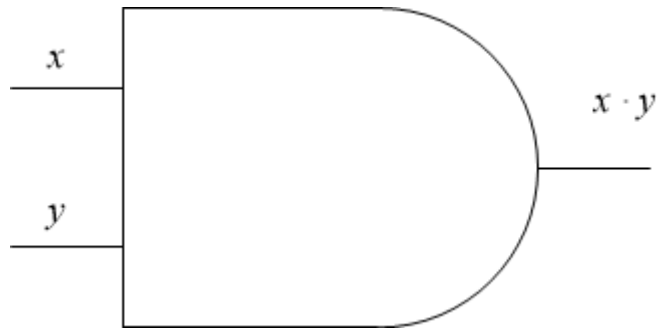
➤ NOT



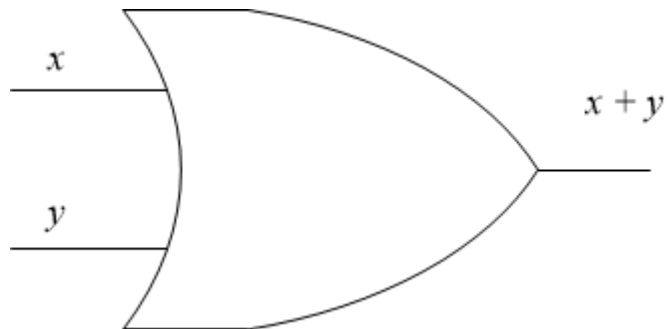


Diagrammatic Representation (cont.)

➤ AND



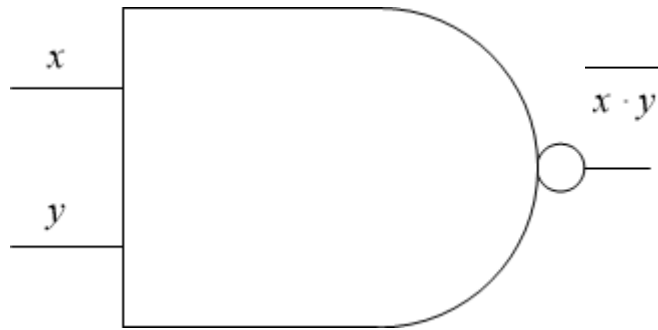
➤ OR



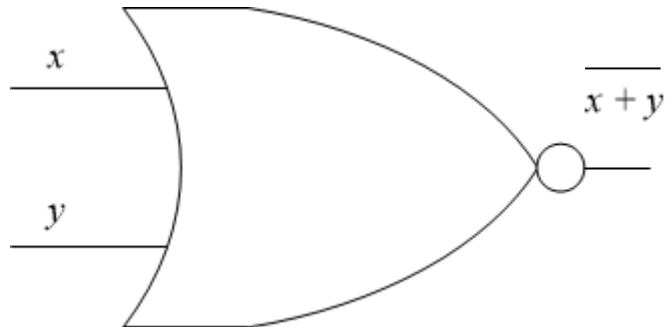


Diagrammatic Representation (cont.)

➤ NAND



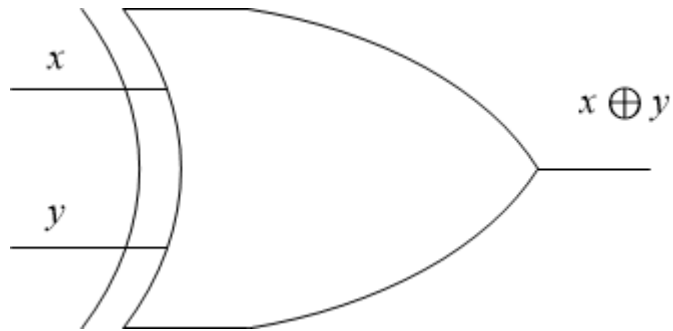
➤ NOR



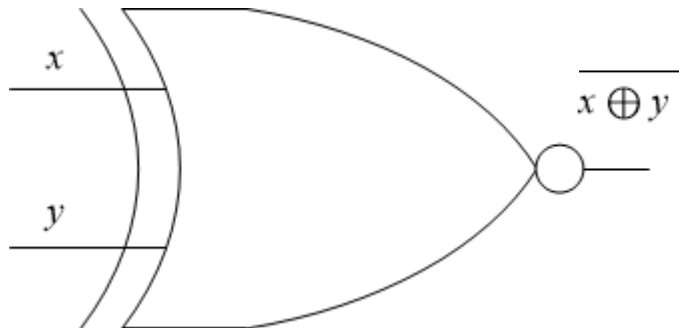


Diagrammatic Representation (cont.)

➤ XOR



➤ XNOR





Truth Table

- Defines the output of a logic function for all possible inputs
 - ❑ 2^n rows in a truth table (n is the number of inputs)
- Truth table for NOT

x	\overline{x}
0	1
1	0



Truth Table (cont.)

➤ Truth table for OR

x	y	$x + y$
0	0	
0	1	
1	0	
1	1	



Truth Table (cont.)

➤ Truth table for AND

x	y	$x.y$
0	0	
0	1	
1	0	
1	1	



Truth Table (cont.)

