Warmup

What is 0101 0001 + 0000 1101?

Respond at **pollev.com/stevenbell**

ES 4: Boolean equations and logic gates

Steven Bell

13 September 2021

Tufts UNIVERSITY

By the end of class today, you should be able to:

- Write truth tables for AND, OR, XOR, NOT, NAND, NOR, XNOR
- Write a boolean equation from an English description and vice-versa
- Draw a logic diagram from a boolean equation and vice-versa
- Write a truth table from a boolean equation and vice-versa

Key representations

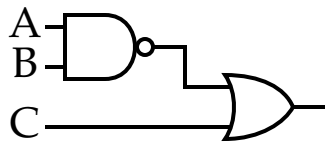
Boolean equation

$$\overline{AB} + C$$

Truth table

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

Logic diagram



Key representations

English description

Not both apples and bananas...

Code

```
!(A&&B) || C
```

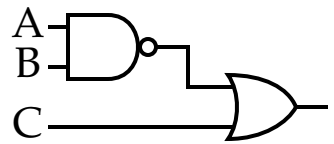
Boolean equation

$$\overline{AB} + C$$

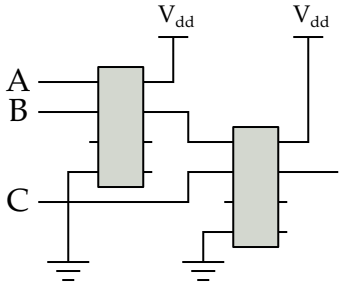
Truth table

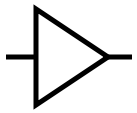
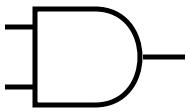
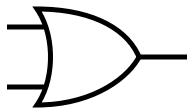
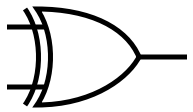
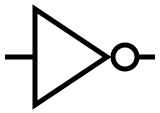
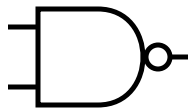
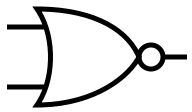
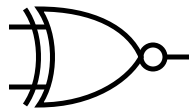
A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1

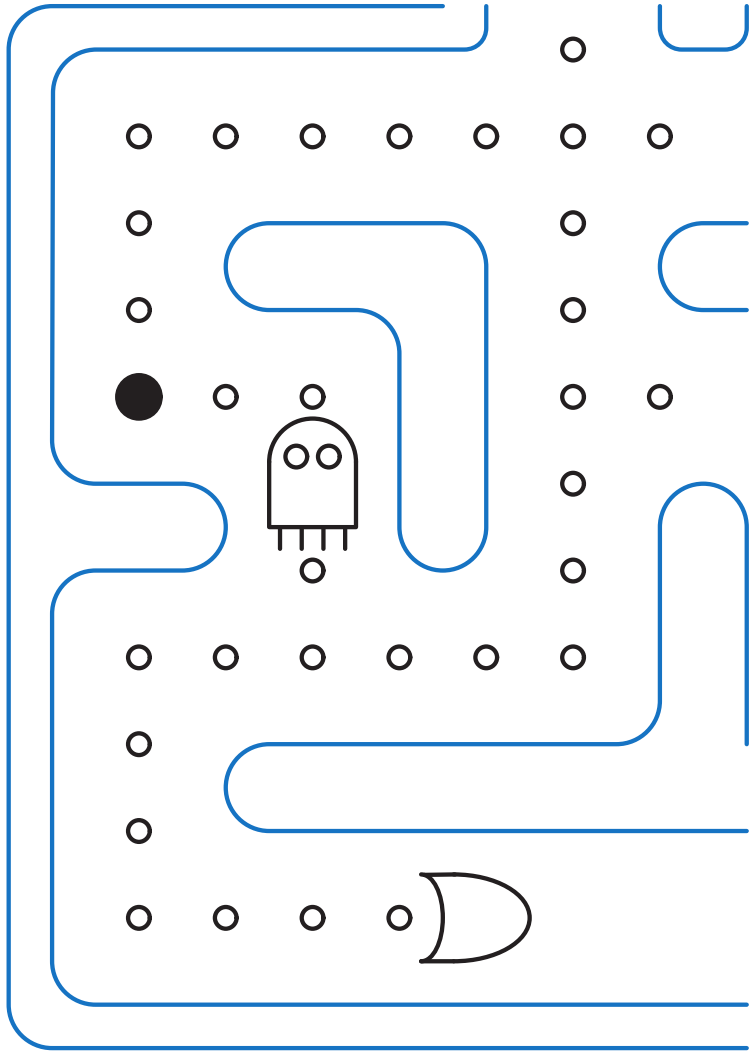
Logic diagram



Circuit diagram



Name	Buffer	AND	OR	XOR	Inverter	NAND	NOR	XNOR
Logic symbol								
Logic equation	A	AB	$A+B$	$A \oplus B$	$\overline{A}$	$\overline{AB}$	$\overline{A+B}$	$\overline{A \oplus B}$
Truth table	A | Y 0 | 0 **1 | 1**	A B | Y 0 0 | 0 0 1 | 0 1 0 | 0 **1 1 | 1**	A B | Y 0 0 | 0 **0 1 | 1** **1 0 | 1** **1 1 | 1**	A B | Y 0 0 | 0 **0 1 | 1** **1 0 | 1** 1 1 | 0	A | Y **0 | 1** 1 | 0	A B | Y **0 0 | 1** **0 1 | 1** **1 0 | 1** 1 1 | 0	A B | Y **0 0 | 1** 0 1 | 0 1 0 | 0 1 1 | 0	A B | Y **0 0 | 1** 0 1 | 0 1 0 | 0 **1 1 | 1**
C++ equivalent	`A`	`A&&B`	`A||B`	`A!=B`	`!A`	`!(A&&B)`	`!(A||B)`	`A==B`
74-series IC	7407	7408	7432	7486	7404	7400	7402	various



Pacman is like an OR gate...?


STAR TREK


Y
STOR TREK
A
B

Bigger equations

We'll use a **bar** to indicate inversion: $\overline{A+B}$

You might also see "¬" for inversion: $\neg(A+B)$

AND takes precedence over OR:

$$AB + C = (AB) + C$$

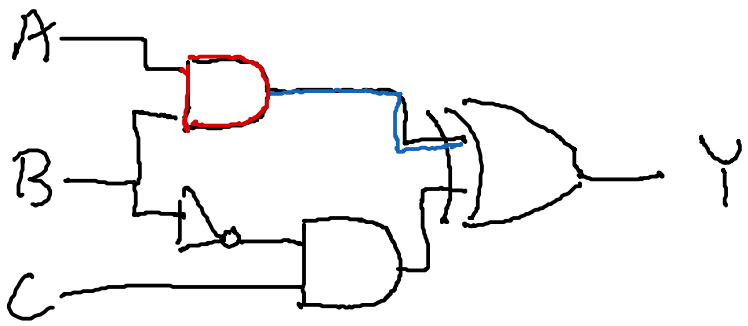
(Think multiplication over addition)

Drawing logic diagrams

Elements are objects

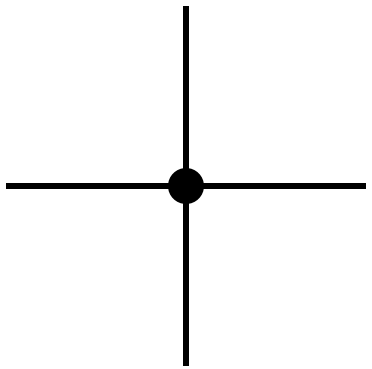
Nodes are the interconnections

$$Y = (AB) \oplus (\overline{B}C)$$

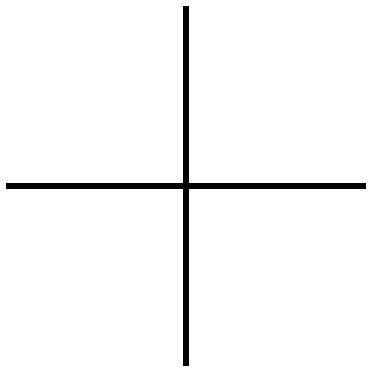


Logic diagram connections

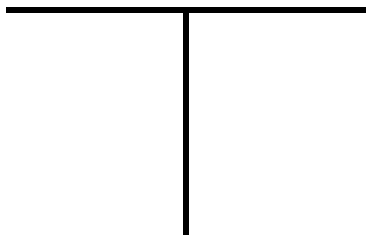
Wires are connected if there's a dot:



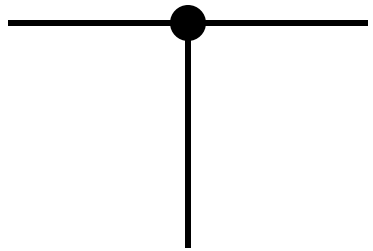
connected



not connected



connected



connected
(dot is only for clarity)

What is combinational?

A circuit is combinational if:

1. It is a discrete logic gate, or
2. It is composed of combinational elements such that:

    There are no cycles, and

    Every node (wire) is only driven by one gate

Who cares, anyway?

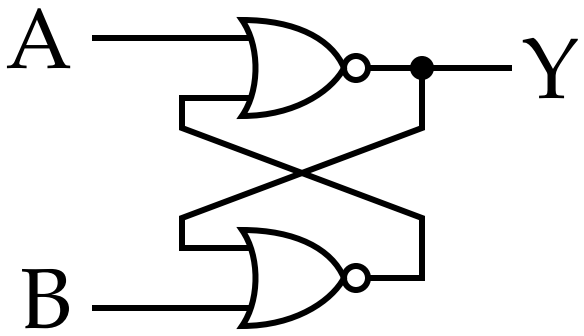
In a combinational circuit,
outputs are a function of only the current inputs

Feedback loops can break this!

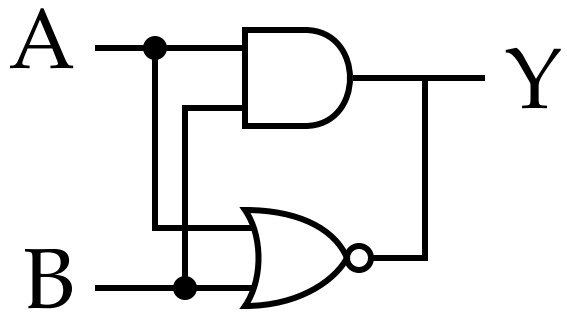
Driving one wire with multiple gates breaks the digital abstraction!

Which of the circuits below are combinational?

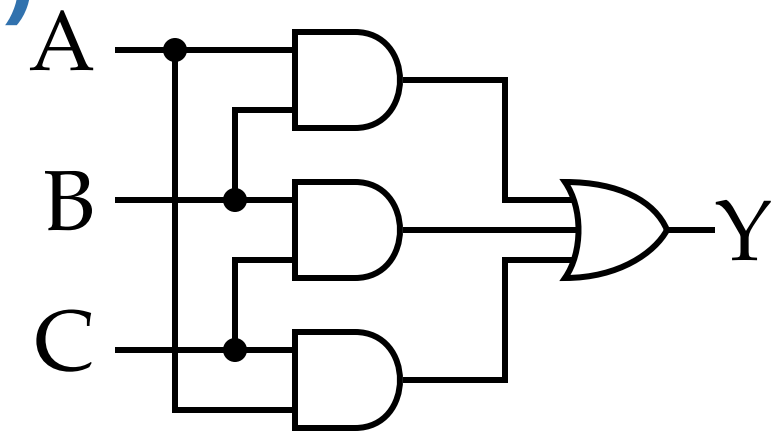
(1)



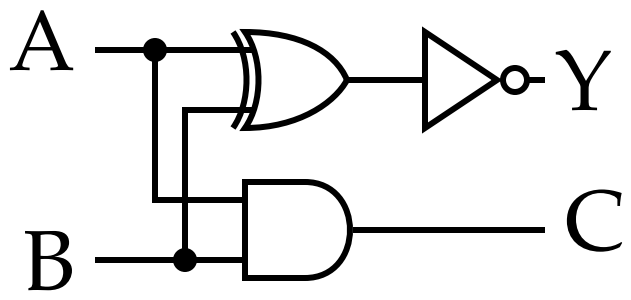
(2)



(3)



(4)



