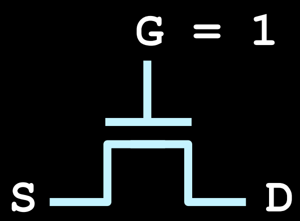
CSC258: Computer Organization

Circuit Creation

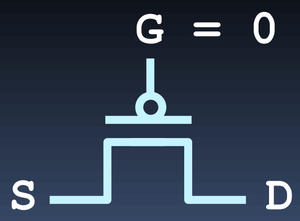
*Based on slides originally created by Prof. Steve Engels

Review - Transistors

- Transistors are switches whose behavior can be controlled electrically
- nMOS conducts electricity between source and drain when positive voltage (5V) is applied to the gate



- pMOS conducts electricity between source and drain when ground (0 V) is applied to the gate

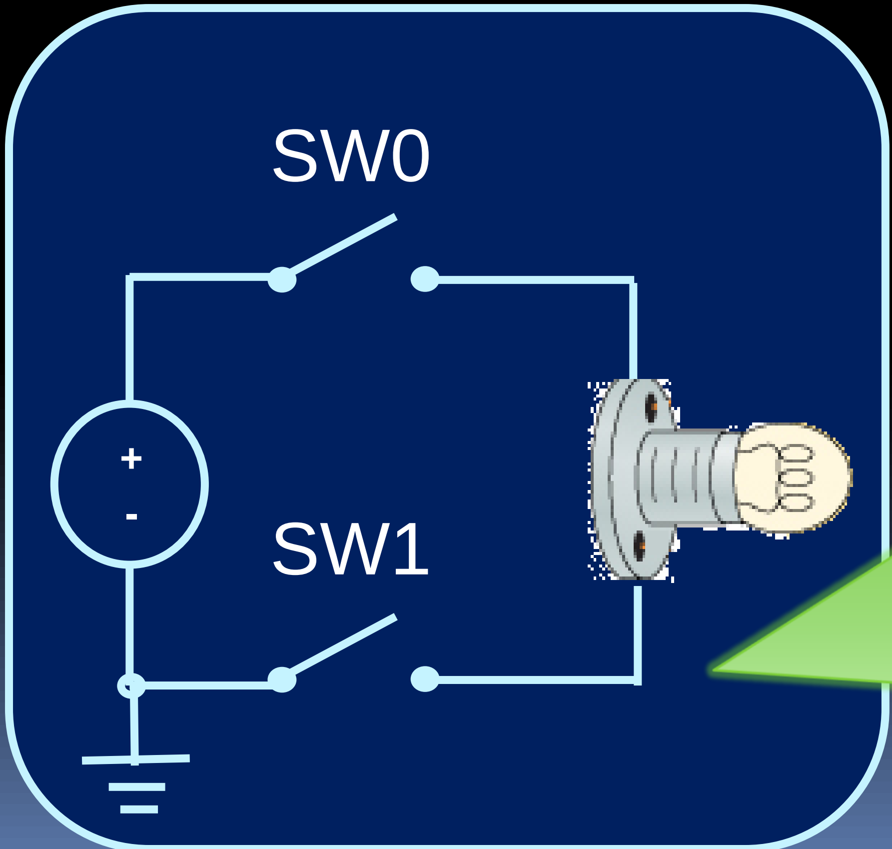


Review – Logic Values

- Logic values are about potential
  - When a point in circuit is at 5V potential, we call that high value or 1
    - The point is at 5V potential, if there is a direct path (no resistance) from Vcc to this point in the circuit
  - When a point in circuit is at 0V potential, we call that low value or 0
    - The point is at 0V potential, if there is a direct path (no resistance) from GND to this point in the circuit
- A point in the circuit which does not have a path to either VCC or GND is at an undefined potential