➤ Truth table for XOR

x	y	$x \oplus y$
0	0	
0	1	
1	0	
1	1	



Truth Table (cont.)

➤ Truth table for NOR

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



Truth Table (cont.)

➤ Truth table for NAND

x	y	$x.y$	$\overline{x.y}$
0	0		
0	1		
1	0		
1	1		



Truth Table (cont.)

➤ Truth table for XNOR

x	y	$x \oplus y$	$\overline{x \oplus y}$
0	0		
0	1		
1	0		
1	1		



Boolean Laws

➤ AND

$$0 \cdot x = 0$$

$$1 \cdot x = x$$

➤ OR

$$0 + x = x$$

$$1 + x = 1$$



Boolean Laws (cont.)

➤ AND

$$x \cdot x = x$$

$$x \cdot \bar{x} = 0$$

➤ OR

$$x + x = x$$

$$x + \bar{x} = 1$$



Boolean Laws (cont.)

➤ AND

❑ Commutativity

$$x \cdot y = y \cdot x$$

❑ Associativity

$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$

❑ Distributivity

$$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$$



Boolean Laws (cont.)

➤ OR

□ Commutativity

$$x + y = y + x$$

□ Associativity

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

□ Distributivity

$$x + (y \cdot z) = (x + y) \cdot (x + z)$$



Boolean Laws (cont.)

➤ DeMorgan's Law

□ NAND

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

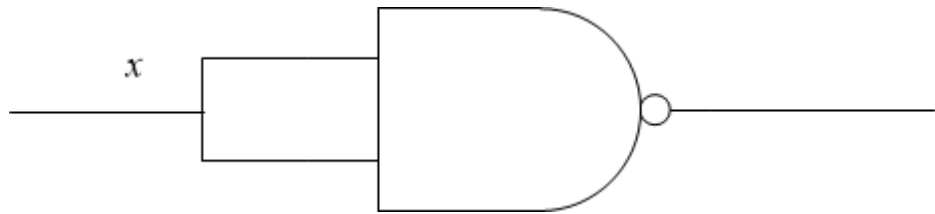
□ NOR

$$\overline{x + y} = \bar{x} \cdot \bar{y}$$



NAND for Every Gate

- Possible to emulate any gate
 - ❑ Using combination of NAND gates
- NOT gate with NAND
 - ❑ Compare with NOT



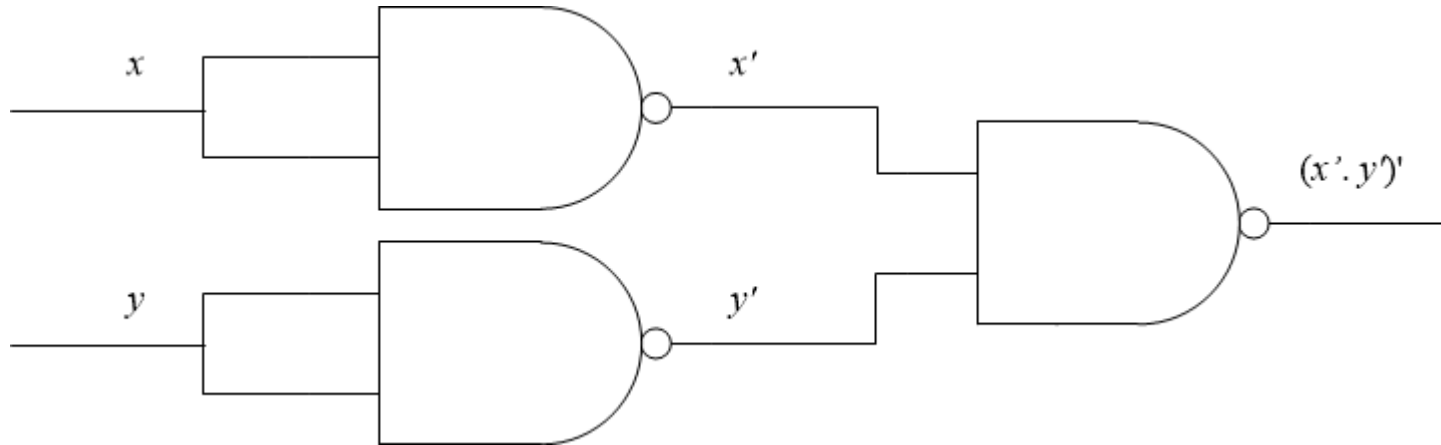
x	$x.x$	$\overline{x.x}$
0	0	1
1	1	0



NAND for Every Gate (cont.)

- OR gate with NAND
 - Recall DeMorgan's Law

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





NAND for Every Gate (cont.)

➤ OR gate with NAND

- Recall DeMorgan's Law

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

- Compare with OR

x	y	$\overline{\overline{x} \cdot \overline{x}} = \overline{x}$	$\overline{\overline{y} \cdot \overline{y}} = \overline{y}$	$\overline{\overline{x} \cdot \overline{y}}$
0	0	1	1	0
0	1	1	0	1
1	0	0	1	1
1	1	0	0	1



Exercise

- Design AND gate using only NAND gates