Homework

- Reading
  - Tokheim, Chapter 3, 4, and 6.1 - 6.3
  - Free Digital Logic Simulator website
- Machine Projects
  - Continue on mp3
- Labs
  - Continue in labs with your assigned section

Digital Logic

- Two types of digital logic design
  - Combinational Logic – has no memory elements
  - Sequential Logic – contains memory elements
- Combinational logic design is sometimes called Boolean algebra after George Boole
  - Based on binary logic
  - Uses AND, OR, XOR, NOT, etc.
  - Develop truth tables and implement a design

Boolean Algebra

- Variables / Expressions

  Two Values Only (0 or 1)

- Basic Operators

  AND *

  OR +

  XOR ⊕

  NOT Bar over the variable or expression

  Or # after variable name

Boolean Algebra

- Precedence of Operators

NOT	(like unary minus)
AND	(like multiply)
Division	None
OR/XOR	(like add)
Subtraction	None

- Parentheses to force precedence

  A * (B + C) is not the same as A * B + C

Boolean Algebra

- Multiplicative Identities

  $A * 0 = 0$

  $A * \overline{A} = A$ (Note: Not A squared)

  $A * A = 0$

- Additive Identities

  $A + 1 = 1$

  $A + \overline{A} = A$ (Note: Not 2A)

  $A + A = 1$

Boolean Algebra

- Negative Identity

  $\overline{\overline{A}} = A$

- Commutative Property

  $A * B = B * A$

  $A + B = B + A$

- Distributive Property

  $A (B + C) = A * B + A * C$

Boolean Algebra

- Common Reductions of Sums of Products (Also called a “Minterm” expression)

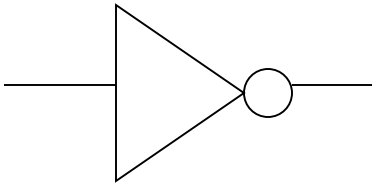
$$A * \overline{B} + \overline{A} * B + A * B = A + B \qquad \text{OR}$$

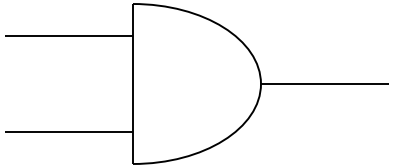
$$A * \overline{B} + \overline{A} * B = A \oplus B \qquad \text{XOR}$$

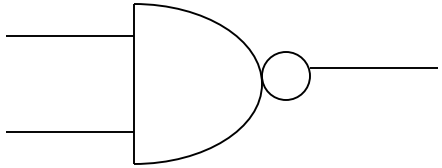
- Common Reductions of Products of Sums (Also called a “Maxterm” expression)

$$(A + B) * (\overline{A} + B) * (A + \overline{B}) = A * B \qquad \text{AND}$$

Binary Logic Symbols/Tables





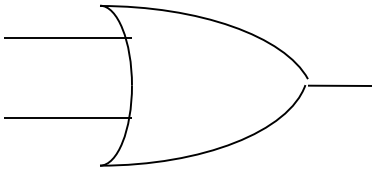


NOT	
0	1
1	0

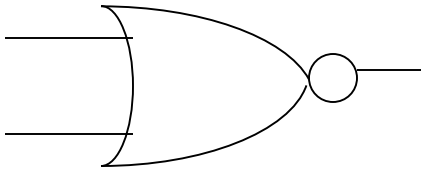
AND	0	1
0	0	0
1	0	1

NAND	0	1
0	1	1
1	1	0

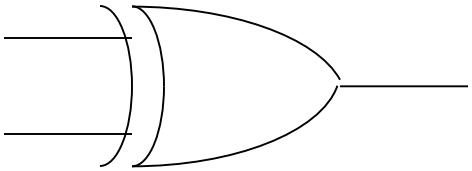
Binary Logic Symbols/Tables



OR	0	1
0	0	1
1	1	1



NOR	0	1
0	1	0
1	0	0



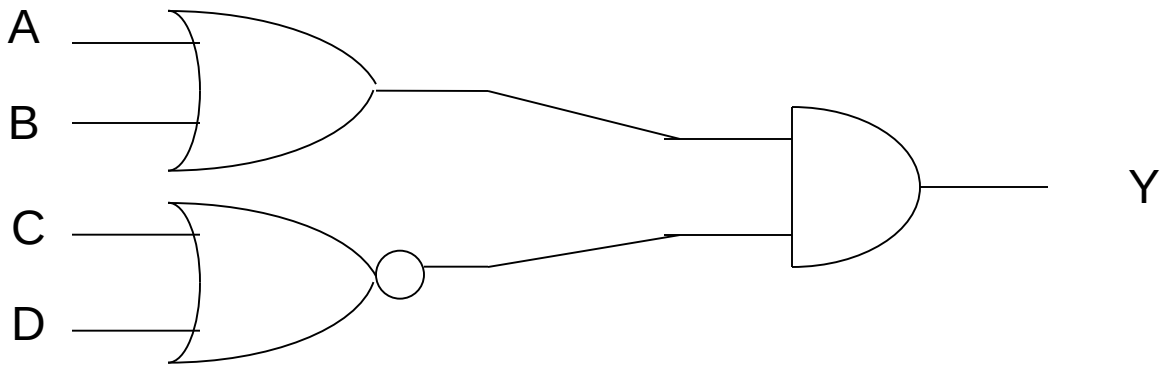
XOR	0	1
0	0	1
1	1	0

Boolean Algebra / Logic Diagram

- Sample Boolean Equation

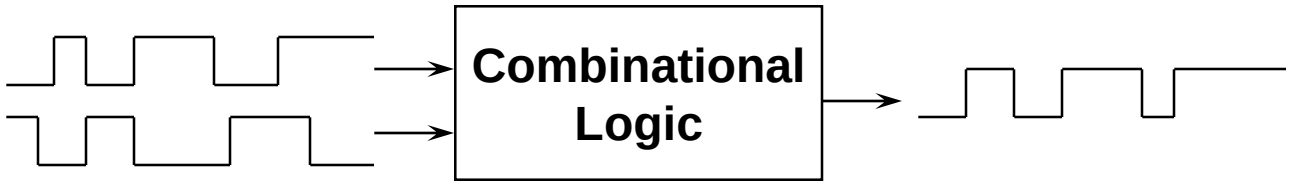
  $Y = (A + B) * \overline{(C + D)}$

- Equivalent Logic Diagram



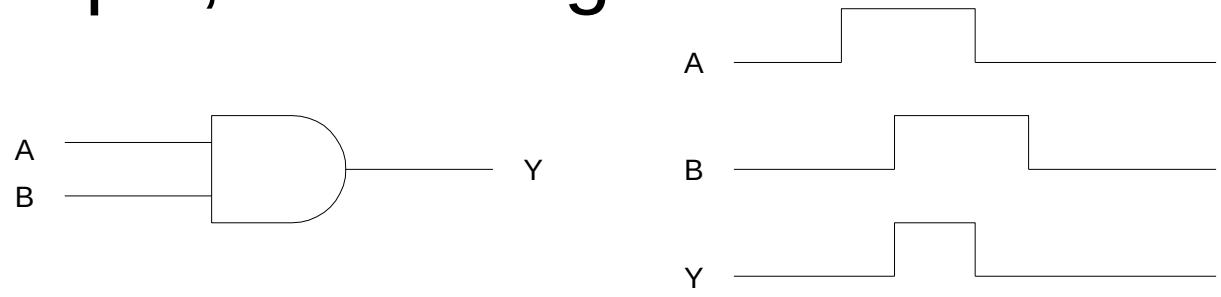
Timing Considerations

- Can look at pulse trains through combinational logic over time rather than just constant inputs



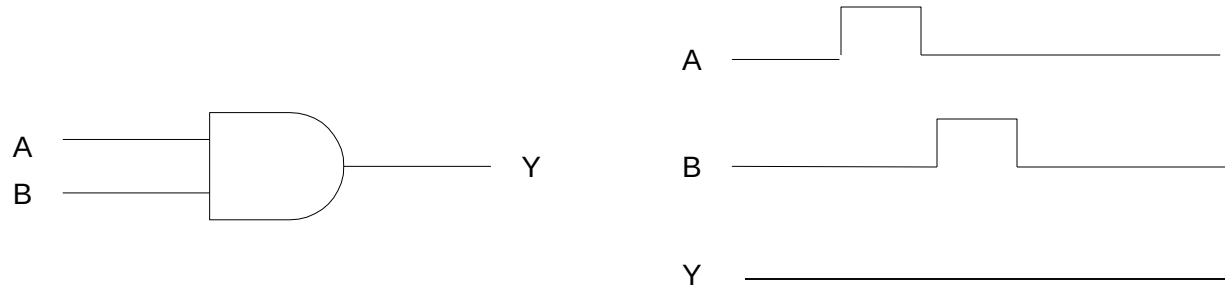
Timing Considerations

- Things don't happen instantaneously
  - Each signal arrives at some unique time
  - Gate logic takes some time to react
  - Changes in the output appear some time after changes to the inputs (at nanosecond level)
- Example, an AND gate



