CS 240 Computer Systems

#3: Binary Math, Two's Complement and Logic Gates

Aug. 31, 2021 · Wade Fagen-Ulmschneider

Bit Manipulation: Binary Addition

For the past two lectures we have focused on the first foundation: **DATA**. Today, we are going to begin the transition away from data and into how data applies to the **CPU.** Binary addition work just like decimal addition, but with only **0**s and **1**s:

```
  0b 010011          0b 0011
+ 0b 001001        + 0b 0111
```

Negative Numbers: _____

```
  0b 010011          0b 0011
- 0b 001001        - 0b 0111
```

Two's Complement

The Two's Complement is a way to represent signed (ex: positive vs. negative) numbers in a way _____!

*For simplicity, let's imagine running on an **7-bit machine**:*

-17 =

-4 =

-1 =

```
  42          18
- 18        - 42
```

```
 -42          31
-  32        + 42
```

Overflow Detection in Two's Complement:

Towards Multiplication

With Two's Complement, we can add and subtract numbers! What about more complex operations?

```
10 x 2 =
```

```
10 x 4 =
```

```
10 x 9 =
```

Bit Shift Operations:

1. [Left Shift]:
2. [Right Shift]:

Logic Gates and Truth Tables
We can begin to define the building blocks of the CPU by basic instructions with input bits and output bits through **logical gates**.

- By convention, you will see that the input bits are labeled **A** and **B** by default.

Logic Gate #1:

Logic Gate #2:

Logic Gate #3:

Logic Gate Challenge: ***A XOR B***

Truth Table: Binary Addition
Can we design a circuit to complete the binary addition?

Half Adder:

A	B	A + B	SUM	CARRY

Truth Table for a Half Adder

Circuit Diagram for a "Half Adder":

Full Adder:

A	B	$CARRY_{in}$			SUM	$CARRY_{out}$

Truth Table for a Full Adder

Circuit Diagram for a "Full Adder":

Chaining Circuits Together: _____

Disadvantages: