UC Berkeley
Teaching Professor
Dan Garcia

CS61C

Great Ideas in Computer Architecture

(a.k.a. Machine Structures)

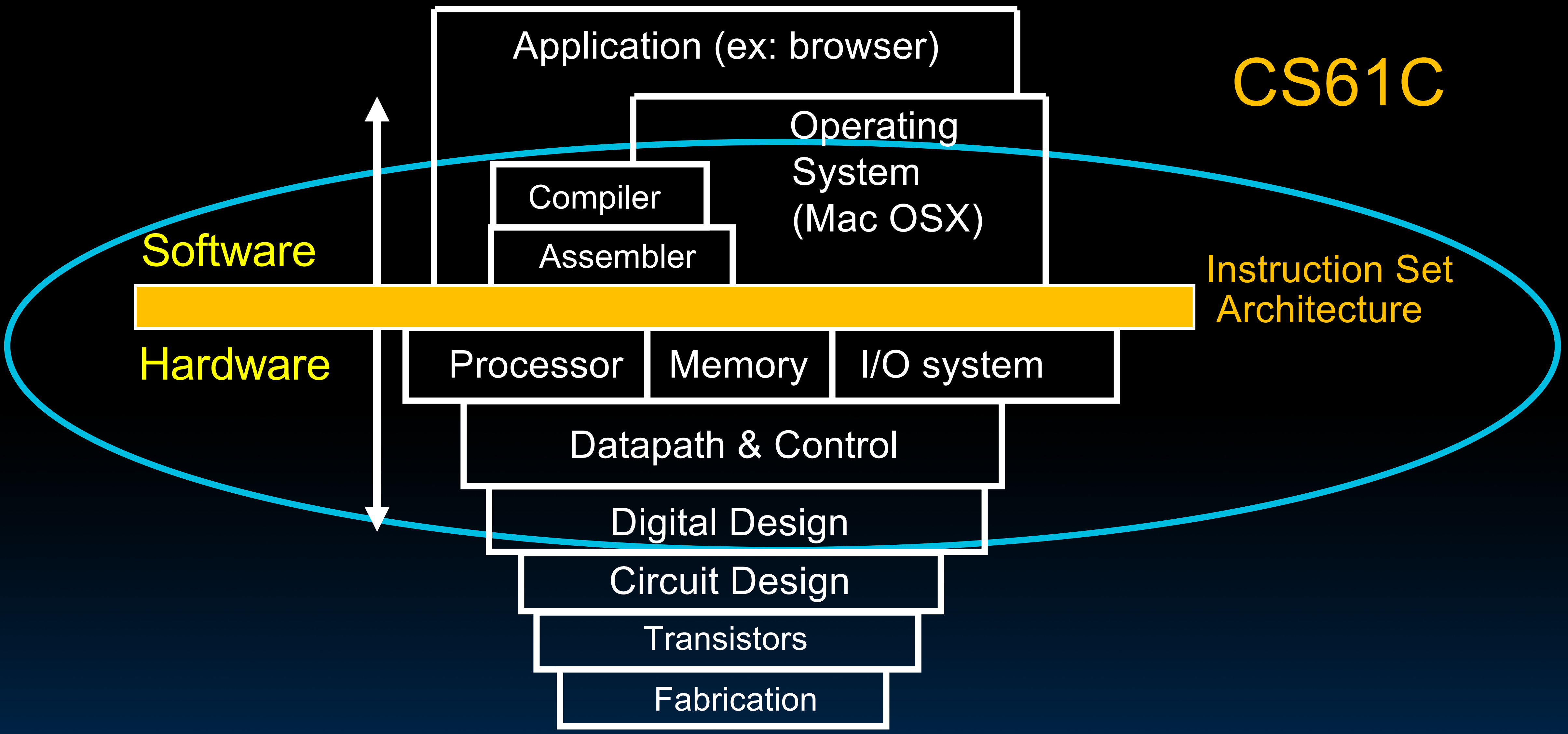
Introduction to Synchronous Digital Systems (SDS): Switches, Transistors, Signals, & Waveforms

Berkeley
UNIVERSITY OF CALIFORNIA

cs61c.org

Switches

Machine Structures



New-School Machine Structures

Software

Harness Parallelism & Achieve High Performance

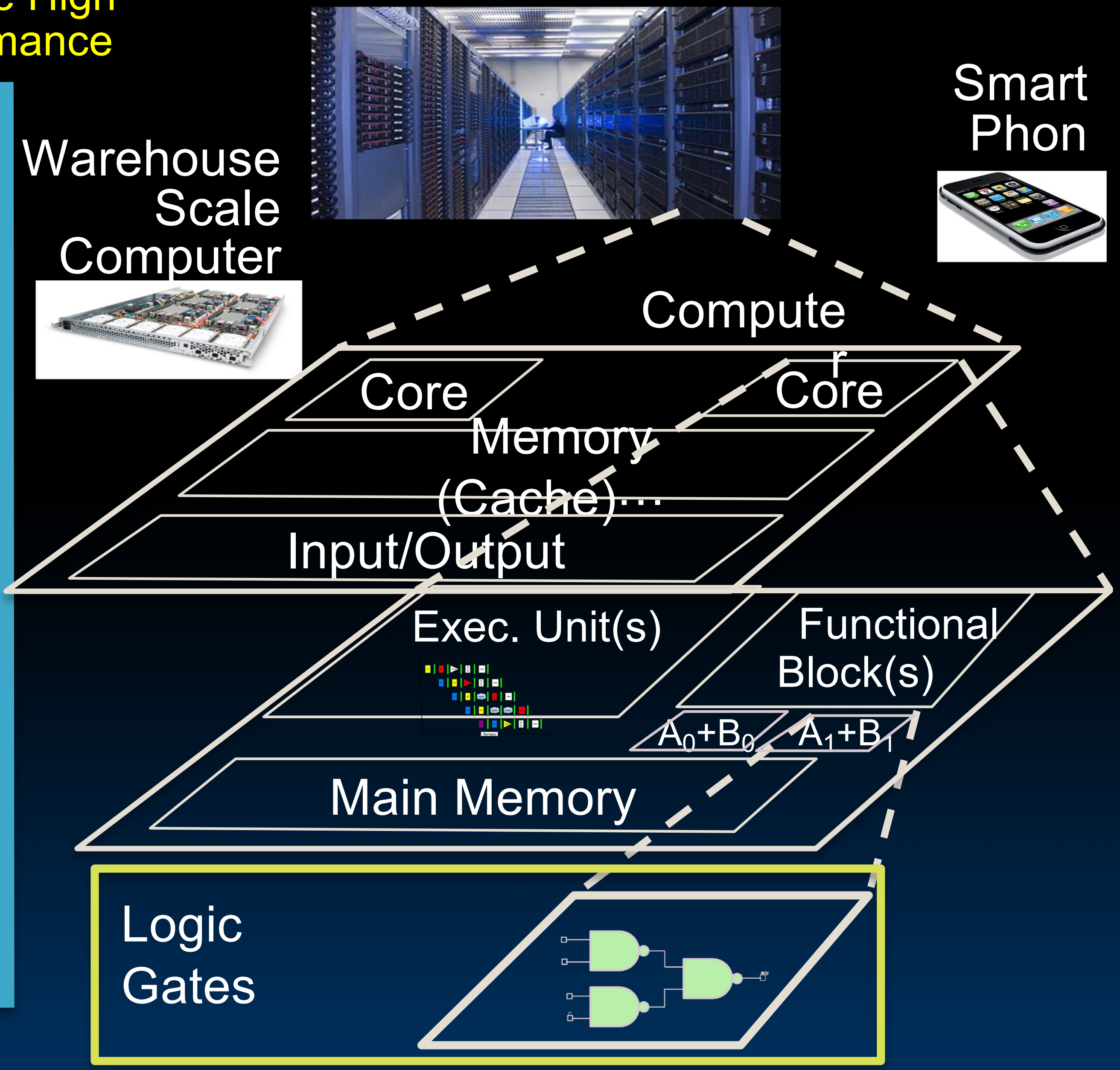
Parallel Requests
Assigned to computer
e.g., Search "Cats"

Parallel Threads
Assigned to core e.g., Lookup, Ads

Parallel Instructions
>1 instruction @ one time
e.g., 5 pipelined instructions

Parallel Data
>1 data item @ one time
e.g., Add of 4 pairs of words

Hardware descriptions
All gates work in parallel at same time



Great Idea #1: Abstraction (Levels of Representation/Interpretation)

High Level Language Program (e.g., C)

```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```

Compiler

Assembly Language Program (e.g., RISC-V)