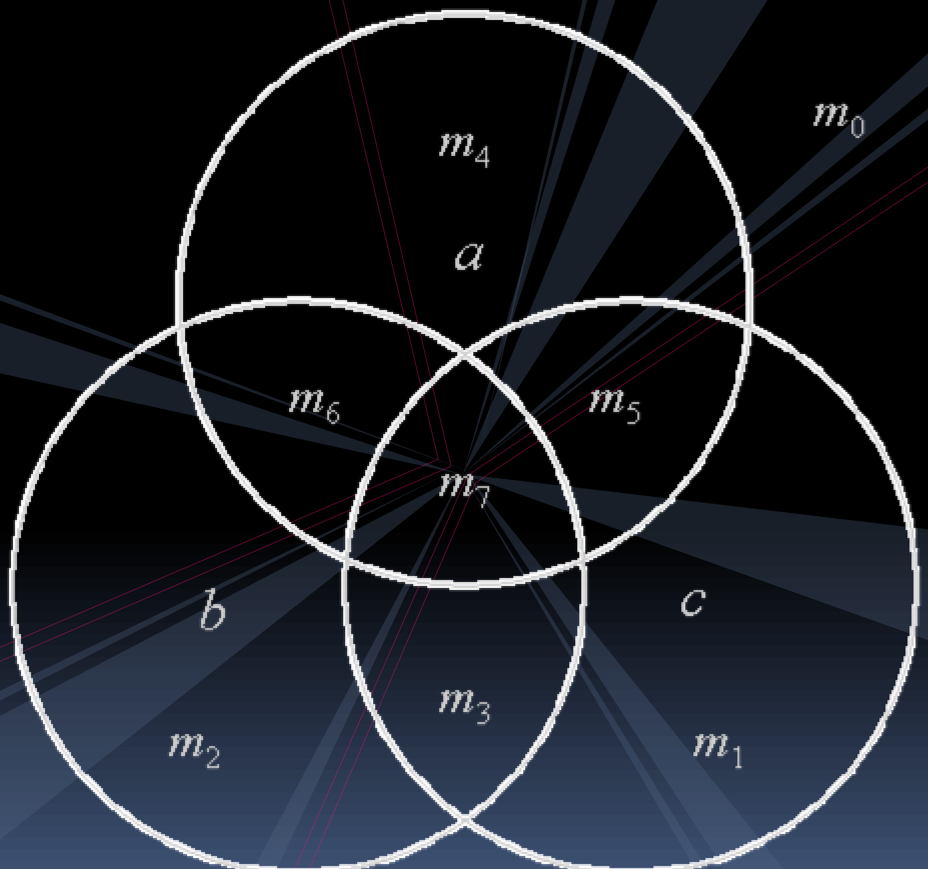
Review: Disconnected ≠ 0

- Easy to demonstrate that disconnected points are not the same as 0



This point in the circuit is disconnected from both Vcc or GND.
If disconnected was the same as 0, then closing switch SW0 would cause the light to come on, because there would be a difference in potential between the light's two contacts!

Minterms



Example truth table

- Consider the following example:
  - *"Given three inputs A, B, and C, make output Y high whenever any of the inputs are low, except when all three are low or when A and C are high."*
- This leads to the truth table on the right.
  - Is there a better way to describe the cases when the circuit's output is high?

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

Minterms

- An easier way to express logic function is to assume a standard truth table format, then list which rows cause high output.
- Each row of the truth table represents one minterm.

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

Minterm	Y
m_0	0
m_1	1
m_2	1
m_3	1
m_4	1
m_5	0
m_6	1
m_7	0

Formally speaking

- A more formal description:
  - Minterm = an AND expression with every input present in true or complemented form.
  - For example, given four inputs (A, B, C, D):
    - Valid minterms:

$$A \cdot \overline{B} \cdot C \cdot D \qquad \overline{A} \cdot B \cdot \overline{C} \cdot D \qquad A \cdot B \cdot C \cdot D$$

    - Not minterms:

$$A \cdot B + C \cdot D \qquad A + B + C + \overline{D} \qquad A \cdot B \cdot D$$

Numbering minterms

- Circuits are often described using minterms, as a form of logic shorthand.
  - Given n inputs, there are 2^n minterms possible (same as rows in a truth table).
  - Naming scheme:
    - Minterms are labeled as m_x
      - The x subscript indicates the row in the truth table.
      - x starts at 0 (when all inputs are low), and ends with $n-1$.
      - x corresponds to the binary number formed by the variables in the truth table, i.e. 000 $\rightarrow m_0$, 001 $\rightarrow m_1$,…, 111 $\rightarrow m_7$
  - Example: Given 3 inputs
    - Minterms are m_0 $(\overline{A} \cdot \overline{B} \cdot \overline{C})$ to m_7 $(A \cdot B \cdot C)$

Using minterms

- What are minterms used for?
  - A single minterm indicates a set of inputs that will make the output go high.
  - Example: m_2
    - Output only goes high in third line of truth table.

