

# Gates and Finite-State Machines

Instructor: Dmitri A. Gusev

Fall 2007

CS 210: Computing and Culture

Lecture 2, September 10, 2007

# Basic Definitions

A *gate* is a device that accepts one or more input electrical signals, and produces a single output electrical signal.

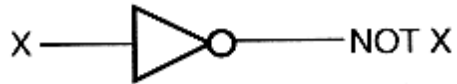
A *circuit* is a combination of gates.

*Boolean algebra* is an algebra in which variables and functions take on one of two values, 0 (“false”) or 1 (“true”).

A *logic diagram* is a graphical representation of a circuit, where each type of gate is represented by a specific graphical symbol.

A *truth table* is a table showing all possible input values and the associated output values.

# NOT Gate

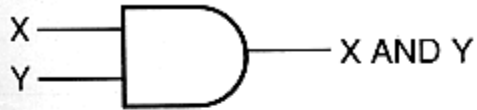


X	NOT X
0	1
1	0

A triangle with an *inversion bubble*.

*Boolean expression:  $Y = X'$*

# AND Gate

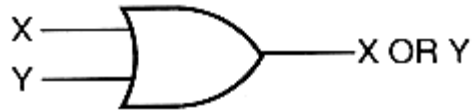


X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

Boolean expression:

$$A = X \cdot Y$$

# OR Gate



X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

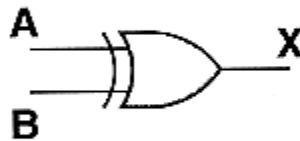
Boolean expression:  $A = X + Y$

# XOR Gate

**Boolean Expression**

$$X = A \oplus B$$

**Logic Diagram Symbol**



**Truth Table**

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

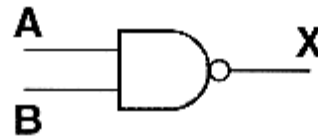
*Exclusive OR.*

# NAND Gate

**Boolean Expression**

$$X = (A \cdot B)'$$

**Logic Diagram Symbol**



**Truth Table**

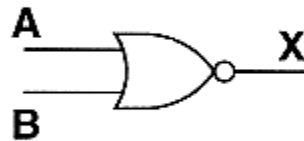
<b>A</b>	<b>B</b>	<b>X</b>
0	0	1
0	1	1
1	0	1
1	1	0

# NOR Gate

**Boolean Expression**

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

**Logic Diagram Symbol**



**Truth Table**

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

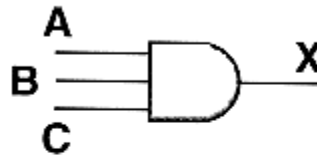


# Three-Input AND Gate

**Boolean Expression**

$$X = A \cdot B \cdot C$$

**Logic Diagram Symbol**



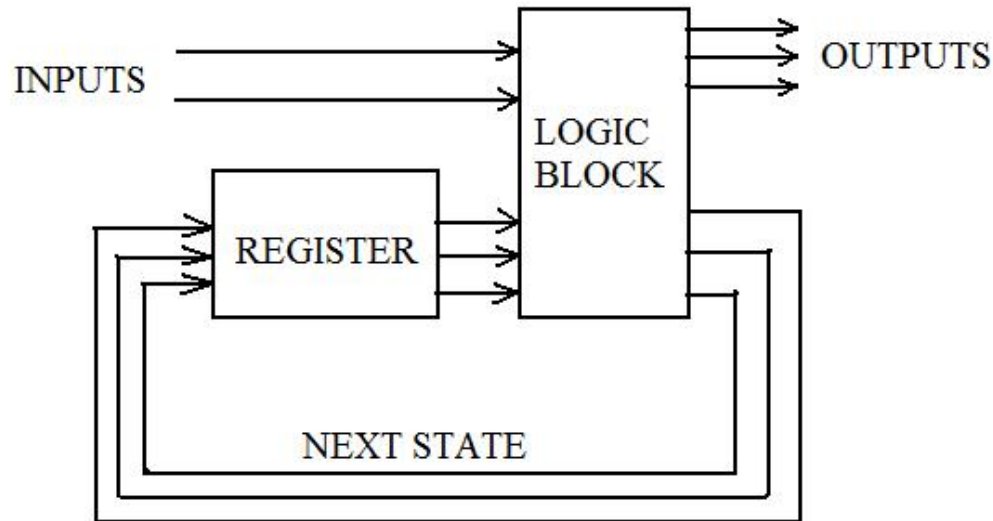
**Truth Table**

A	B	C	X
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

# Constructing Gates

Gates can be constructed using transistors.

# Finite-State Machines



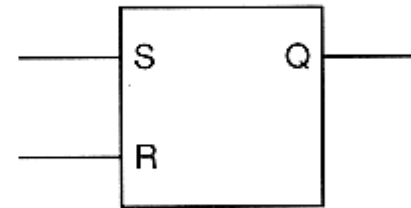
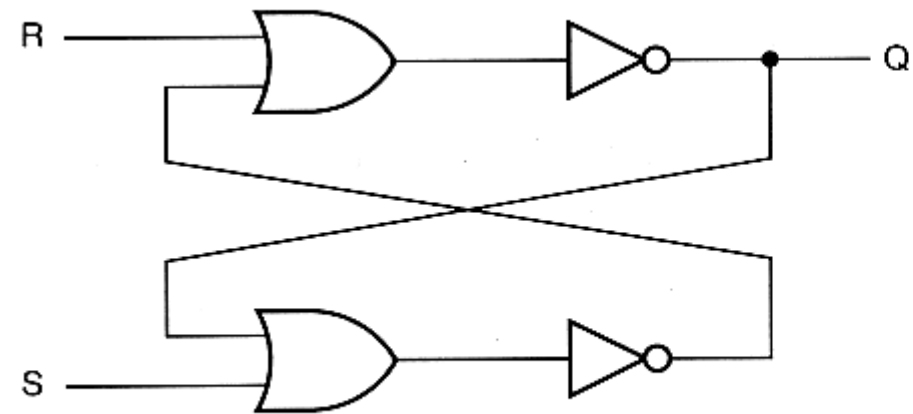
A finite state machine (FSM) is a model of behavior composed of a finite number of states, transitions between those states, and actions.

# Combinatorial and Sequential Circuits: Definitions

*A combinatorial circuit* is a circuit whose output is solely determined by its input values.

*A sequential circuit* is a circuit whose output is a function of input values **and** the current state of the circuit.

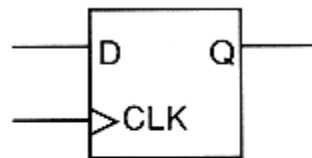
# Circuits as Memory



S	R	Q
0	0	last Q
0	1	0
1	0	1
1	1	illegal

A simple S-R flip-flop: (a) circuit; (b) symbol; (c) function table.

# Circuits as Memory (cont'd)



(a)

CLK	D	Q
	0	0
	1	1
0	x	last Q
1	x	last Q

(b)

---

Positive edge-triggered D flip-flop: (a) symbol; (b) function table.

*D flip-flops* are grouped together into *registers* to store multi-bit quantities in a computer.