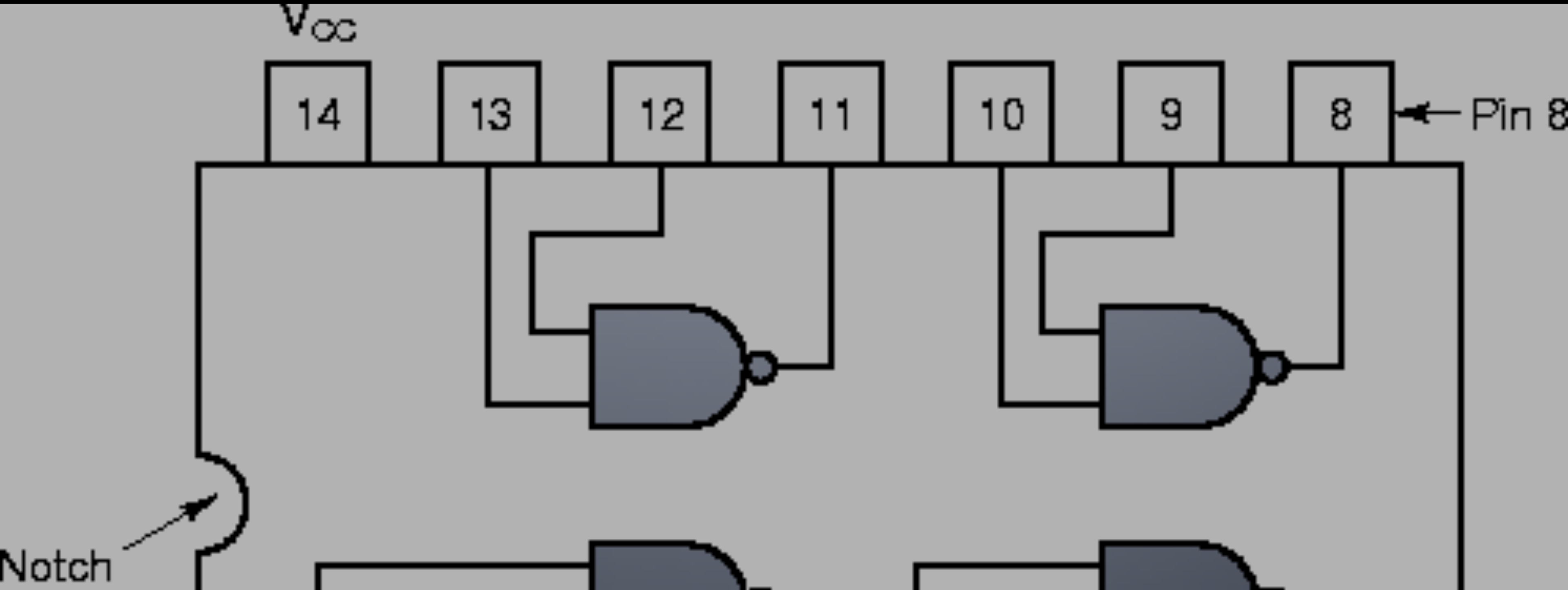
E11 - Autonomous Vehicles

Digital Systems

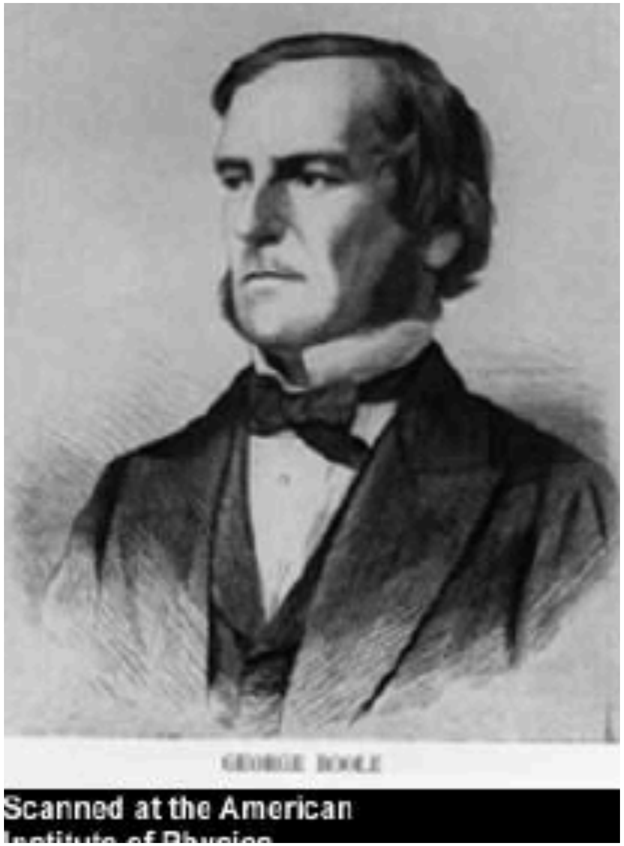


Outline

- Zeros and Ones
- Binary Numbers
- Boolean Logic

Boolean Logic

- **Charles Boole**
  - Born to working class parents
  - Taught himself mathematics and joined the faculty of Queen's College in Ireland.
  - Wrote *An Investigation of the Laws of Thought (1854)*



Boolean Logic

- **Charles Boole**
  - Introduced binary variables
  - Introduced the three fundamental logic operations: AND, OR, and NOT.

Zeros & Ones

- **Most digital systems today use voltage to process 0 and 1**

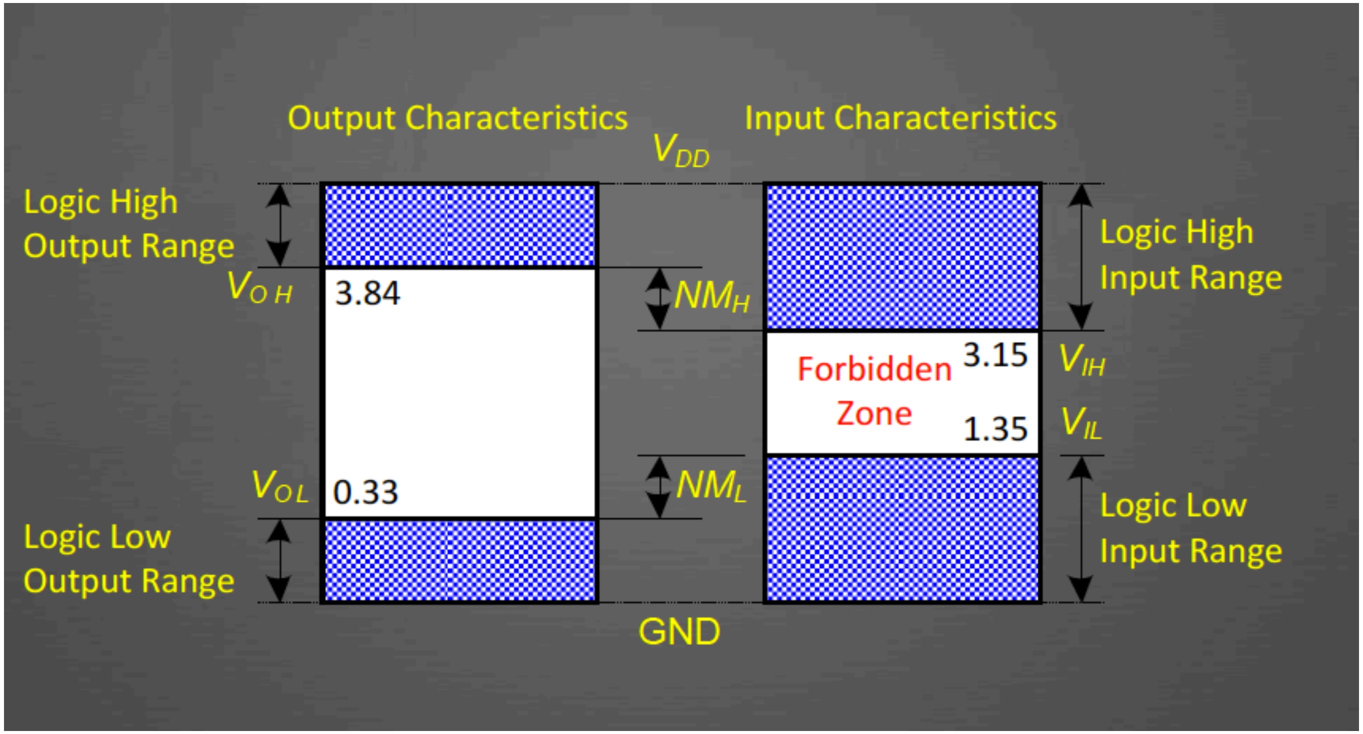
  0 = low voltage

  1 = high voltage

- **Power supply voltage: VDD (or VCC)**
  - Ground = 0 V
  - We'll use VDD = 5 V