A	B	C	D	m_2
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Using minterms

- What happens when you OR two minterms?
  - Result is output that goes ________________ ______________________
  - For $m_2 + m_8$, both third and ninth lines of truth table result in high output.

A	B	C	D	m_2	m_8	m_2+m_8
0	0	0	0	0	0	
0	0	0	1	0	0	
0	0	1	0	1	0	
0	0	1	1	0	0	
0	1	0	0	0	0	
0	1	0	1	0	0	
0	1	1	0	0	0	
0	1	1	1	0	0	
1	0	0	0	0	1	
1	0	0	1	0	0	
1	0	1	0	0	0	
1	0	1	1	0	0	
1	1	0	0	0	0	
1	1	0	1	0	0	
1	1	1	0	0	0	
1	1	1	1	0	0	

Creating Boolean expressions

- Sum-of-Minterms (SOM):
  - Since each minterm corresponds to a single high output in the truth table, the combined high outputs are a union of these minterm expressions.
  - Also known as ________________.

$Y = m_2 + m_6 + m_7 + m_{10}$ (SOM)

A	B	C	D	m_2	m_6	m_7	m_{10}	Y
0	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	0	0	0					
1	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					

Maxterms

- An alternative to minterms
  - Express logic function by specifying which rows cause low output.
  - Numbering of maxterms is the same as numbering of minterms (with a caveat)

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

↔

Maxterm	Y
M_0	0
M_1	1
M_2	1
M_3	1
M_4	1
M_5	0
M_6	1
M_7	0

Formally speaking

- A more formal description:
  - Maxterm = an OR expression with every input present in true or complemented form.
  - For example, given four inputs (A, B, C, D):
    - Valid maxterms:

$$A+\overline{B}+C+D \qquad \overline{A}+B+\overline{C}+D \qquad A+B+C+D$$

    - Not maxterms:

$$A\cdot B+C\cdot D \qquad A\cdot B\cdot C\cdot \overline{D} \qquad A+B+D$$

Numbering maxterms

- Similar to minterms
  - Given n inputs, there are 2^n maxterms possible (same as rows in a truth table).
  - Maxterms are labeled as M_x
    - The x subscript indicates the row in the truth table.
      - x starts at 0 (when all inputs are low), and ends with $n-1$.
      - x corresponds to the binary number formed by the variables in the truth table, i.e. 000 $\rightarrow M_0$, 001 $\rightarrow M_1$,…, 111 $\rightarrow M_7$
  - Example: Given 3 inputs
    - Maxterms are M_0 $(A+B+C)$ to M_7 $(\overline{A}+\overline{B}+\overline{C})$

Using maxterms

- A single maxterm indicates a set of inputs that will make the output go low.
  - Example: M_2
    - Output only goes low in the third line of truth table.
  - M_2 = $A+B+\overline{C}+D$

A	B	C	D	M_2
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Using maxterms

- What happens when you AND two maxterms?
  - Result is output that goes low in both maxterm cases.
  - For $M_2 \cdot M_8$, both third and ninth lines of truth table result in low output.

A	B	C	D	M_2	M_8	$M_2 \cdot M_8$
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	1	0	0	1	0
0	0	1	1	1	1	1
0	1	0	0	1	1	1
0	1	0	1	1	1	1
0	1	1	0	1	1	1
0	1	1	1	1	1	1
1	0	0	0	1	0	0
1	0	0	1	1	1	1
1	0	1	0	1	1	1
1	0	1	1	1	1	1
1	1	0	0	1	1	1
1	1	0	1	1	1	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1

$$Y = M_3 \cdot M_5 \cdot M_7 \cdot M_{10} \cdot M_{14} \text{ (POM)}$$

A	B	C	D	M_3	M_5	M_7	M_{10}	M_{14}	Y
0	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	0	0	0						
1	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						

Creating Boolean expressions

- Two canonical forms of Boolean expressions:
  - Sum-of-Minterms (SOM):
    - Since each minterm corresponds to a single high output in the truth table, the combined high outputs are a _____________ of these minterm expressions.
    - Also known as: Sum-of-Products.
  - Product-of-Maxterms (POM):
    - Since each maxterm only produces a single low output in the truth table, the combined low outputs are an ____________ of these maxterm expressions.
    - Also known as Product-of-Sums.

Using Minterms and Maxterms

- Using minterms or maxterms, logic functions can be expressed in a more compact way than showing entire truth tables
- Only works if ordering of columns in the truth table is agreed upon
- Sum-of-minterms are useful in cases ______

  ____

  ____
- Product-of-maxterms useful when expressing truth tables that have ________________

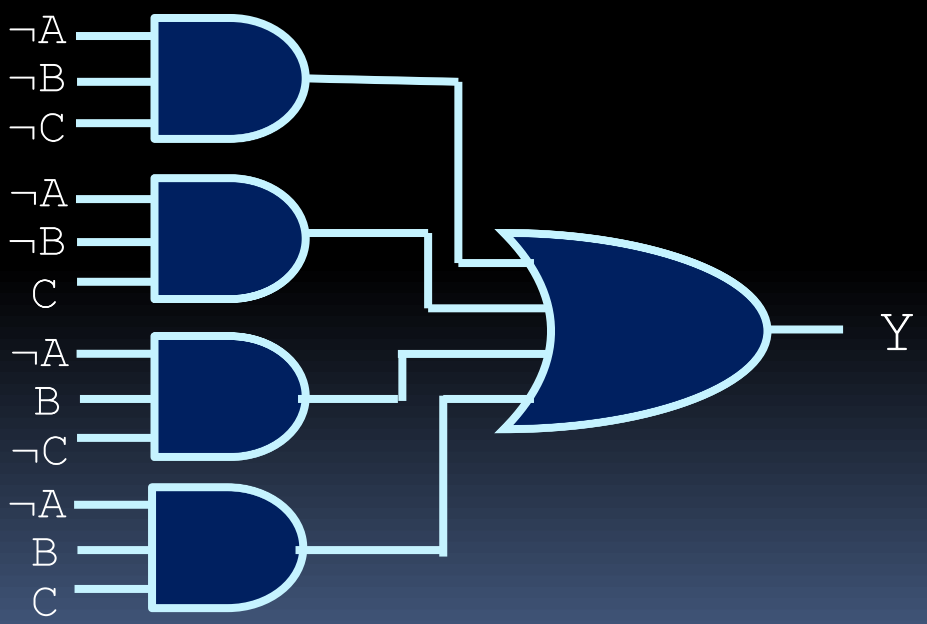
  ____

Converting SOM to gates

- Once you have a Sum-of-Minterms expression, it is easy to convert this to the equivalent combination of gates:

$$m_0 + m_1 + m_2 + m_3 =$$

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



Quick Exercises

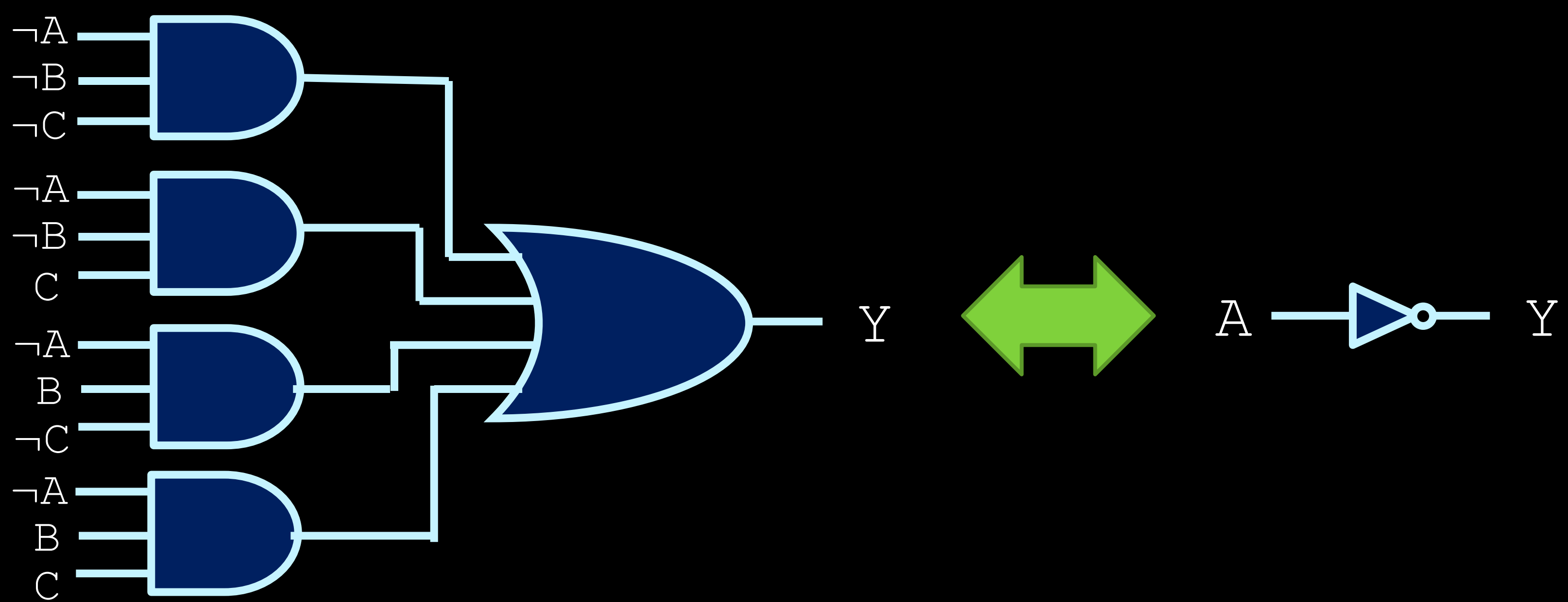
- Given 4 inputs A, B, C and D write:
  - m9 =>
  - m15 =>
  - m16 =>
  - M2 =>

- Which minterm is this one?
  - $A' \cdot B \cdot C' \cdot D'$ [redacted]

- Which maxterm is this one?
  - $A + B + C + D'$ [redacted]

Reducing circuits

Reasons for reducing circuits



- Note example of Sum-of-Minterms circuit design.
- To minimize the number of gates, we want to reduce the Boolean expression as much as possible from a collection of minterms to something smaller.
- This is where CSC165 skills come in handy ☺

Boolean algebra review

- Axioms:

$$0 \cdot 0 = 0 \qquad 0 \cdot 1 = 1 \cdot 0 = 0$$
$$1 \cdot 1 = 1 \qquad \text{if } x = 1,\ \overline{x} = 0$$

- From this, we can extrapolate:

$$x \cdot 0 = \qquad x+1 =$$
$$x \cdot 1 = \qquad x+0 =$$
$$x \cdot x = \qquad x+x =$$
$$x \cdot \overline{x} = \qquad x+\overline{x} =$$
$$\overline{\overline{x}} =$$

If one input of a 2-input AND gate is 1, then the output is whatever value the other input is.

If one input of a 2-input OR gate is 0, then the output is whatever value the other input is.

Other Boolean identities

- Commutative Law:

$$x \cdot y = y \cdot x \qquad x+y = y+x$$

- Associative Law:

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

- Distributive Law:

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

This part is non-intuitive but true

Other Boolean identities

- Simplification Law:

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

- Consensus Law:

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

  - Proof by Venn diagram:

Consensus Law Proof -Venn diagram

- Consensus Law:

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

- Proof by Venn diagram:
  - $x \cdot y$
  - $\overline{x} \cdot z$
  - $y \cdot z$
    - Already covered!

Proving Boolean identities

- The most straightforward way to prove a Boolean identity is to write the truth table for left and right side, and show that they produce the same function
- Exercise:
  - Prove the “non-intuitive” distributive law
    - $x+(y\cdot z) = (x+y)\cdot(x+z)$

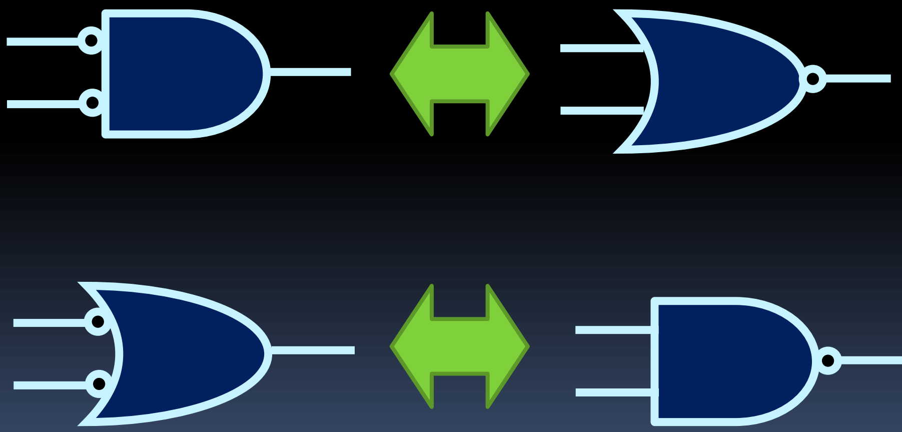
Other Boolean identities

- Absorption Law:

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

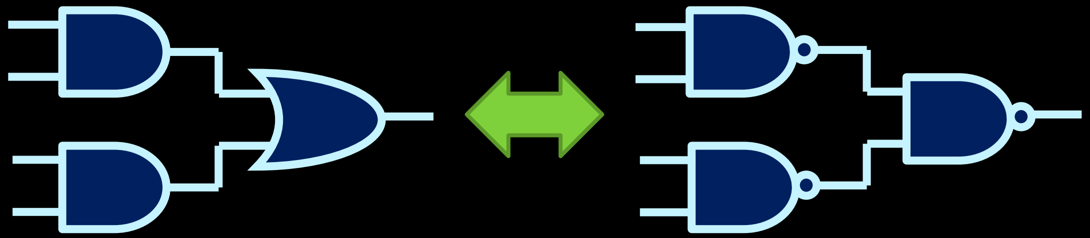
- De Morgan's Laws:

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

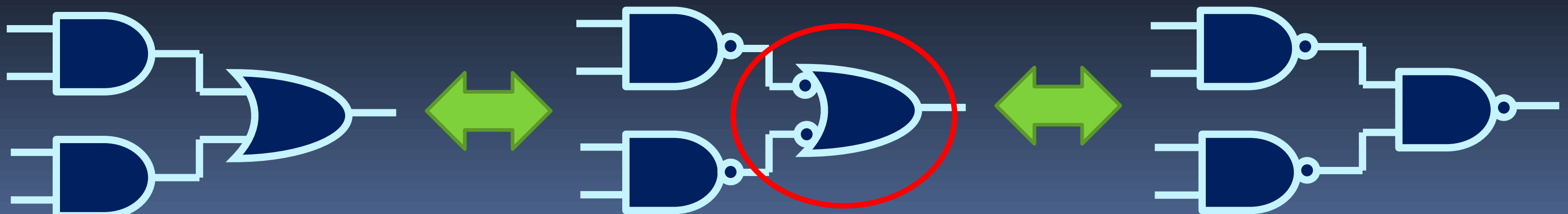


Converting to NAND gates

- De Morgan's Law is important because out of all two-input gates, NANDs are the cheapest to fabricate.
  - A Sum-of-Products circuit could be converted into an equivalent circuit of NAND gates:



- This is all based on de Morgan's Law:



Reducing Boolean expressions

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

- Assuming function on the left, we get the following:

$$Y = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C} + A \cdot B \cdot \bar{C} + A \cdot B \cdot C$$

- Now start combining terms, like the last two:

$$Y = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot \bar{C} \mathbf{+ A \cdot B}$$

Reducing Boolean expressions

- Different final expressions possible, depending on what terms you combine.
- For instance, given the previous example:

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

- If you combine the end and middle terms…

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

- Which reduces the number of gates and inputs!