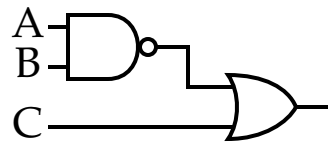
Respond at **pollev.com/stevenbell**

Translating representations

Boolean equation

$$\overline{AB} + C$$

Logic diagram



Truth table

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

Boolean equation ↔ logic diagram

Replace operations with gates, following precedence

Replace gates with operations, nesting to create precedence

Translating representations

Boolean equation

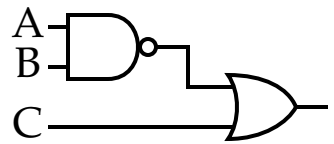
$$\overline{AB} + C$$

Evaluate

Truth table

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

Logic diagram



Translating representations

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

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

Boolean equation

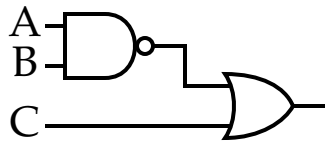
$\overline{AB} + C$

Evaluate

Truth table

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

Logic diagram



Translating representations

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

Boolean equation

$\overline{AB} + C$

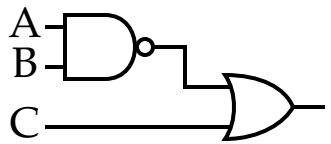
Evaluate

Standard forms

Truth table

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

Logic diagram



Translating representations

English description

Not both apples and bananas...

Self-explanatory (hopefully!)

Code

```
!(A&&B) || C
```

Boolean equation

$\overline{AB} + C$

Evaluate

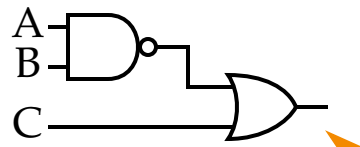
Standard forms

Truth table

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

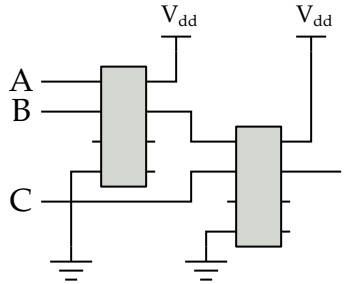
Next week!

Logic diagram



Prelab #2!

Circuit diagram



For Wednesday

1. Read the book (2.3-2.7) and complete the pre-class reading check

   www.ece.tufts.edu/es/4/

   Quiz is due at **10 AM** the day of class, so I can review it

   ```
   cp /es/4/public_html/readingchecks/readingcheck_02.txt ./
   ```

   ```
   provide es4 rc2 readingcheck_02.txt
   ```

2. Come to your lab section!

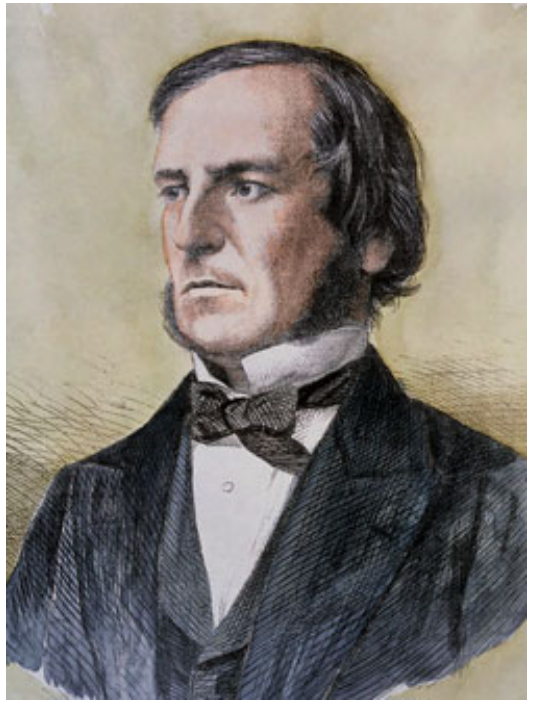
Bonus material

Backup slides, extra information, or just stuff
I had to cut out of the lecture for the sake of time.

A useful circuit

We have two 2-bit numbers. Design a circuit to calculate A > B.

Boolean or boolean?



George Boole
(1815-1864)

(Photo from Wikimedia)

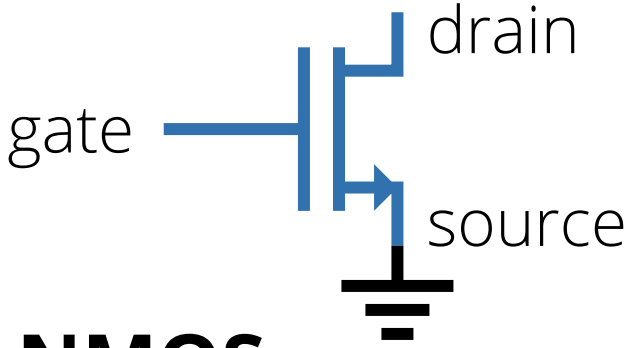
Why is NAND a big deal?

For the purposes of this course,
MOS transistors are electrically-controlled **switches**.

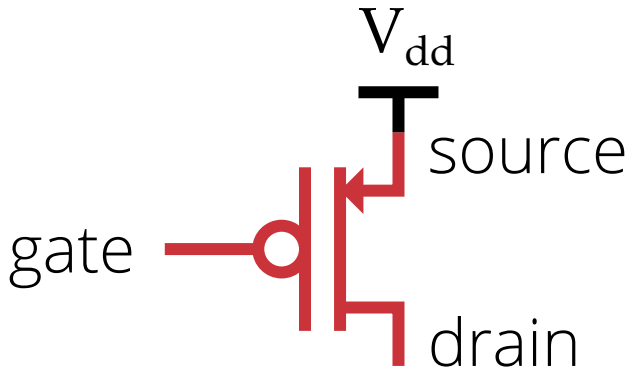
They have one terminal that controls the "switch", called the **gate**.

When the transistor is on, current can flow between the other terminals, called the **source** and **drain**.

They come in two types:

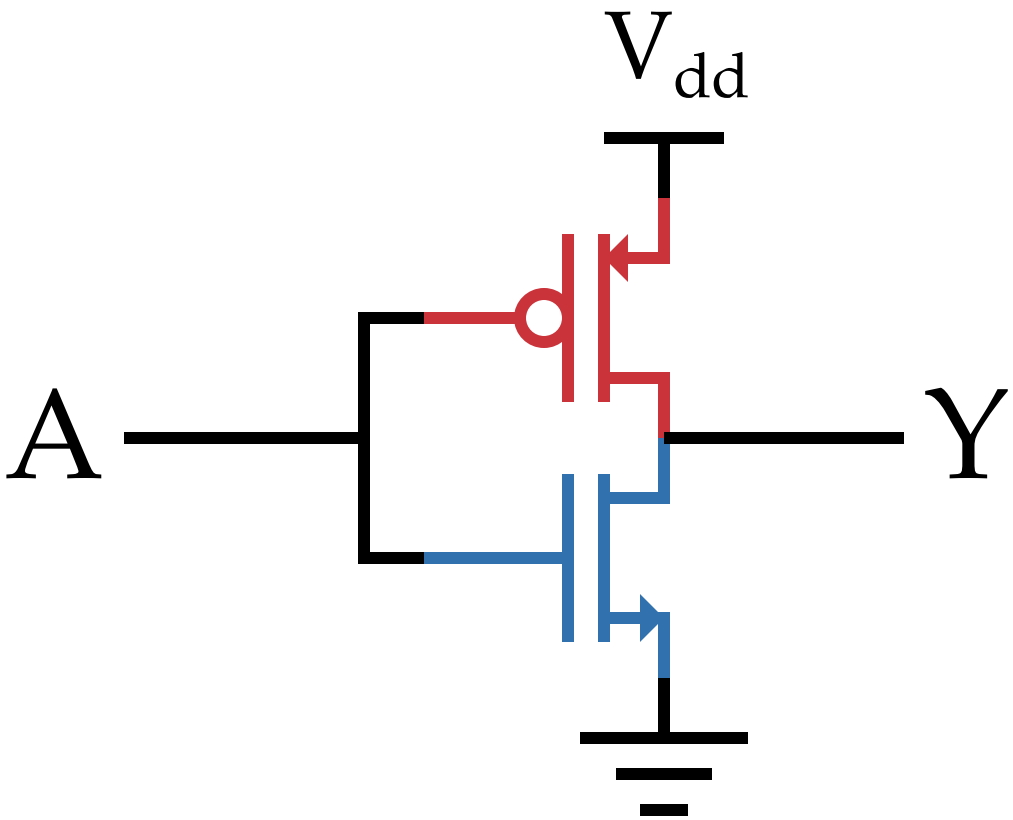


NMOS



PMOS

A simple gate



PMOS: "closed" when gate is **low**.

NMOS: "closed" when gate is **high**.

A more complex gate