Zeros & Ones

- **Digital Signals…**



Zeros & Ones

- **Bits**
  - Take on one of two states, **0** or **1**
  - Also called Binary digits
  - If there are N bits,

    we can represent ____ different states

Binary Numbers

- **Binary Numbers**
  - Use many bits to represent a number with value greater than 1
  - For 4 bits, the smallest value is represented by

    0000

    The largest value is represented by

    1111

Binary Numbers

- **Least Significant Bit**
  - The rightmost bit in a binary number

    000<u>0</u>

  - Represents a 0 or 1
- **Most Significant Bit**
  - The leftmost bit in a binary number

    <u>0</u>000

  - Represents a 0 or 2^{N-1}

Binary Numbers

- **Decimal Numbers**

$$5374_{10} = 5 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 4 \times 10^0$$

1000's column
100's column
10's column
1's column

- **Binary Numbers**

$$1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

8's column
4's column
2's column
1's column

Binary Numbers

- **Powers of 2**

$2^0 =$

$2^1 =$

$2^2 =$

$2^3 =$

$2^4 =$

$2^5 =$

$2^6 =$

$2^7 =$

Binary Numbers

- **More Powers of 2**

2^{10} = 1 kilo ≈ 1000 (1024)

2^{20} = 1 mega ≈ 1 million (1,048,576)

2^{30} = 1 giga ≈ 1 billion (1,073,741,824)

2^{40} = 1 tera ≈ 1 trillion

2^{50} = 1 peta ≈ 1 quadrillion

Binary Numbers

- **Converting Binary to decimal**
  - Convert 10011_2 to decimal
- **Converting Decimal to binary**
  - Convert 47_{10} to binary

Binary Numbers

- **Addition**

```
  3734
 +5168
 _____
```

```
  1011
 +0011
 _____
```

Binary Numbers

- **More Addition**

```
  1001
+0011
______

  1011
+0110
______
```

Binary Numbers

- Signed Numbers
  - How do we represent negative numbers with binary digits?
  - We could just use the left most bit to represent the negative/positive sign
  - Instead, we use the “Two’s Complement”

Binary Numbers

- Two's Complement
  - There is a two step process to represent a negative number

    1. Flip all bits
    2. Add 1

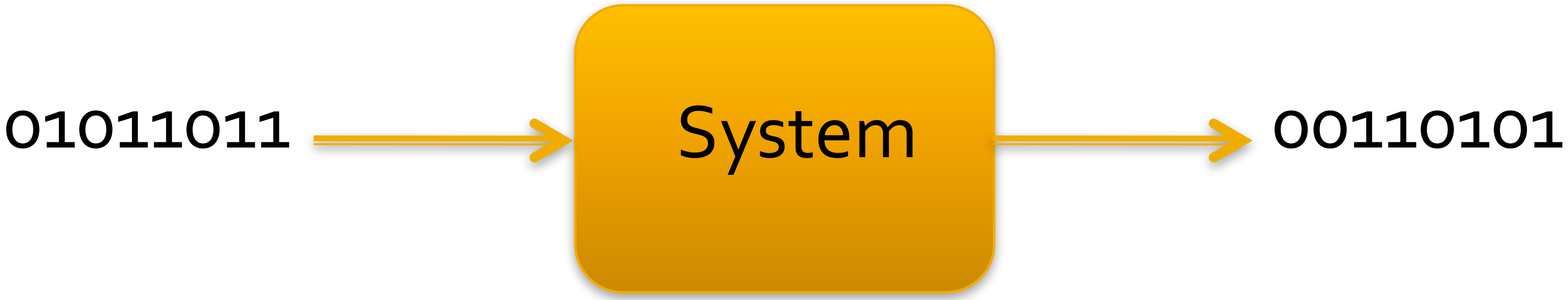
  - Note 1 - left most bit still indicates negative!
  - Note 2 – use same steps to get positive value
  - Note 3 – preserves addition!

