# Digital Components: Combinational Circuits

Hui Chen [a]

[a]CUNY Brooklyn College, Brooklyn, NY, USA

October 6, 2023

# Outline

# Acknowledgement

The content of most slides come from the authors of the textbook:

Null, Linda, & Lobur, Julia (2018). The essentials of computer organization and architecture (5th ed.). Jones & Bartlett Learning.

# Table of Contents

# Lesson Objectives

Students are expected to be able to

1. Apply Boolean algebra and functions.
2. *Understand the relationship between Boolean logic and digital computer circuits.*
3. *Learn how to design simple logic circuits.*
4. Understand how digital circuits work together to form complex computer systems.

# Table of Contents

# Combinational Circuits

Combinational logic circuits produce a specified output (almost) at the instant when input values are applied.

- Half adder
- Full adder
- Ripple-carry adder
- Decoders
- Multiplexer
- Shifter

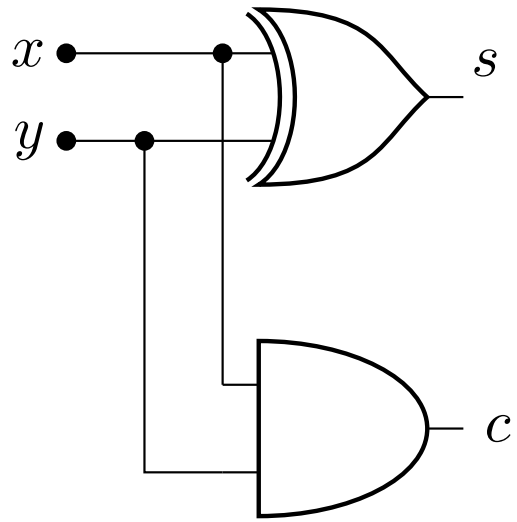
# Half Adder: Truth Table

Finds the sum of two bits.

| Inputs | | Outputs | |
|---|---|---|---|
| $x$ | $y$ | sum $(s)$ | carry $(c)$ |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

# Half Adder

Finds the sum of two bits.




| Inputs | | Outputs | |
|---|---|---|---|
| $x$ | $y$ | sum $(s)$ | carry $(c)$ |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

