CHW 261: Logic Design

Instructors:
Prof. Hala Zayed http://www.bu.edu.eg/staff/halazayed14
Dr. Ahmed Shalaby http://bu.edu.eg/staff/ahmedshalaby14#

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Digital Fundamentals

CHAPTER 3
Logic Gates

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

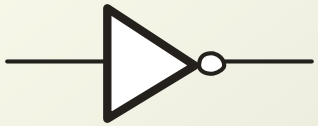
Logic Gates

- Inverter Gate
- AND Gate
- NAND Gate
- OR Gate
- NOR Gate
- Exclusive-OR Gate
- Exclusive-NOR Gate

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – Inverter

The Inverter



The inverter performs the Boolean **NOT** operation. When the input is LOW, the output is HIGH; when the input is HIGH, the output is LOW.

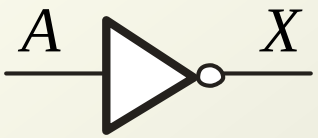
Input	Output
A	*X*
LOW (0)	HIGH (1)
HIGH (1)	LOW(0)

The **NOT** operation (complement) is shown with an overbar. Thus, the Boolean expression for an inverter is $X = \overline{A}$.

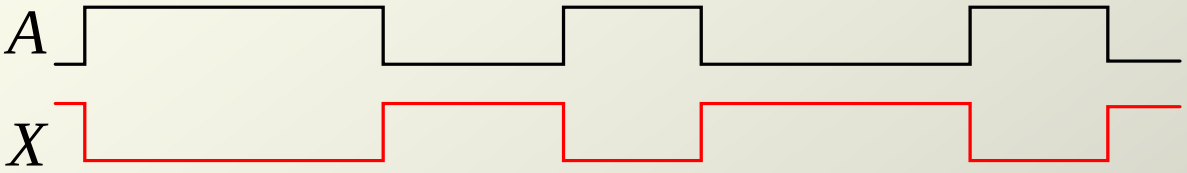
Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – Inverter

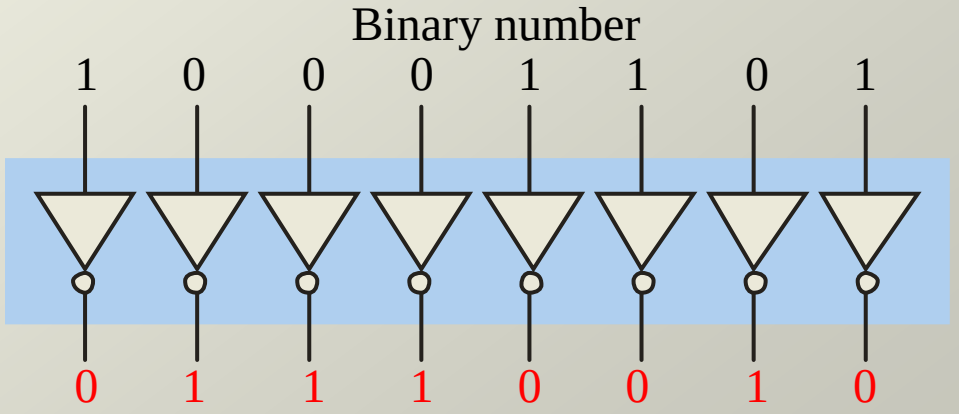
The Inverter



Example waveforms:

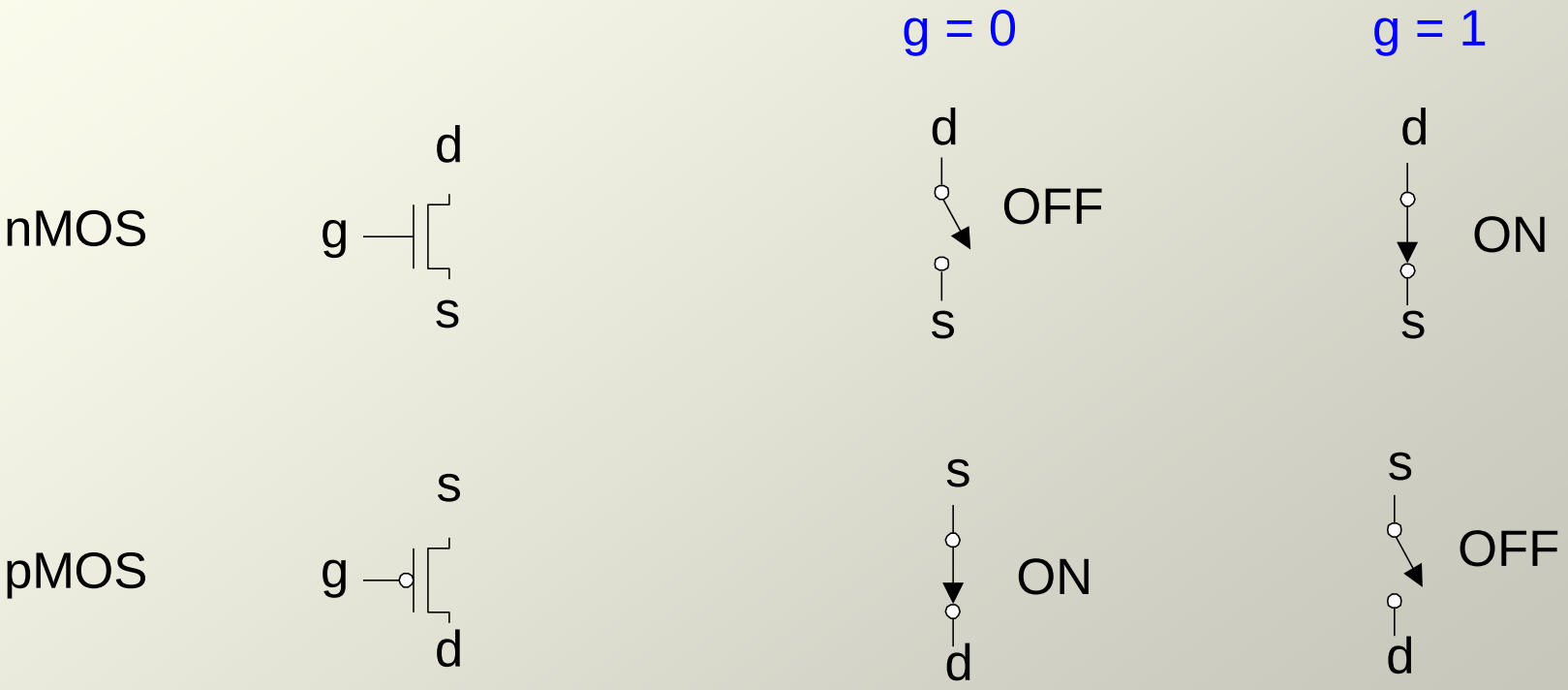


A group of inverters can be used to form the 1's complement of a binary number:



Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

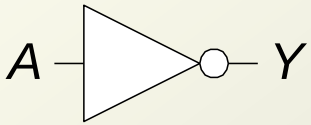
CMOS Transistor Function



Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

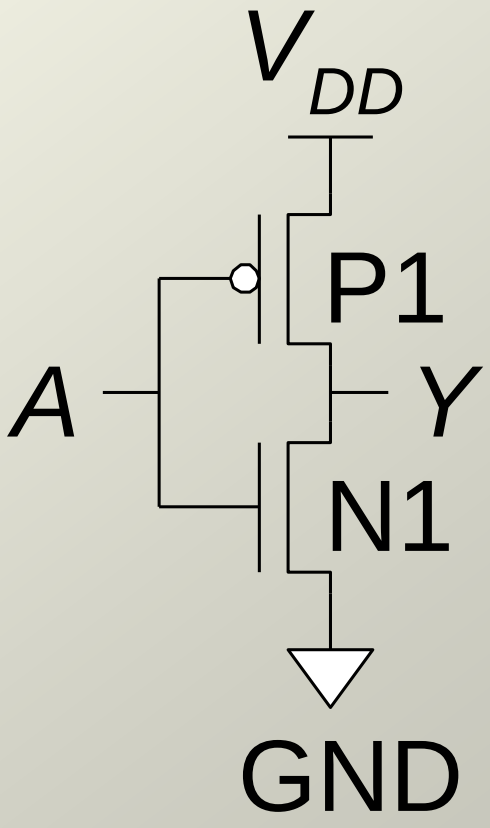
CMOS Gates – Inverter

NOT



$Y = \overline{A}$

A	Y
0	1
1	0



A	P1	N1	Y
0	ON	OFF	1
1	OFF	ON	0

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – AND

The AND Gate

The **AND gate** produces a HIGH output when all inputs are HIGH; otherwise, the output is LOW. For a 2-input gate, the truth table is

Inputs		Output
A	*B*	*X*
0	0	0
0	1	0
1	0	0
1	1	1

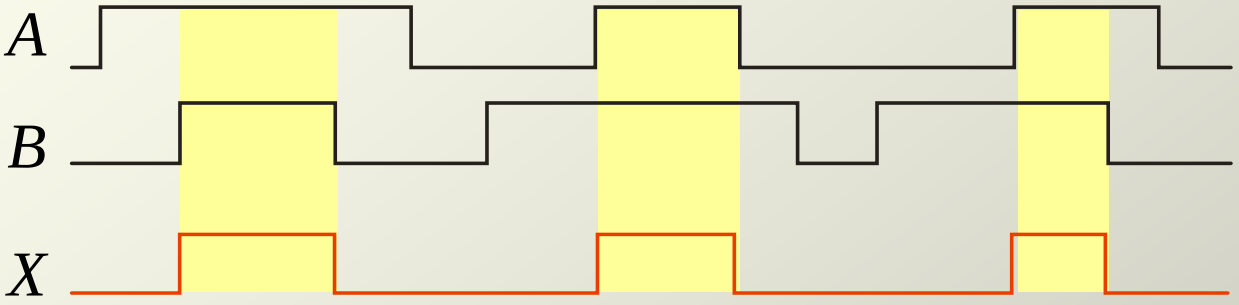
The **AND** operation is usually shown with a dot between the variables but it may be implied (no dot). Thus, the AND operation is written as $X = A \cdot B$ or $X = AB$.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – AND

The AND Gate

Example waveforms:



The AND operation is used in computer programming as a selective mask. If you want to retain certain bits of a binary number but reset the other bits to 0, you could set a mask with 1's in the position of the retained bits.

Example

If the binary number 10100011 is ANDed with the mask 00001111, what is the result? 00000011

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – NAND

The NAND Gate

The **NAND gate** produces a LOW output when all inputs are HIGH; otherwise, the output is HIGH. For a 2-input gate, the truth table is

Inputs		Output
A	*B*	*X*
0	0	1
0	1	1
1	0	1
1	1	0

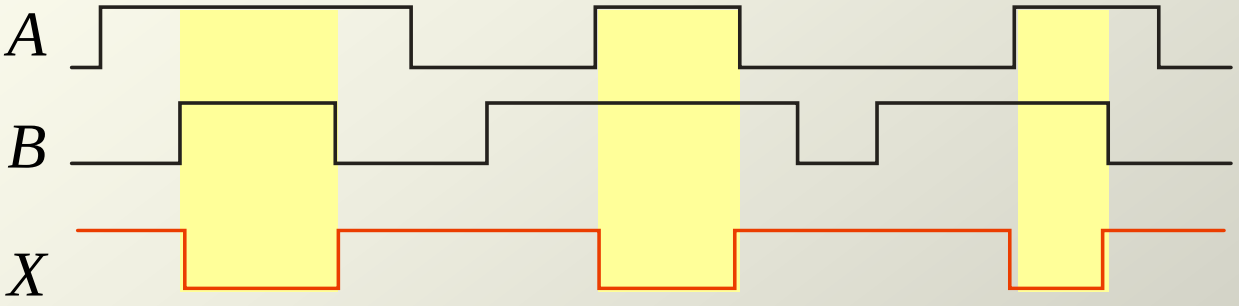
The **NAND** operation is shown with a dot between the variables and an overbar covering them. Thus, the NAND operation is written as $X = \overline{A \cdot B}$ (Alternatively, $X = \overline{AB}$.)