```
lw    x3, 0(x10)
lw    x4, 4(x10)
sw    x4, 0(x10)
sw    x3, 4(x10)
```

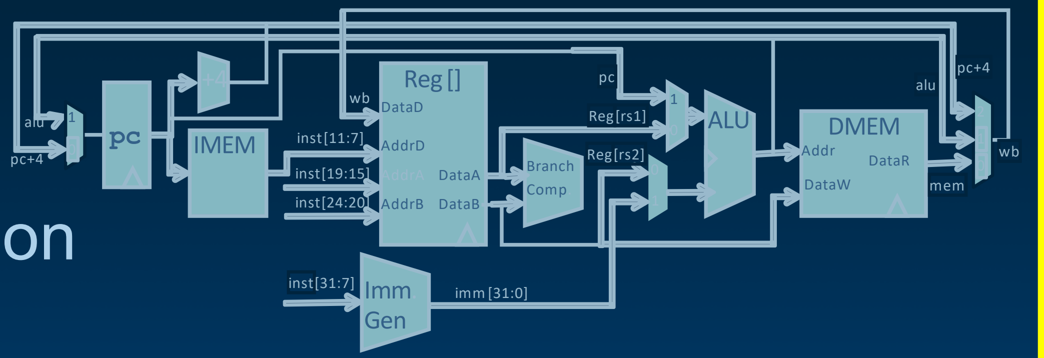
Assembler

Machine Language Program (RISC-V)

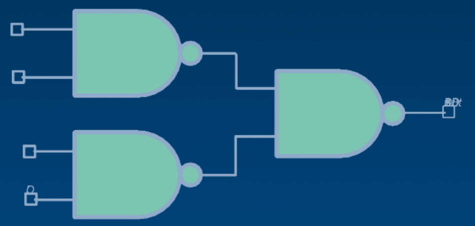
```
1000 1101 1110 0010 0000 0000 0000 0000
1000 1110 0001 0000 0000 0000 0000 0100
1010 1110 0001 0010 0000 0000 0000 0000
1010 1101 1110 0010 0000 0000 0000 0100
```

Hardware Architecture Description (e.g., block diagrams)

Architecture Implementation



Logic Circuit Description (Circuit Schematic Diagrams)



Synchronous Digital Systems

Hardware of a processor, e.g., RISC-V, is a Synchronous Digital System

Synchronous:

- All operations coordinated by a central clock
- "Heartbeat" of the system!

Digital:

- All values represented by discrete values