Outline

- Zeros and Ones
- Binary Numbers
- Boolean Logic

Boolean Logic

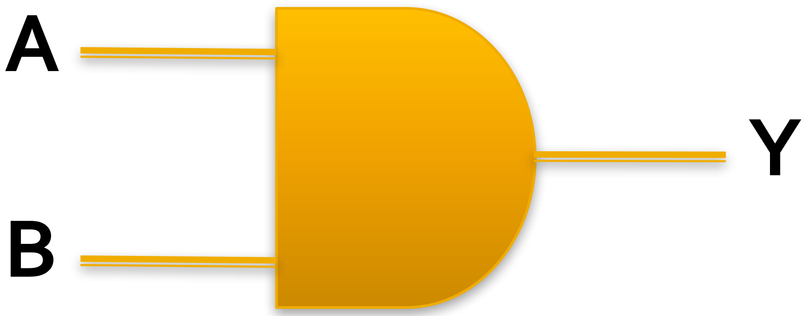
- Digital Systems
  - Have **0**'s and **1**'s as inputs
  - Have **0**'s and **1**'s as outputs



Boolean Logic

- **AND Gate**

A	B	Y
0	0	
0	1	
1	0	
1	1	

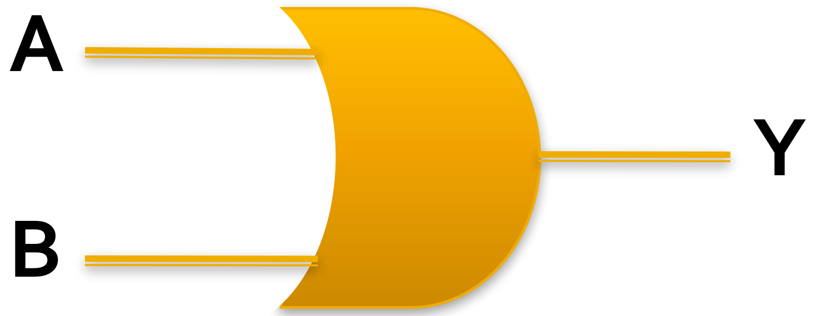


Y = AB

Boolean Logic

- OR Gate