# Full Adder: Truth Table

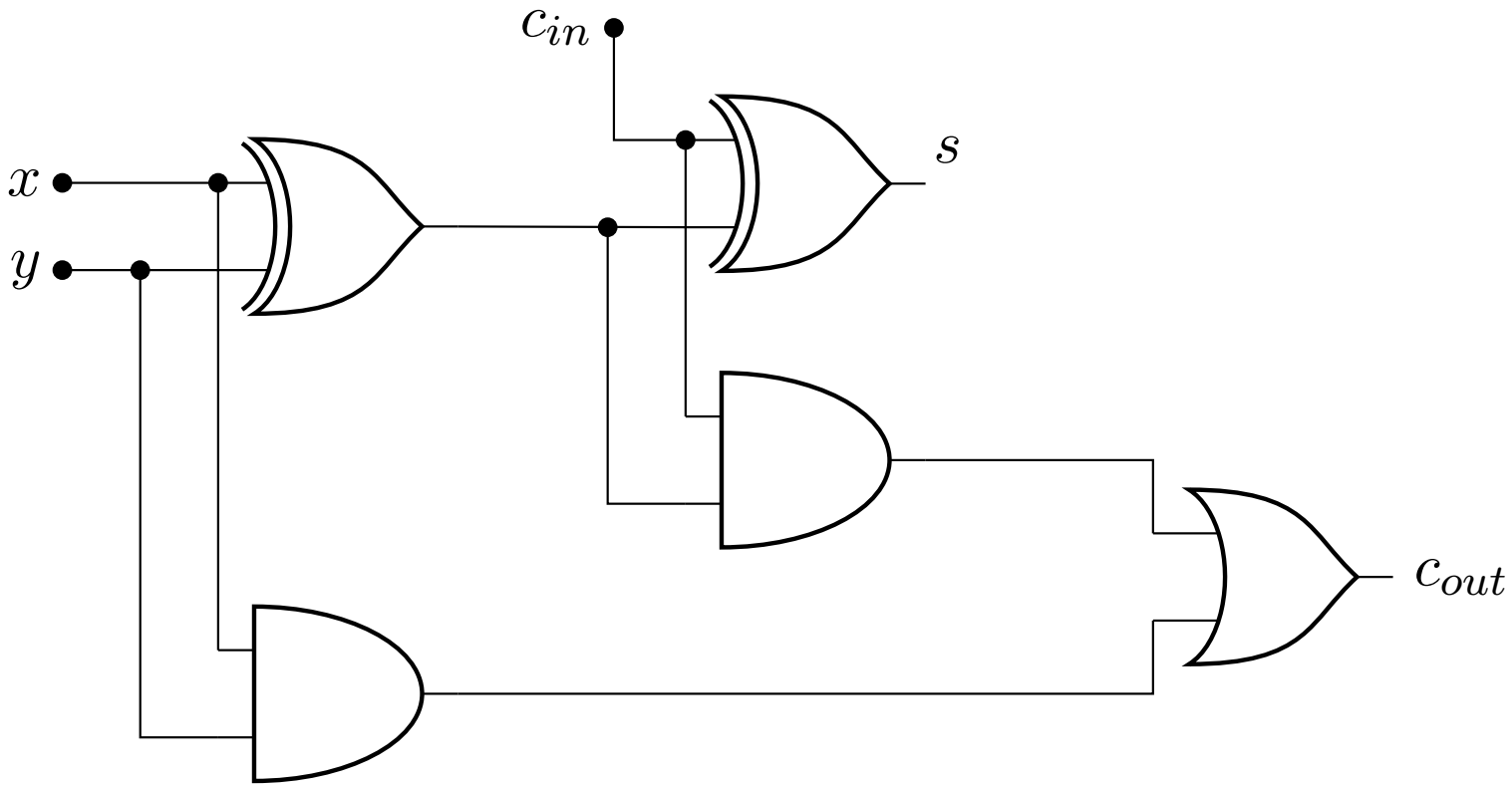
Finds the sum of two bits and carry bit

| Inputs | | | Outputs | |
|---|---|---|---|---|
| $x$ | $y$ | carry-in ($c_{in}$) | sum ($s$) | carry-out ($c_{out}$) |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Can we use the half adder as a component to construct a full adder?

# Full Adder

Finds the sum of two bits and carry bit



# Ripple-Carry Adder

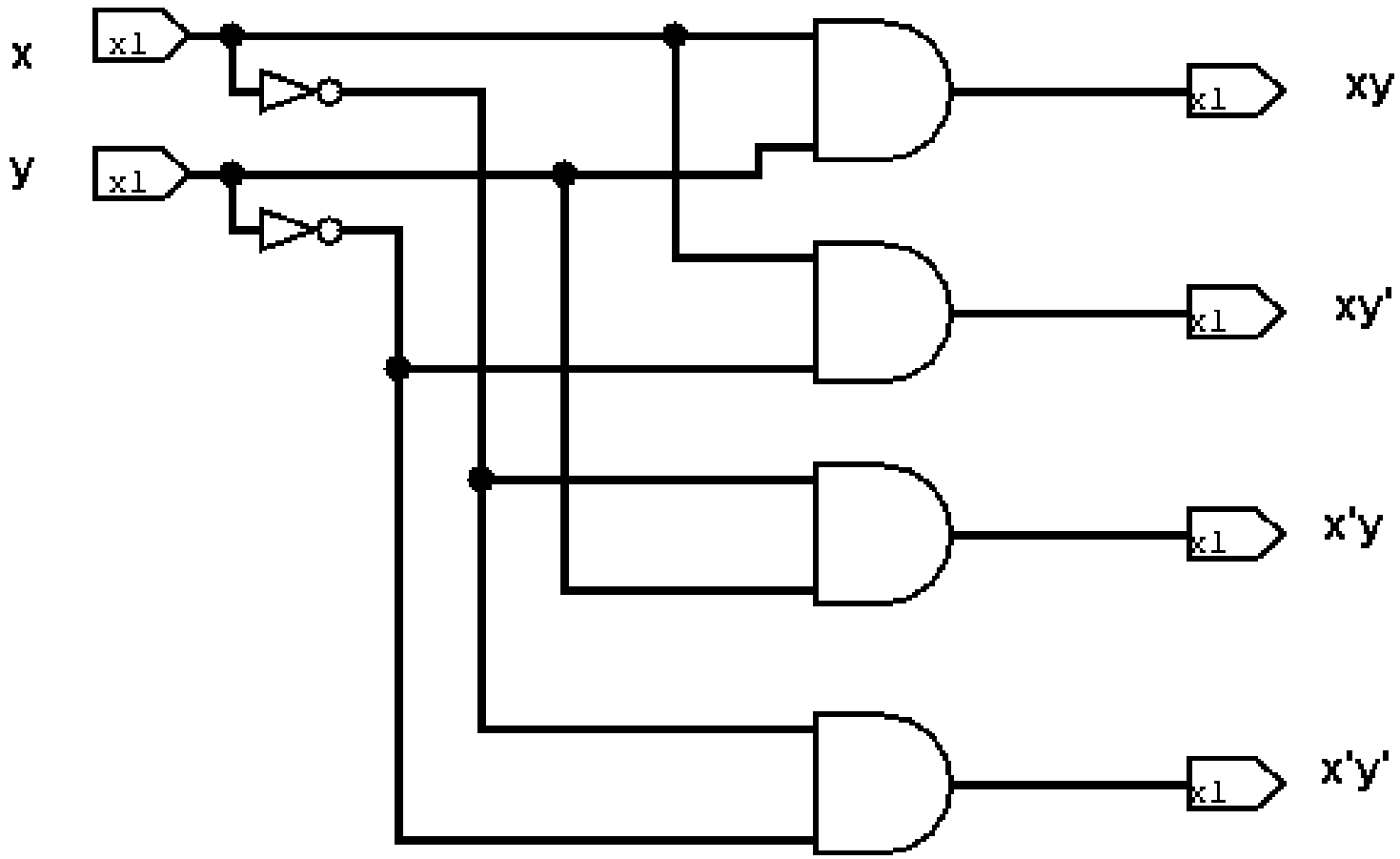
Connect full adders in series where the carry bit "ripples" from one adder to the next; hence, this configuration is called a ripple-carry adder.