- Electrical signals are treated as 1s and 0s; grouped together to form words

Logic Design

Next several weeks: we'll study how a modern processor is built; starting with basic elements as building blocks

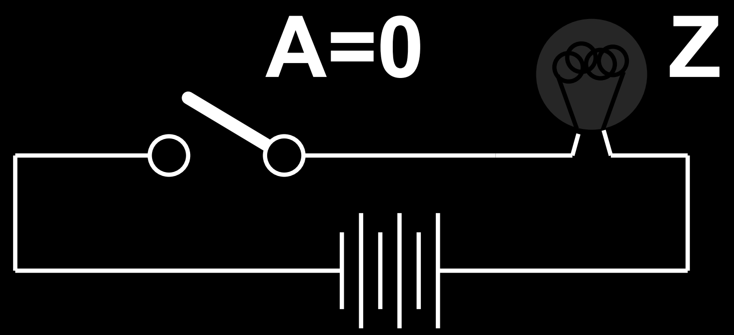
Why study hardware design?

- Understand capabilities and limitations of HW in general and processors in particular
- What processors can do fast and what they can't do fast (avoid slow things if you want your code to run fast!)
- Background for more in depth HW courses (150, 152)
- There is just so much you can do with standard processors: you may need to design own custom HW for extra performance

Water Analogy for Electricity

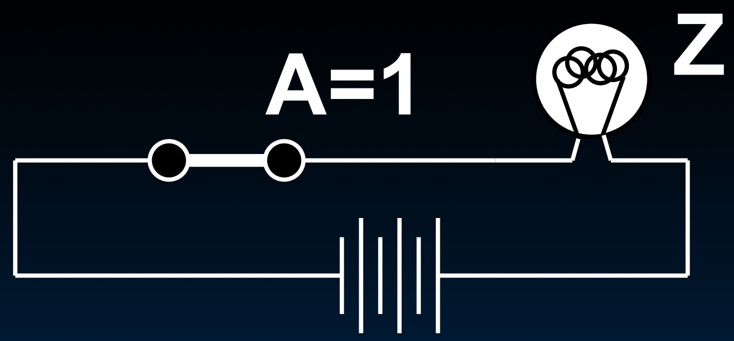
Switches: Basic Element of Physical Circuit

- Implementing a simple circuit
  - Switch open when **A** is 0, closed when **A** is 1