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – NAND

The NAND Gate

Example waveforms:

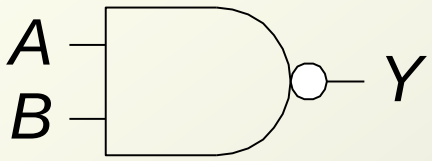


The NAND gate is particularly useful because it is a "universal" gate – all other basic gates can be constructed from NAND gates.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

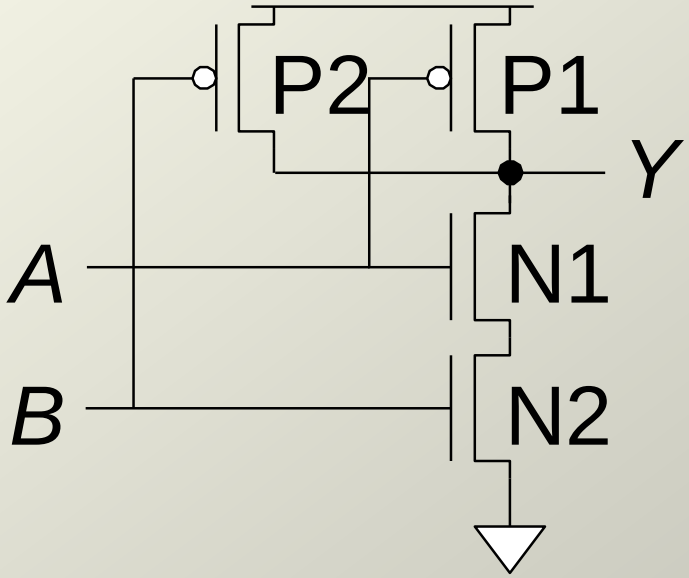
Logic Gates – NAND

NAND



$$Y = \overline{AB}$$

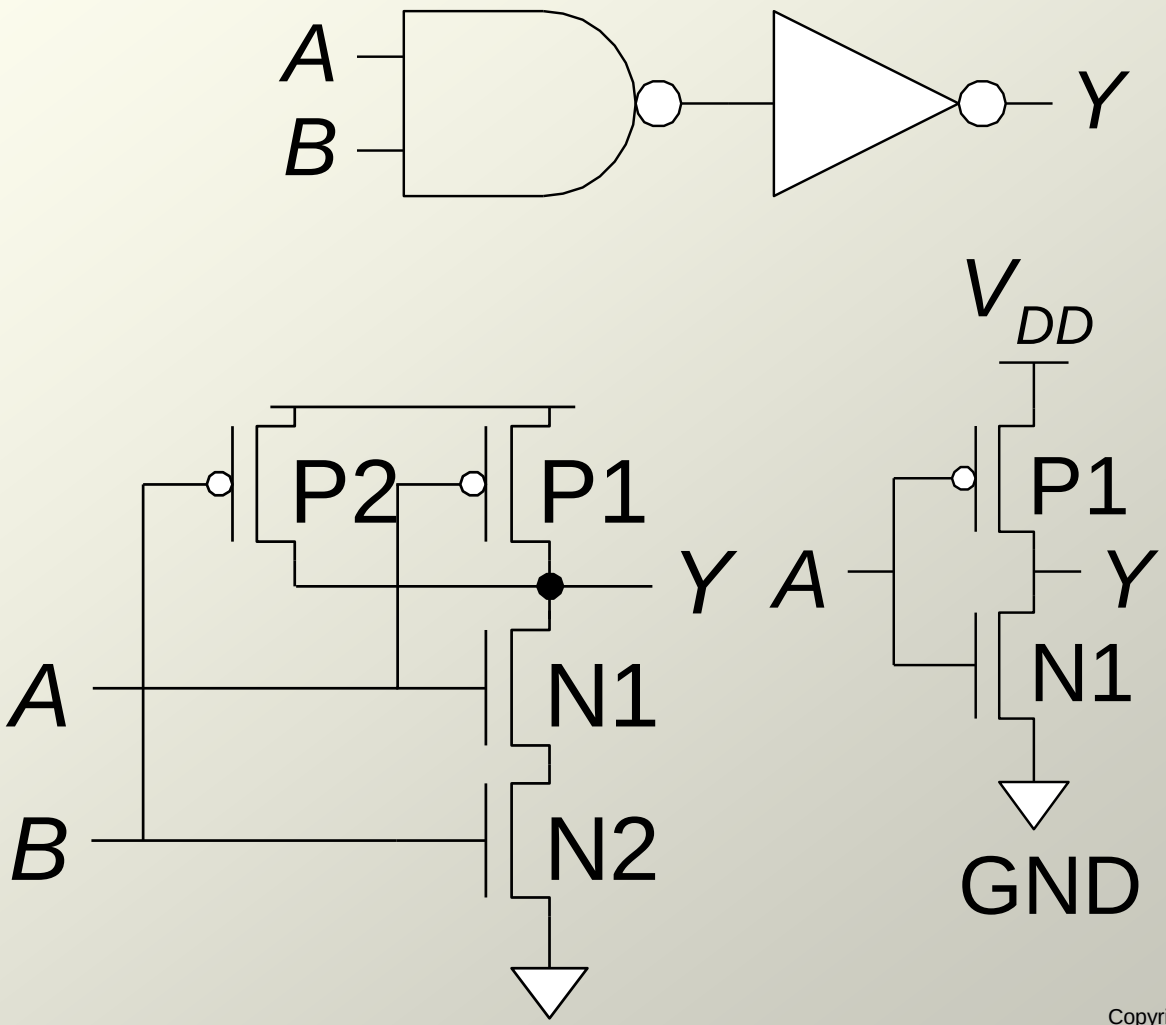
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



