Digital logic: Gates, Truth tables, logic equations

Yipeng Huang

Rutgers University

April 20, 2021

Table of contents

Looking ahead

Class plan

1. PA5 due Monday, 4/26.
2. Digital logic. Reading assignment: CS:APP Chapter 4.2. Recommended reading: Patterson & Hennessy, Computer organization and design, appendix on "The Basics of Logic Design." Available online via Rutgers Libraries.

Table of contents

Computer organization

Layer cake

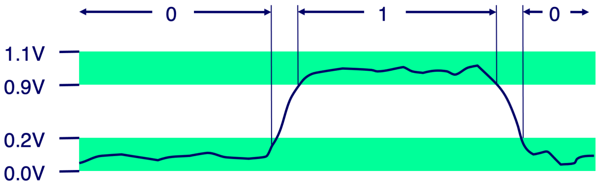
- Society
- Human beings
- Applications
- Algorithms
- High-level programming languages — Java, Python
- Interpreters
- Low-level programming languages — C, assembly
- Compilers
- Architectures
- Microarchitectures
- Sequential/combinational logic
- Transistors
- Semiconductors
- Materials science

Why binary

Everything is bits

Data representation: bits, ints, floats

- **Each bit is 0 or 1**
- **By encoding/interpreting sets of bits in various ways**
  - Computers determine what to do (instructions)
  - … and represent and manipulate numbers, sets, strings, etc…
- **Why bits? Electronic Implementation**
  - Easy to store with bistable elements
  - Reliably transmitted on noisy and inaccurate wires



To build logic, we need switches

Vacuum tubes a.k.a. valves

Figure: Source: By Stefan Riepl (Quark48) - Self-photographed, CC BY-SA 2.0 https://commons.wikimedia.org/w/index.php?curid=14682022

Transistors

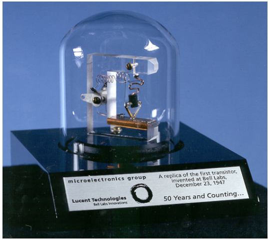


Figure: Source: Wikimedia

- The first transistor developed at Bell Labs, Murray Hill, New Jeresy
- https://www.bell-labs.com/about/locations/murray-hill-new-jersey-usa/

MOSFETs

MOS: Metal-oxide-semiconductor

- A sandwich of conductor-insulator-semiconductor.

FET: Field-effect transistor

- Gate exerts electric field that changes conductivity of semiconductor.

NMOS, PMOS, CMOS

PMOS: P-type MOS

- positive gate voltage, acts as open circuit (insulator)
- negative gate voltage, acts as short circuit (conductor)

NMOS: N-type MOS

- positive gate voltage, acts as short circuit (conductor)
- negative gate voltage, acts as open circuit (insulator)

CMOS: Complementary MOS

- A combination of NMOS and PMOS to build logical gates such as NOT, AND, OR.
- We'll go to slides posted in supplementary material to see how they work.

Combinational vs. sequential logic

Combinational logic

- No internal state nor memory
- Output depends entirely on input
- Examples: NOT, AND, NAND, OR, NOR, XOR, XNOR gates, decoders, multiplexers.

Sequential logic

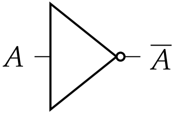
- Has internal state (memory)
- Output depends on the inputs and also internal state
- Examples: latches, flip-flops, Mealy and Moore machines, registers, pipelines, SRAMs.

Table of contents

All truth tables can be expressed in just NOT, AND and OR gates (sum-of-products form)

Just either the NAND or the NOR gate are universal to implement all combinational logic

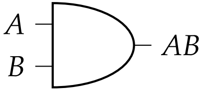
NOT gate



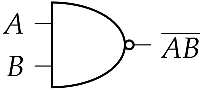
A	$\overline{A}$
0	1
1	0

Table: Truth table for NOT gate

AND gate, NAND gate



We write AND like a product



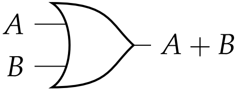
A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

Table: Truth table for AND gate

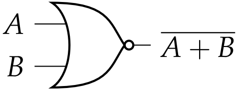
A	B	$\overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

Table: Truth table for NAND gate

OR gate, NOR gate



We write OR like a sum



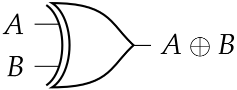
A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

Table: Truth table for OR gate

A	B	$\overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

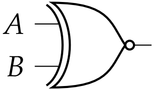
Table: Truth table for NOR gate

XOR gate, XNOR gate



A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Table: Truth table for XOR gate

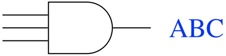


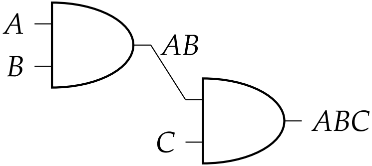
$\overline{A \oplus B}$

A	B	$\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

Table: Truth table for XNOR gate

More-than-2-input AND gate

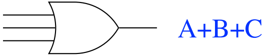


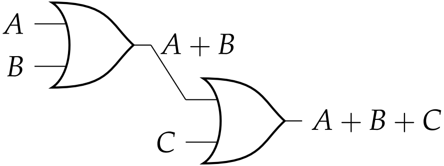


A	B	C	ABC
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Table: Truth table for three-input AND gate

More-than-2-input OR gate





A	B	C	$A+B+C$
0	0	0	0
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	1

Table: Truth table for three-input AND gate