Timing Considerations

- Things don't happen instantaneously
  - As clock speed increases, the "skew" due to differential delays narrows output pulses
  - Beyond the designed maximum clock speed, the circuit may fail
- Example, an AND gate

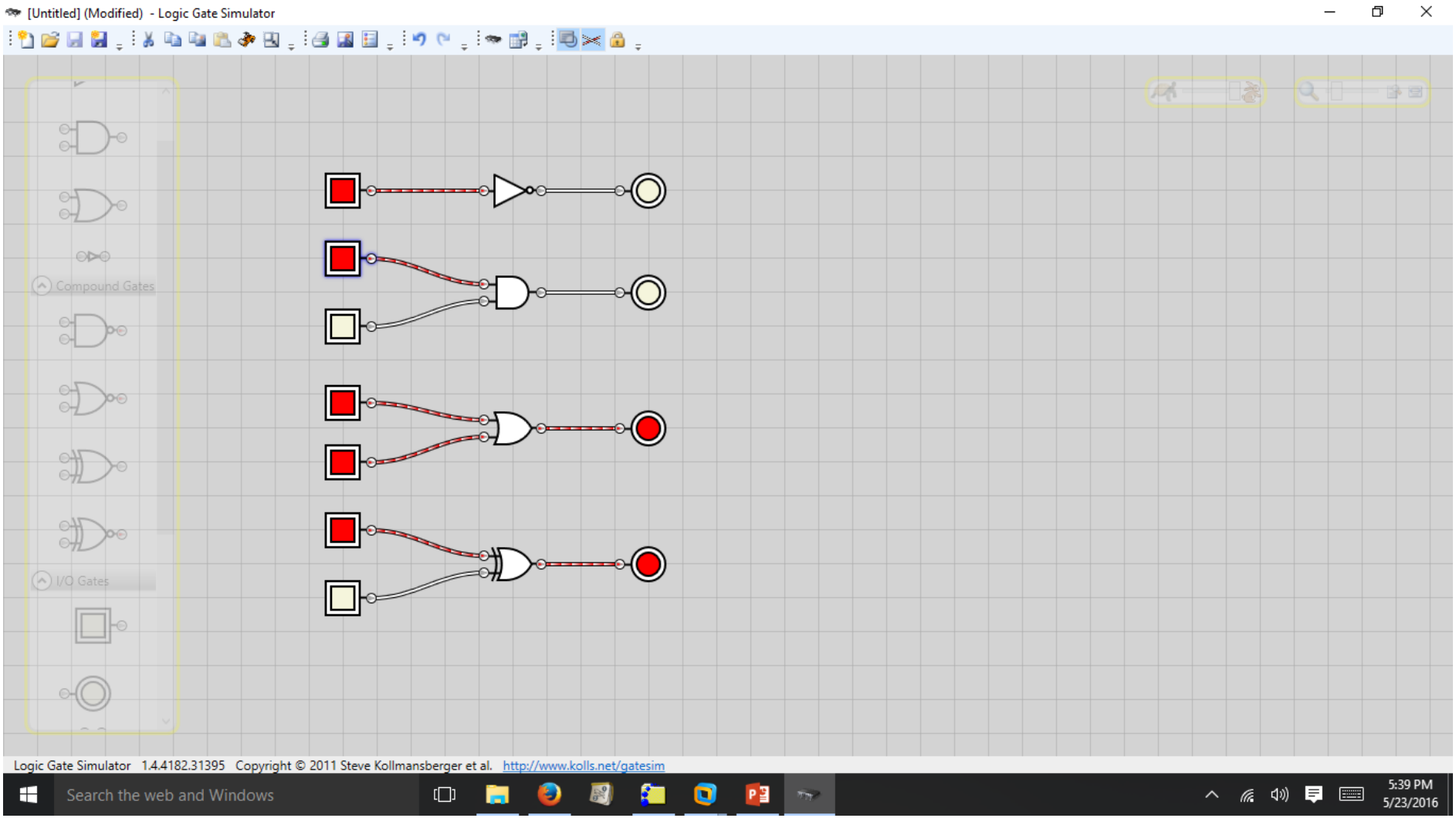


Free Logic Design Tool

- Practical software design tools such as VHDL are now being used in industry instead of building and testing physical “breadboards”
- VHDL is complex and has a learning curve
- Logic Gate Simulator is a simple design tool
- Allows us to design and test simple hardware logic via software building and simulation
- Download:

  https://www.kolls.net/gatesim/

Free Logic Design Tool



Free Logic Design Tool

- Demonstration in Class
  - Button
  - Indicator
  - Inverter
  - And Gate
  - Or Gate
  - Exclusive-Or Gate