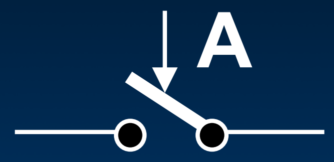


Switch open (if **A** is "0" or unasserted)
so no current flow, light bulb (**Z**) off = 0

$\boxed{Z \equiv A}$



Switch closed (if **A** is "1" or asserted)
so current flows, light bulb (**Z**) on = 1

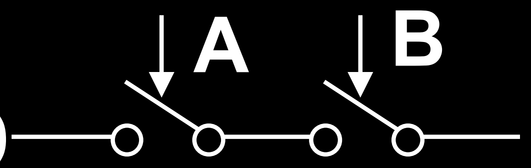


In general, we'll draw A near an arrow, indicating that setting A to 1 involes "closing" the switch.

Switches (continued)

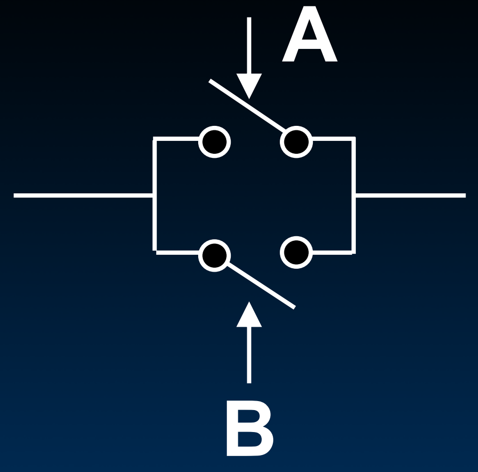
Compose switches into more complex ones (Boolean functions):

AND



$Z \equiv A$ **and** B

OR



$Z \equiv A$ **or** B

Historical Note

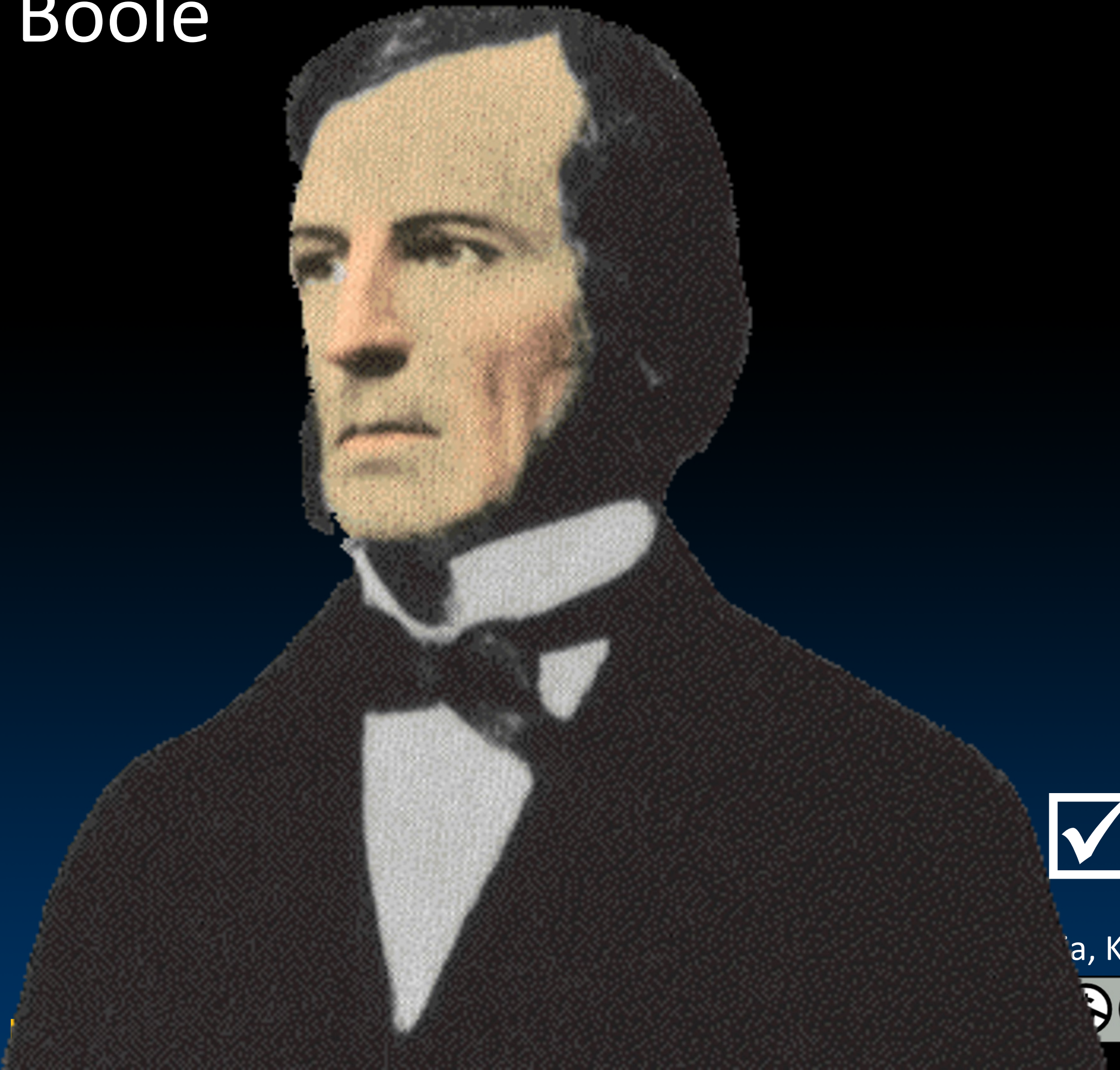
Early computer designers built ad hoc circuits from switches

Began to notice common patterns in their work: ANDs, ORs, …

Master's thesis (by Claude Shannon) made link between transistors and 19th Century Mathematician George Boole

- Called it "Boolean" in his honor

Could apply math to give theory to hardware design, minimization, …



Transistors

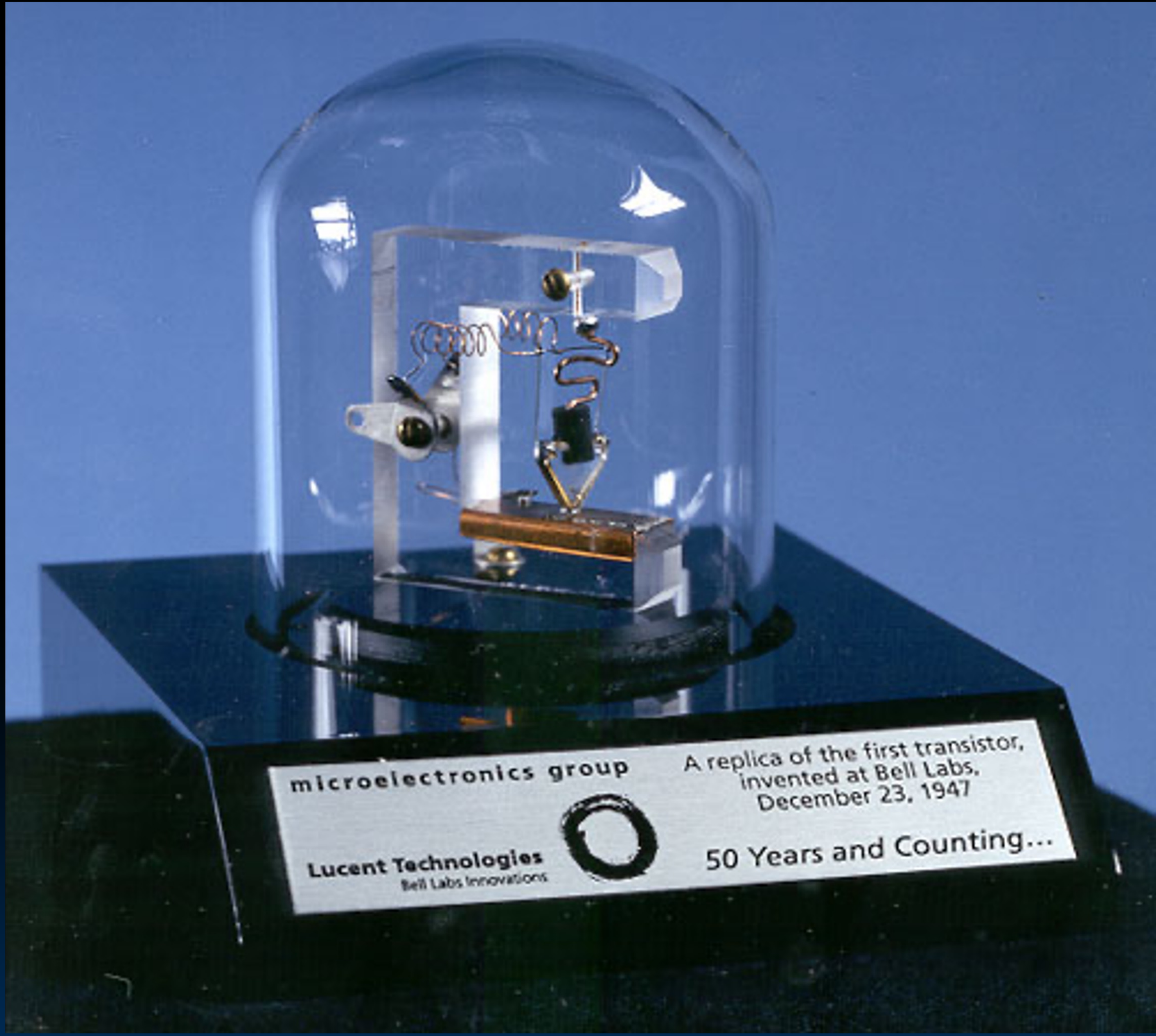
The Transistor ("born" 1947-12-23)

en.wikipedia.org/wiki/History_of_the_transistor
www.pbs.org/transistor
youtu.be/-td7YT-Pums
youtu.be/OwS9aTE2Go4

- Semiconductor device to amplify or switch signals
  - Key component in ALL modern electronics
- Before that?
  - Vacuum Tubes

- After that?
  - Integrated circuit, microprocessor

"The Transistor was probably THE most important invention of the 20th Century"
- Ira Flatow, Transistorized! (PBS Special)



Transistor Networks

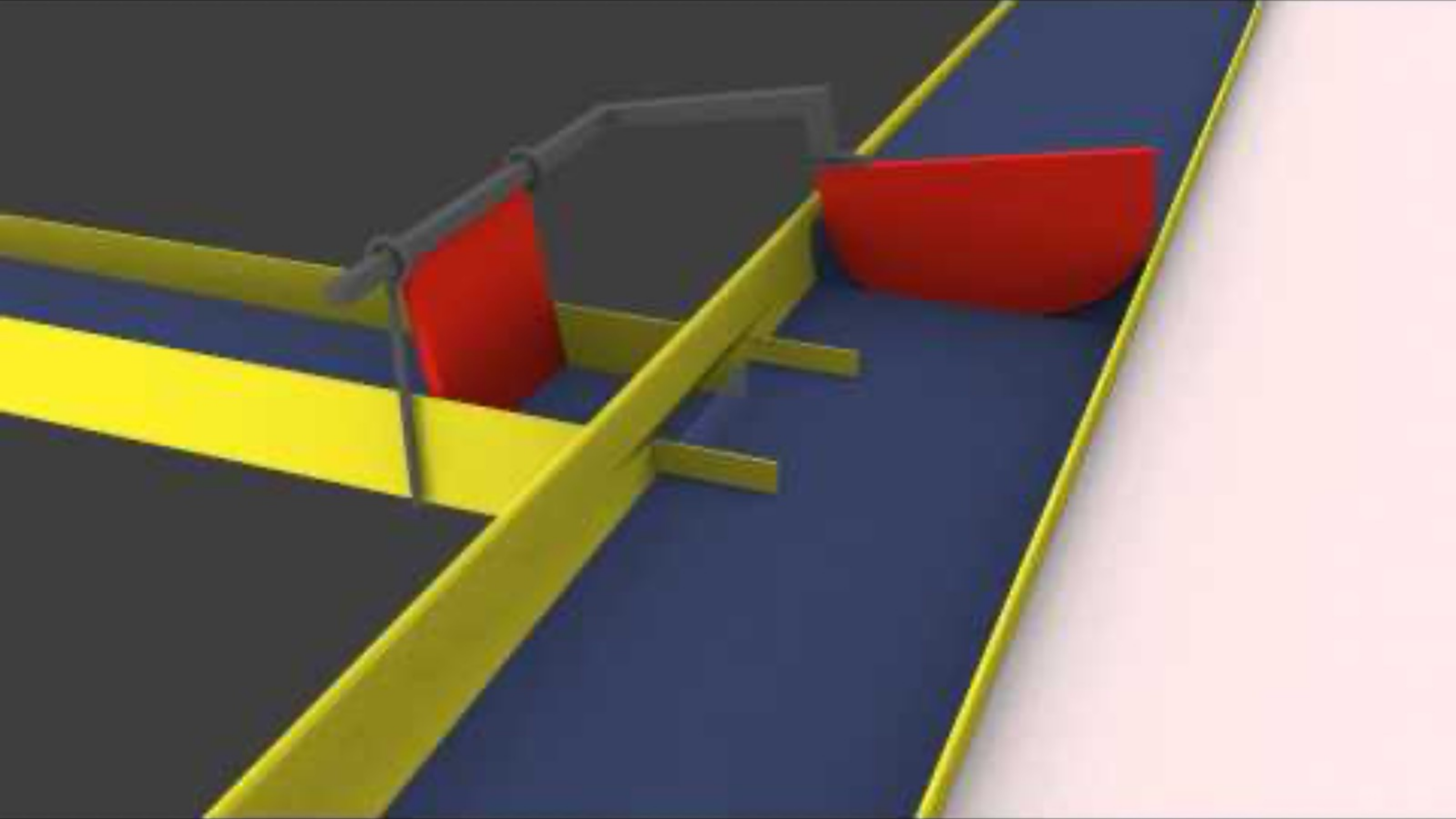
Modern digital systems designed in CMOS

- MOS: Metal-Oxide on Semiconductor
- C for complementary: normally-open and normally-closed switches

MOS transistors act as voltage-controlled switches

Analogy … WATER is controlling the switch!

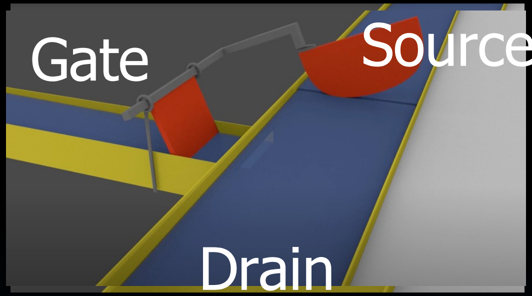
Garcia, Kao

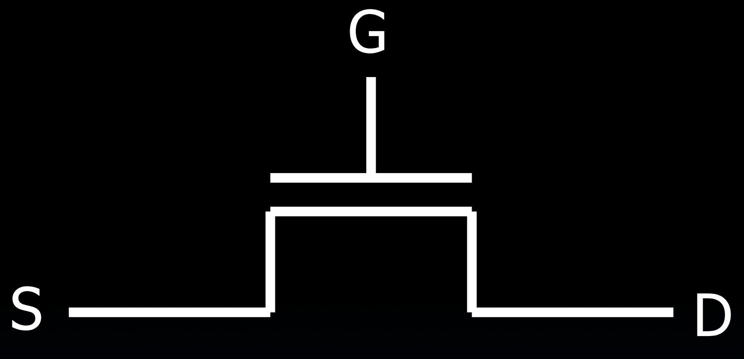


MOS (metal oxide semiconductor) Transistors

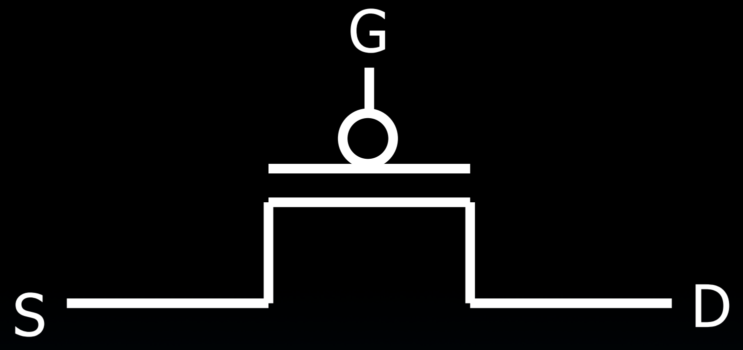
- Three terminals: Drain, Gate, Source
  - Switch action: Dan Garcia Says
    if voltage on gate terminal is (some amount) higher/lower than source terminal then conducting path established between drain and source terminals

To remember: n ("normal"),
p (has a circle, like the top part of P itself)





n-channel
open when voltage at G is low
closes when voltage at G is high
voltage(G) > voltage (S) + ε