A	B	Y
0	0	
0	1	
1	0	
1	1	

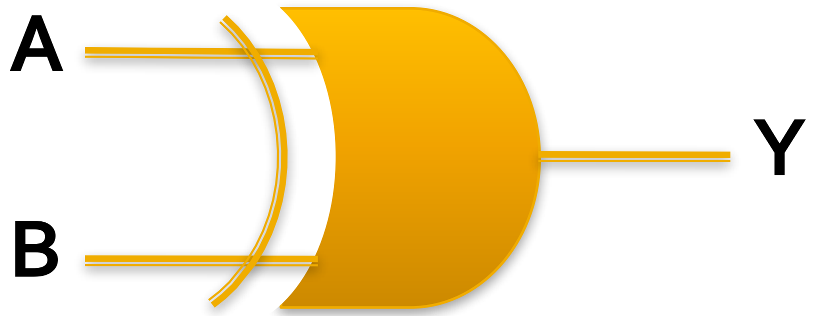


Y = A + B

Boolean Logic

- XOR Gate

A	B	Y
0	0	
0	1	
1	0	
1	1	

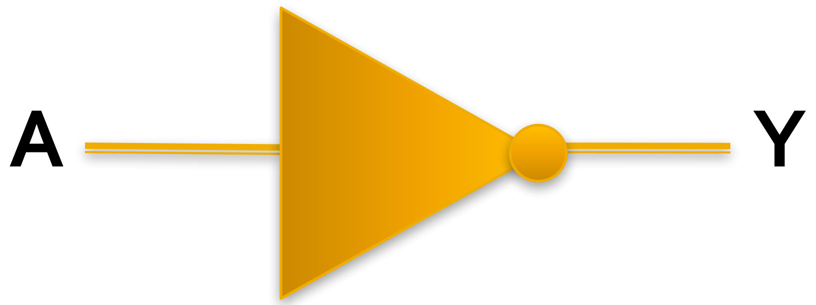


$Y = A \oplus B$

Boolean Logic

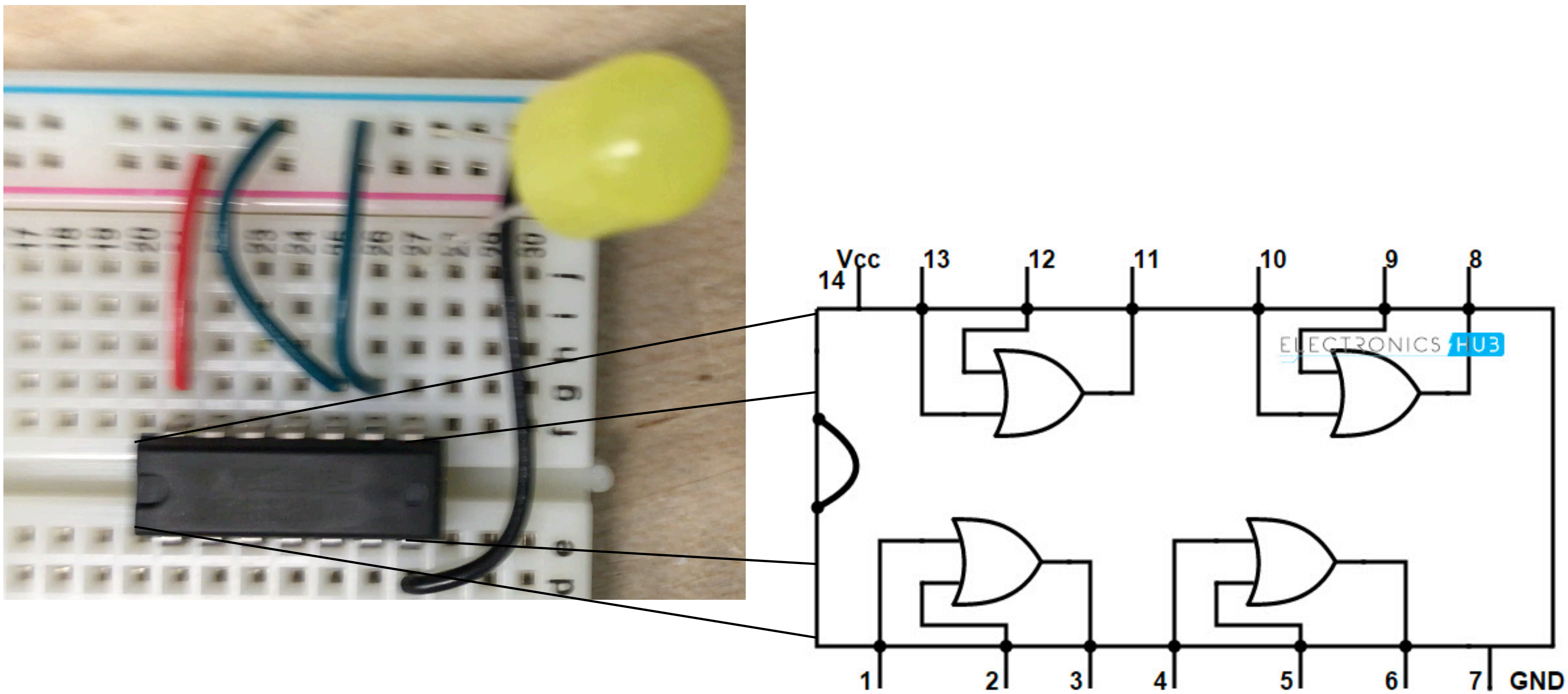
- **NOT Gate**

A	Y
0	
1	

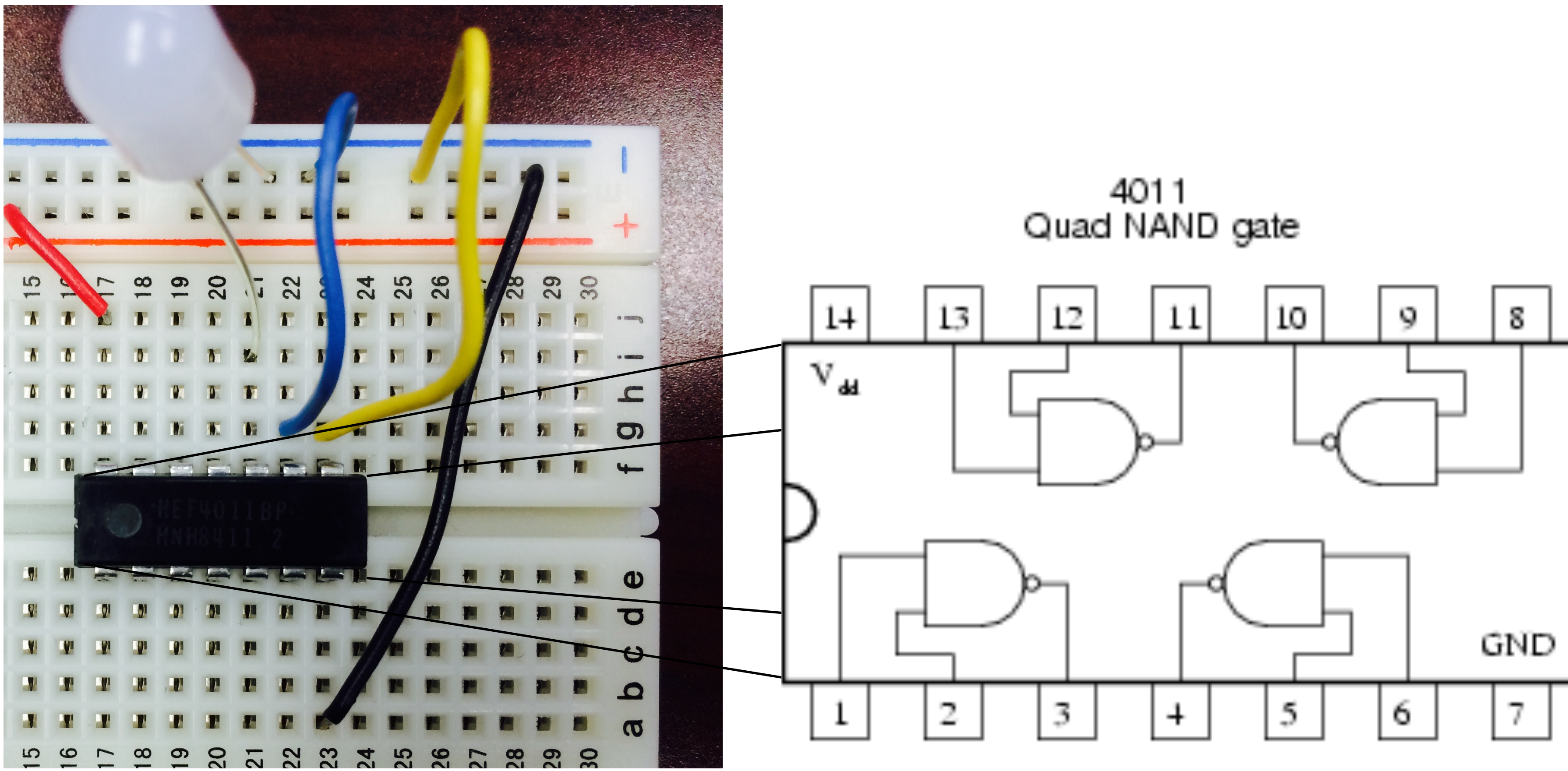


$$Y = \overline{A}$$