A	B	P1	P2	N1	N2	Y
0	0	ON	ON	OFF	OFF	1
0	1	ON	OFF	OFF	ON	1
1	0	OFF	ON	ON	OFF	1
1	1	OFF	OFF	ON	ON	0

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – AND



Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – OR

The OR Gate

The **OR gate** produces a HIGH output if any input is HIGH; if all inputs are LOW, the output is LOW. For a 2-input gate, the truth table is

Inputs		Output
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	1

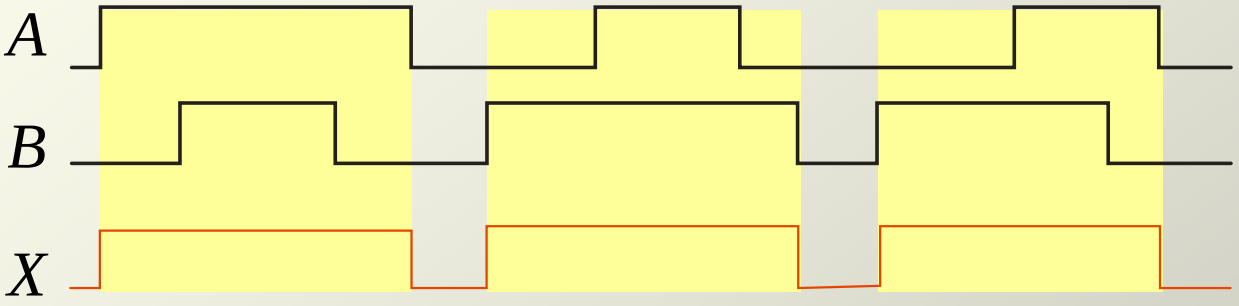
The **OR** operation is shown with a plus sign (+) between the variables. Thus, the OR operation is written as $X = A + B$.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – OR

The OR Gate

Example waveforms:



The OR operation can be used in computer programming to set certain bits of a binary number to 1.

Example ASCII letters have a 1 in the bit 5 position for lower case letters and a 0 in this position for capitals. (Bit positions are numbered from right to left starting with 0.) What will be the result if you OR an ASCII letter with the 8-bit mask 00100000?

Solution The resulting letter will be lower case.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – NOR

The NOR Gate

The **NOR gate** produces a LOW output if any input is HIGH; if all inputs are HIGH, the output is LOW. For a 2-input gate, the truth table is

Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	0

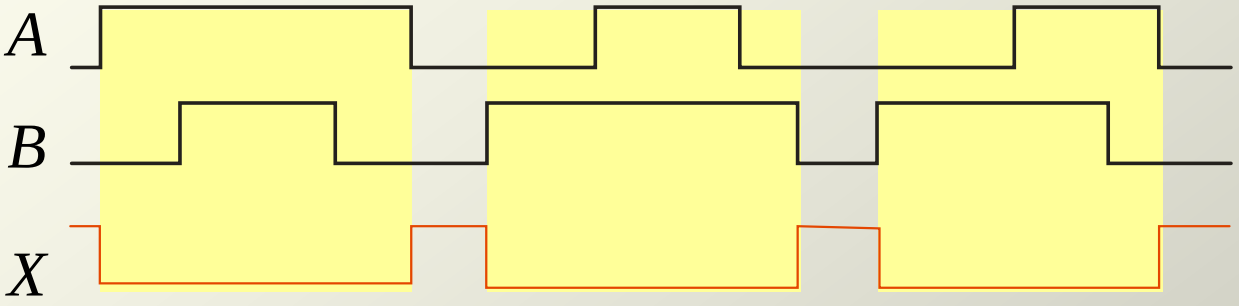
The **NOR** operation is shown with a plus sign (+) between the variables and an overbar covering them. Thus, the NOR operation is written as $X = \overline{A + B}$.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – NOR

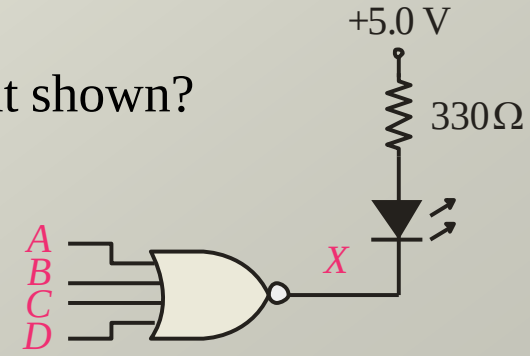
The NOR Gate

Example waveforms:



The NOR operation will produce a LOW if any input is HIGH.

Example When is the LED is ON for the circuit shown?



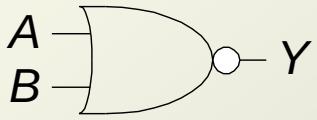
Solution The LED will be on when any of the four inputs are HIGH.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

QUIZ

Question How do you build a two-input NOR gate?

NOR



$Y = \overline{A + B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

A	B	P1	P2	N1	N2	Y
0	0					
0	1					
1	0					
1	1					

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – XOR

The XOR Gate

The **XOR gate** produces a HIGH output only when both inputs are at opposite logic levels. The truth table is

Inputs		Output
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	0

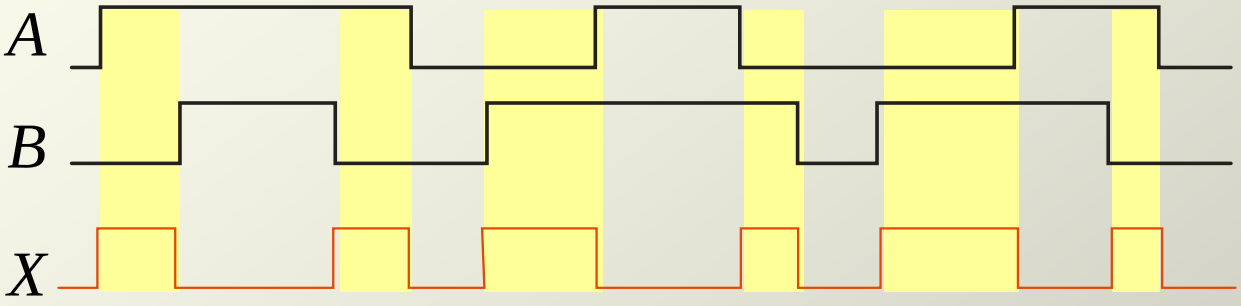
The **XOR** operation is written as $X = \overline{A}B + A\overline{B}$. Alternatively, it can be written with a circled plus sign between the variables as $X = A \oplus B$.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – XOR

The XOR Gate

Example waveforms:



Notice that the XOR gate will produce a HIGH only when exactly one input is HIGH.

Question If the *A* and *B* waveforms are both inverted for the above waveforms, how is the output affected?

There is no change in the output.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – XNOR

The XNOR Gate

The **XNOR gate** produces a HIGH output only when both inputs are at the same logic level. The truth table is

Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	1

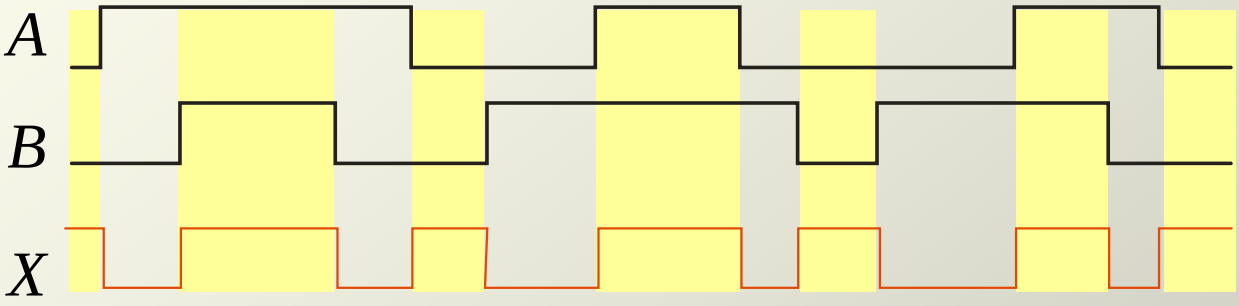
The **XNOR** operation shown as $X = \overline{A}\,\overline{B} + AB$. Alternatively, the XNOR operation can be shown with a circled dot between the variables. Thus, it can be shown as $X = A \odot B$.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates – XNOR

The XNOR Gate

Example waveforms:



Notice that the XNOR gate will produce a HIGH when both inputs are the same. This makes it useful for comparison functions.

Question If the *A* waveform is inverted but *B* remains the same, how is the output affected?

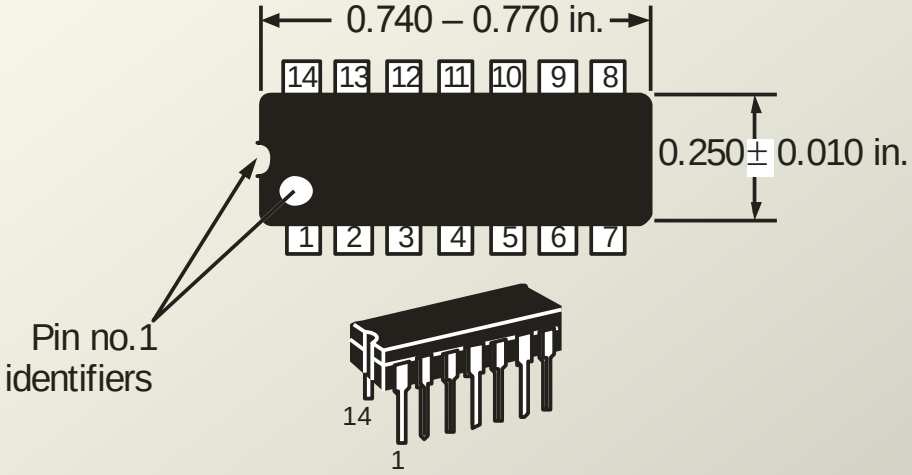
The output will be inverted.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

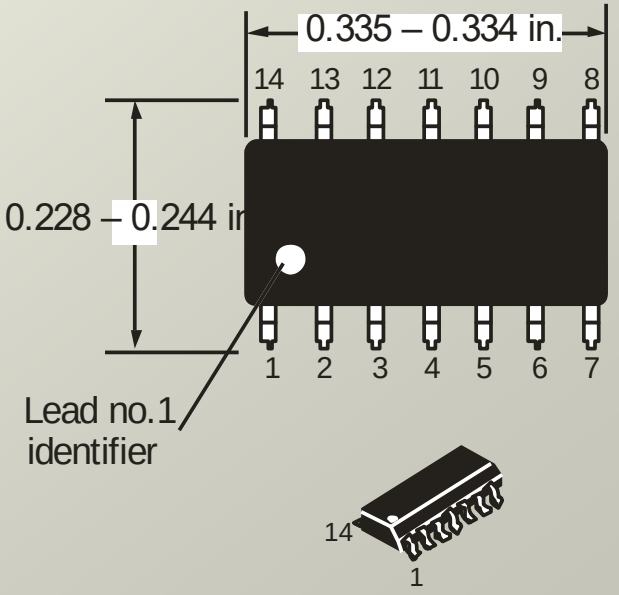
Logic Gates

Fixed Function Logic

Two major fixed function logic families are TTL and CMOS. A third technology is BiCMOS, which combines the first two. Packaging for fixed function logic is shown.