# Decoder

- Decoders are useful in selecting a memory location according a binary value placed on the address lines of a memory bus.
- Address decoders with $n$ inputs can select any of $2^n$ locations.

# Decoder: Example: 2-to-4 Decoder

This is what a 2-to-4 decoder looks like on the inside.

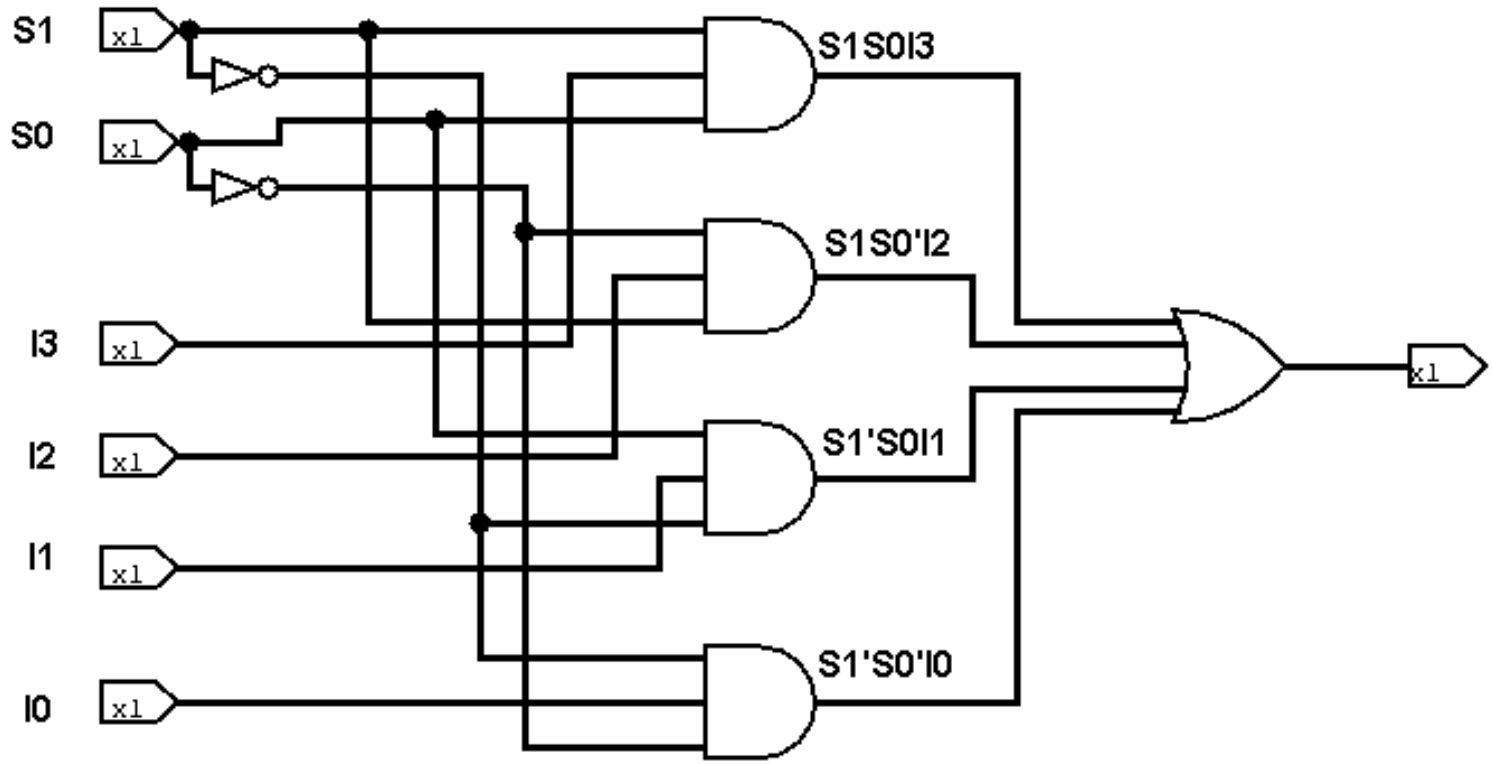


# Multiplexer

- A multiplexer does just the opposite of a decoder.
- It selects a single output from several inputs.
- The particular input chosen for output is determined by the value of the multiplexer's control lines.
- To be able to select among $n$ inputs, $\log_2 n$ control lines are needed.

# Multiplexer: Example: 4-to-1 Multiplexer

A 4-to-1 multiplexer can be designed as follows.

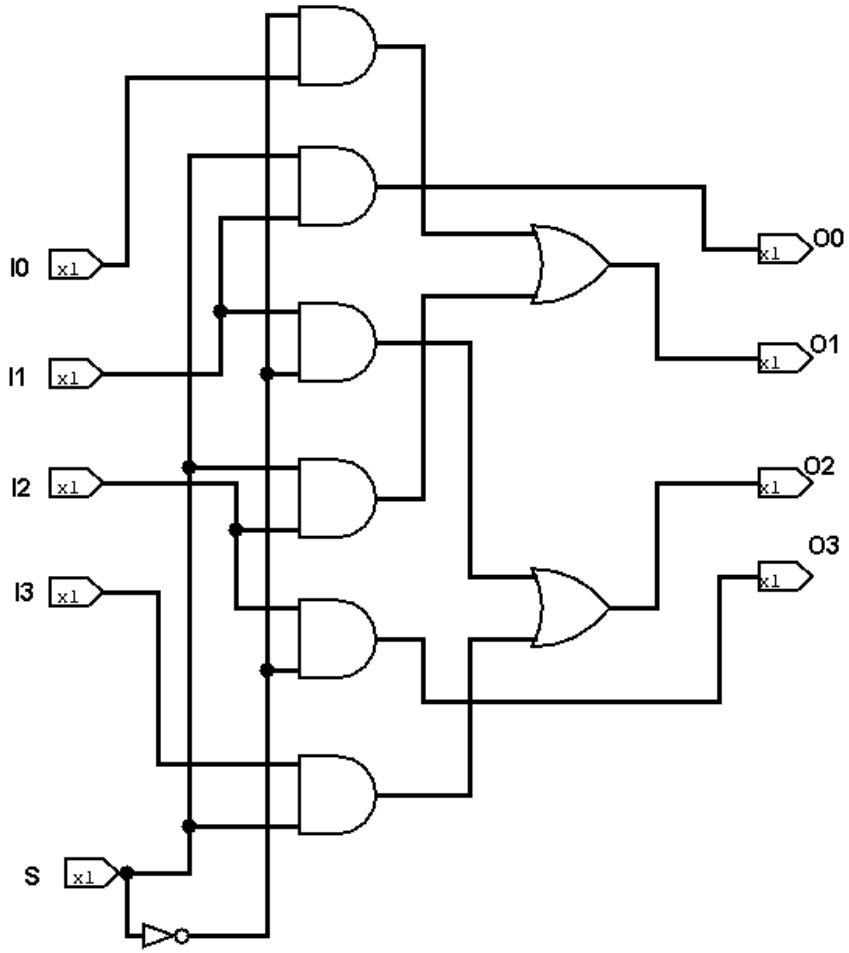


# Shifter

This shifter moves the bits of a nibble one position to the left or right.

# Shifter: Design

This shifter moves the bits of a nibble one position to the left or right.



# Table of Contents

# Summary and Q&A

You are expected to be able to

1. *Understand the relationship between Boolean logic and digital computer circuits.*
2. *Learn how to design simple logic circuits.*

Any questions on:

- Half Adder
- Full Adder
- Ripple-Carry Adder
- Decoder
- Multiplexer
- Shifter