Boolean Logic



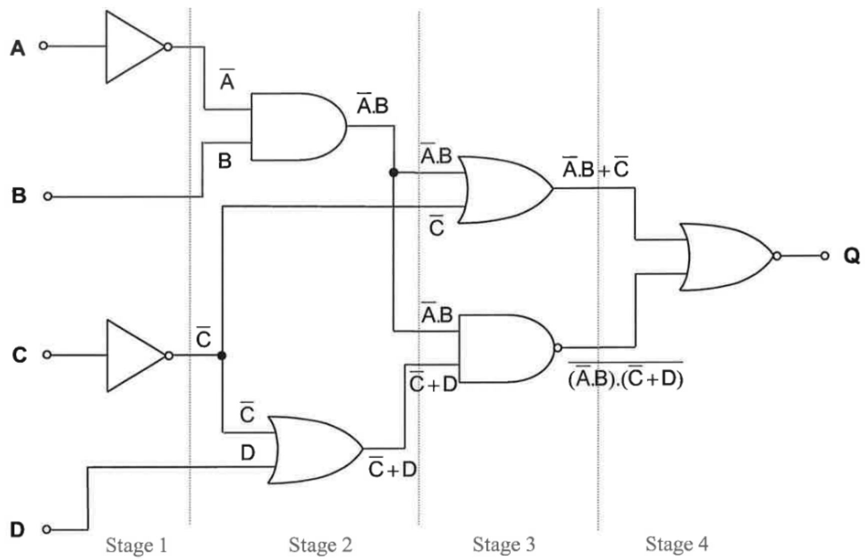
Boolean Logic



Boolean Logic

- Combining!
  - We can combine these gates to build complicated logic

Q = ?



Quick Review

- **Bits**
  - Can be understood by processors
- **Combine Bits**
  - Can represent binary numbers
- **Logic Gates**
  - Act as building blocks to create digital circuits