DIP package



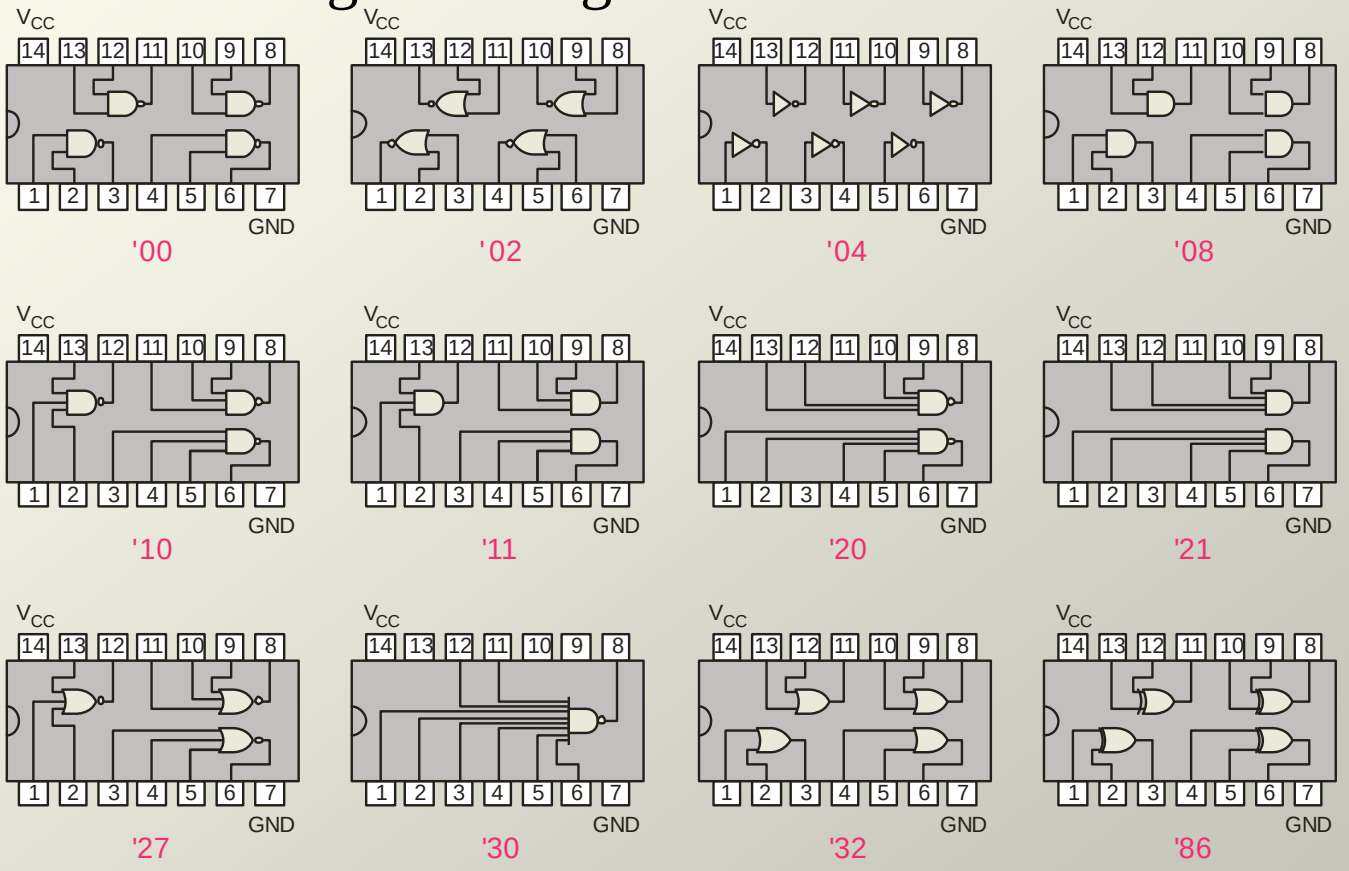
SOIC package

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates

Fixed Function Logic

Some common gate configurations are shown.

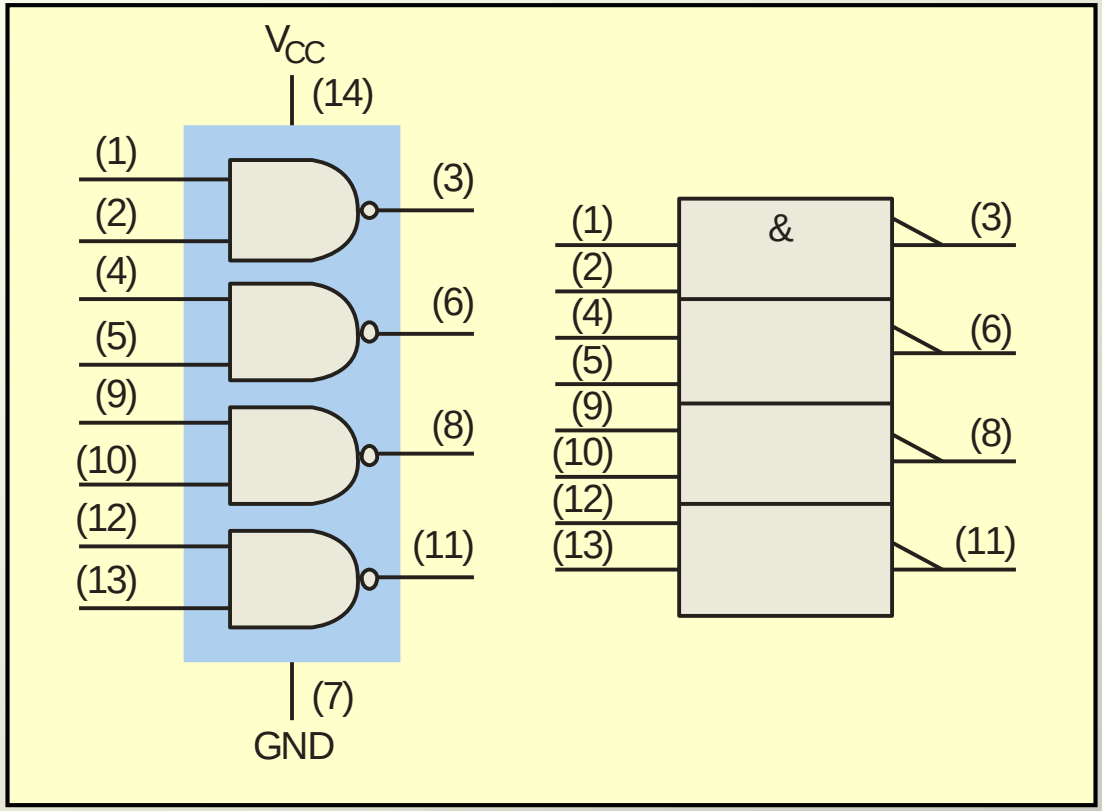


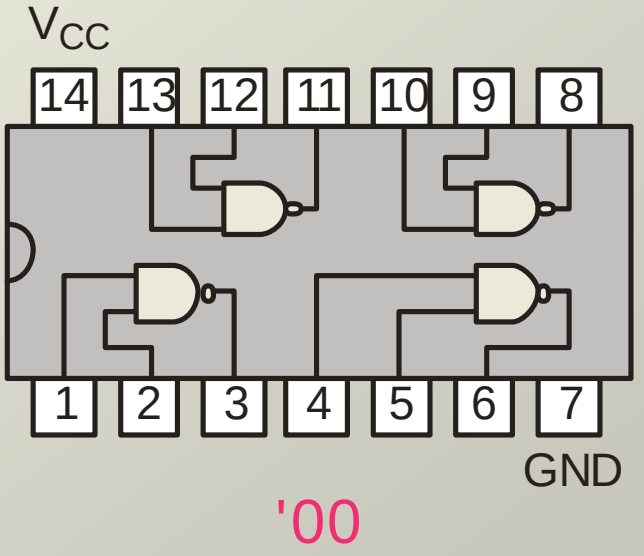
Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates

Fixed Function Logic

Logic symbols show the gates and associated pin numbers.





Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates

Fixed Function Logic

Data sheets include limits and conditions set by the manufacturer as well as DC and AC characteristics. For example, some maximum ratings for a 74HC00A are:

MAXIMUM RATINGS

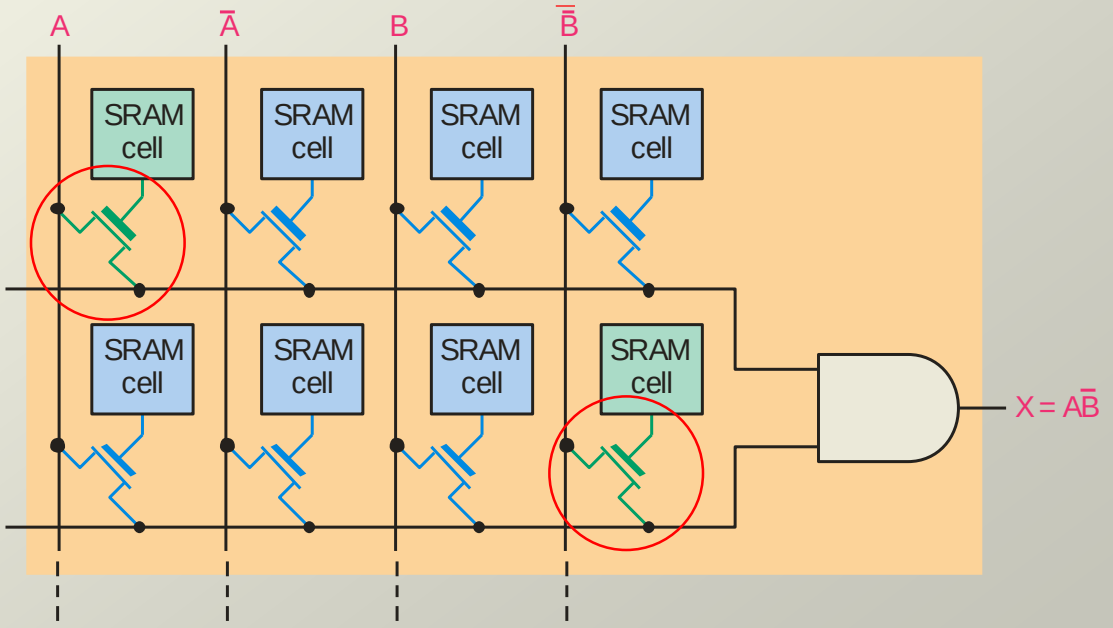
Symbol	Parameter	Value	Unit
V_{CC}	DC Supply Voltage (Referenced to GND)	–0.5 to + 7.0 V	V
V_{in}	DC InputVoltage (Referenced to GND)	–0.5 to V_{CC} +0.5 V	V
V_{out}	DC Output Voltage (Referenced to GND)	– 0.5 to V_{CC} +0.5 V	V
I_{in}	DC Input Current, per pin	±20	mA
I_{out}	DC Output Current, per pin	±25	mA
I_{CC}	DC Supply Current, V_{CC} and GND pins	±50	mA
P_D	Power Dissipation in Still Air, Plastic or Ceramic DIP † SOIC Package † TSSOP Package †	750 500 450	mW
T_{stg}	Storage Temperature	–65 to + 150	°C
T_L	Lead Temperature, 1 mm from Case for 10 Seconds Plastic DIP, SOIC, or TSSOP Package Ceramic DIP	 260 300	°C

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates - Programmable Logic

Programmable Logic

A Programmable Logic Device (PLD) can be programmed to implement logic. There are various technologies available for PLDs. Many use an internal array of AND gates to form logic terms. Many PLDs can be programmed multiple times.



Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates - Programmable Logic

Programmable Logic

In general, the required logic for a PLD is developed with the aid of a computer. The logic can be entered using a Hardware Description Language (HDL) such as VHDL. Logic can be specified to the HDL as a text file, a schematic diagram, or a state diagram.

Example

A text entry for a programming a PLD in VHDL as a 2-input NAND gate is shown for reference in the following slide. In this case, the inputs and outputs are first specified. Then the signals are described. Although you are probably not familiar with VHDL, you can see that the program is simple to read.

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates - Programmable Logic

Programmable Logic

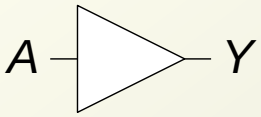
```
entity NandGate is
        port(A, B: in bit;
        LED: out bit);
end entity NandGate;

architecture GateBehavior of NandGate is
signal A, B: bit;
begin
        X <= A nand B;
        LED <= X;
end architecture GateBehavior;
```


Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates

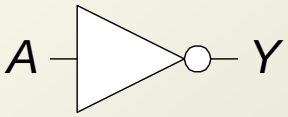
BUF



$Y = A$

A	Y
0	0
1	1

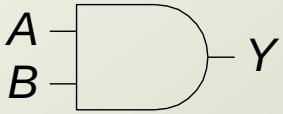
NOT



$Y = \overline{A}$

A	Y
0	1
1	0

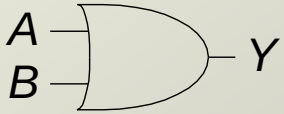
AND



$Y = AB$

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR



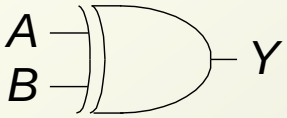
$Y = A + B$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates

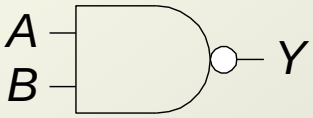
XOR



$Y = A \oplus B$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

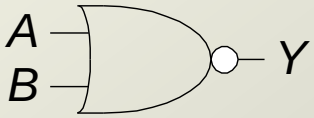
NAND



$Y = \overline{AB}$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

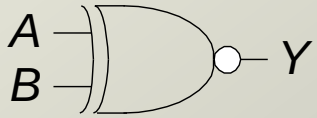
NOR



$Y = \overline{A + B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



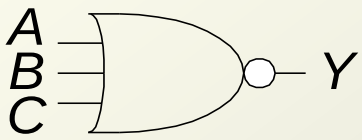
$Y = \overline{A \oplus B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Multiple-Input Logic Gates

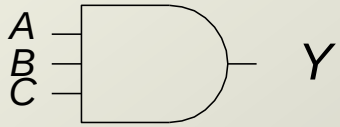
NOR3



$$Y = \overline{A+B+C}$$

A	B	C	Y
0	0	0	1
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	0

AND3

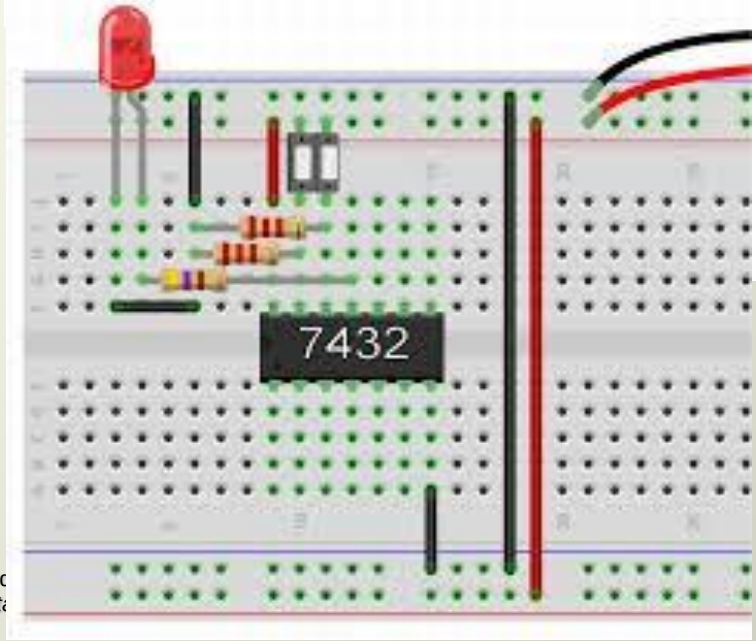


$$Y = ABC$$

A	B	C	Y
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

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates - Lab



Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.

Logic Gates - Lab

- **Breadboard Power Supply.**
- **Power Source (Battery, PowerBank, Laptop)**
- **USB Cable (female – female)**
- **LEDs (10 units – different colors)**
- **Resistor (10 units – 330 Ohm)**
- **Jumpers – Group of wires**
- **NAND Gate(00), NOR Gate(02), NOT Gate (04)**

Logic Design Lab

Copyright ©2006 by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458
All rights reserved.