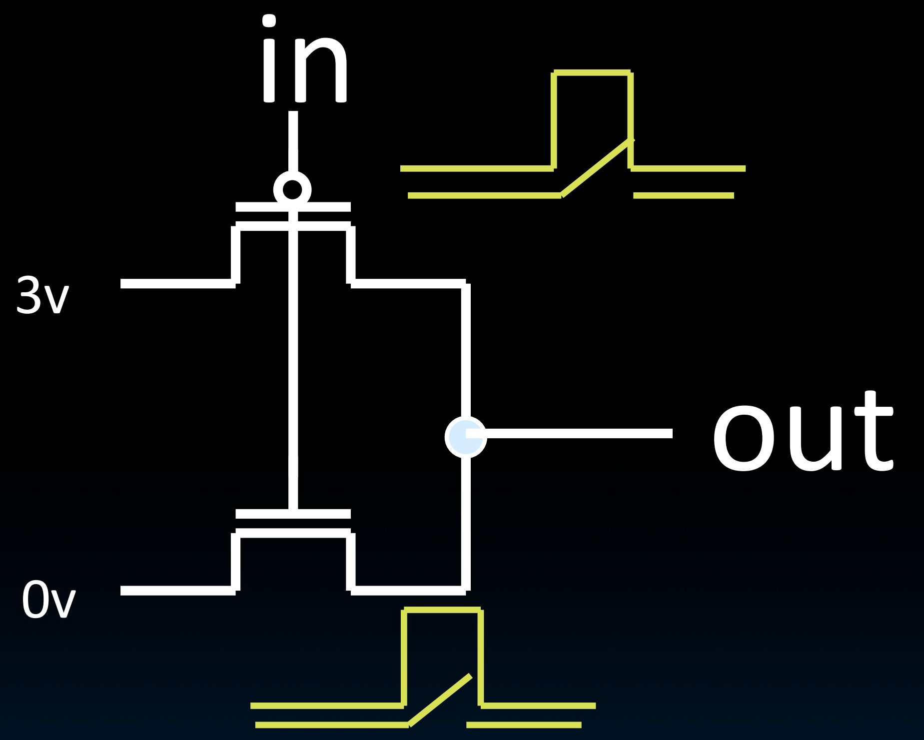
p-channel
closed when voltage at G is low
opens when voltage at G is high
voltage(G) > voltage (S) + ε

G LOW
G HIGH

MOS Networks

“1”
(voltage
source)

“0”
(ground)



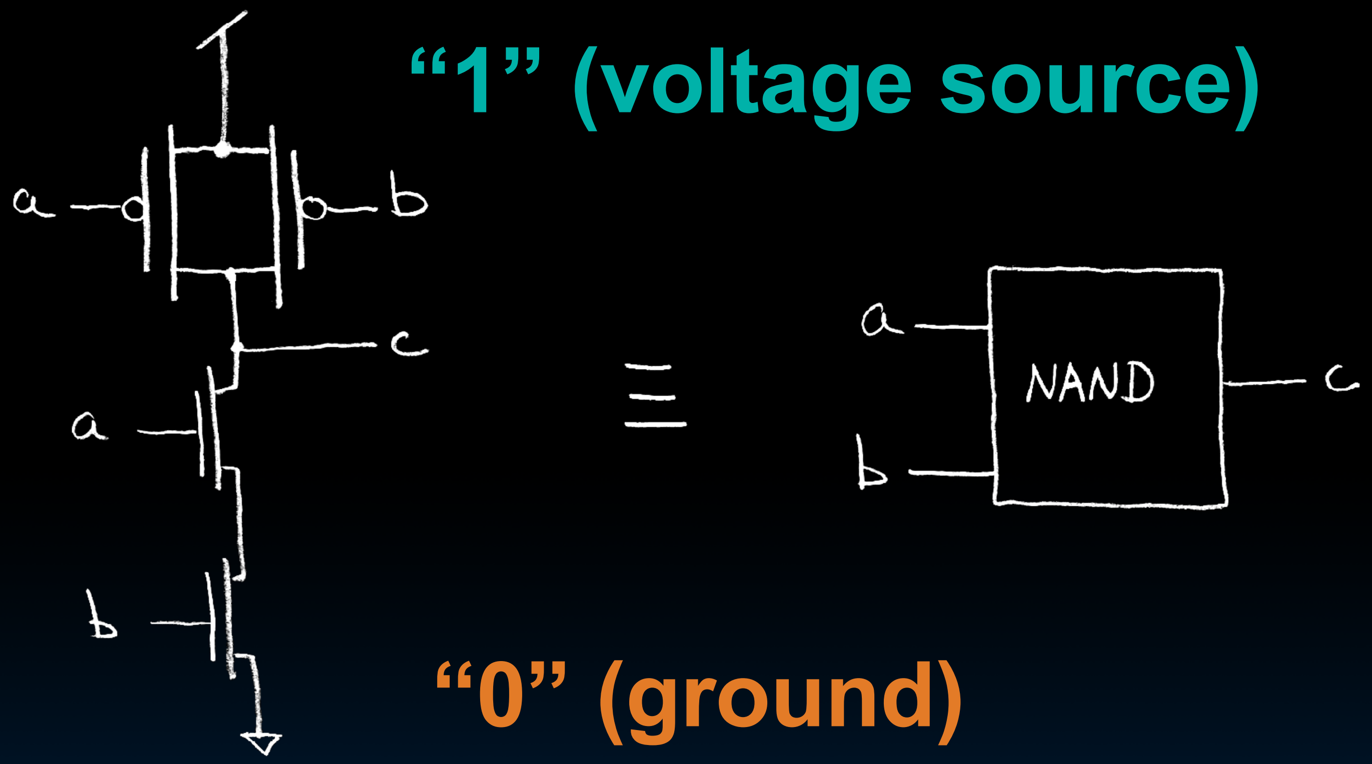
Relationship between in and out ?

in	out
0 volts	3 volts
3 volts	0 volts

Transistor Circuit Rep. vs. Block diagram

Chips are composed of nothing but transistors and wires.

Small groups of transistors form useful building blocks.



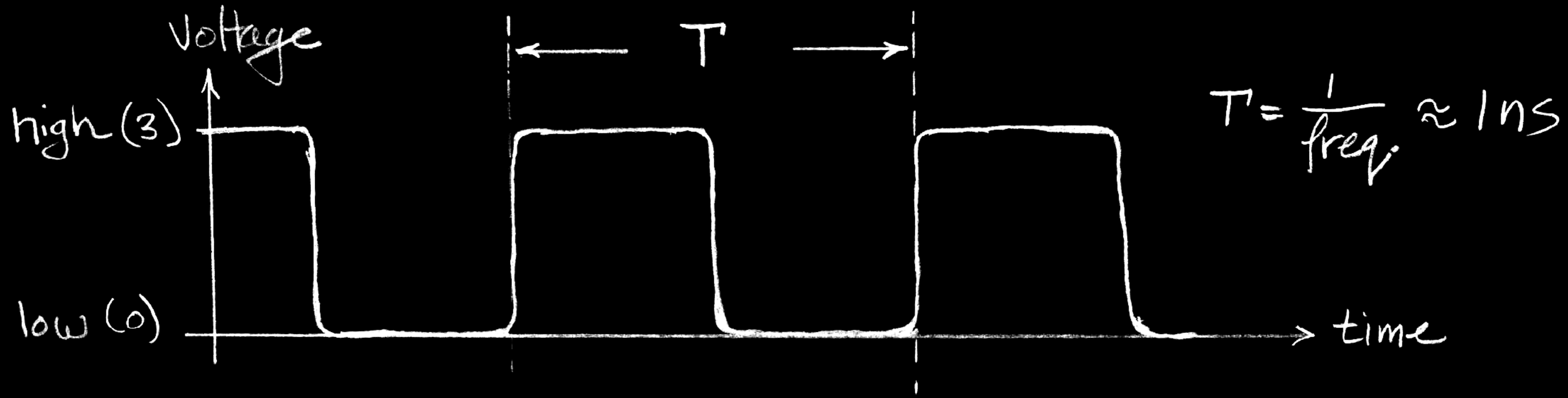
a	b	c
0	0	1
0	1	1
1	0	1
1	1	0

Block are organized in a hierarchy to build higher-level blocks: ex: adders.

You can build AND, OR, NOT out of NAND!

Signals and Waveforms

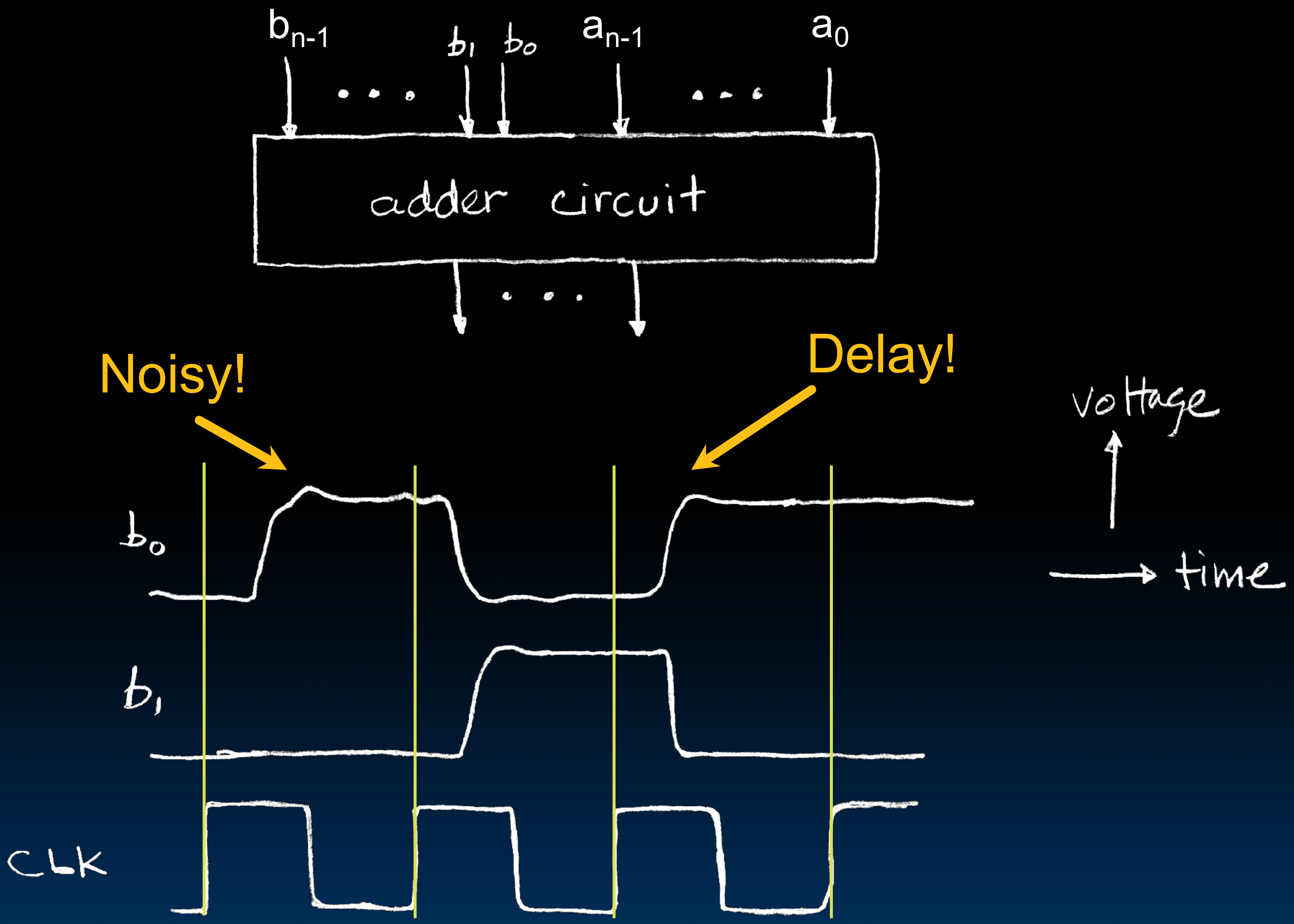
Signals and Waveforms: Clocks



Signals

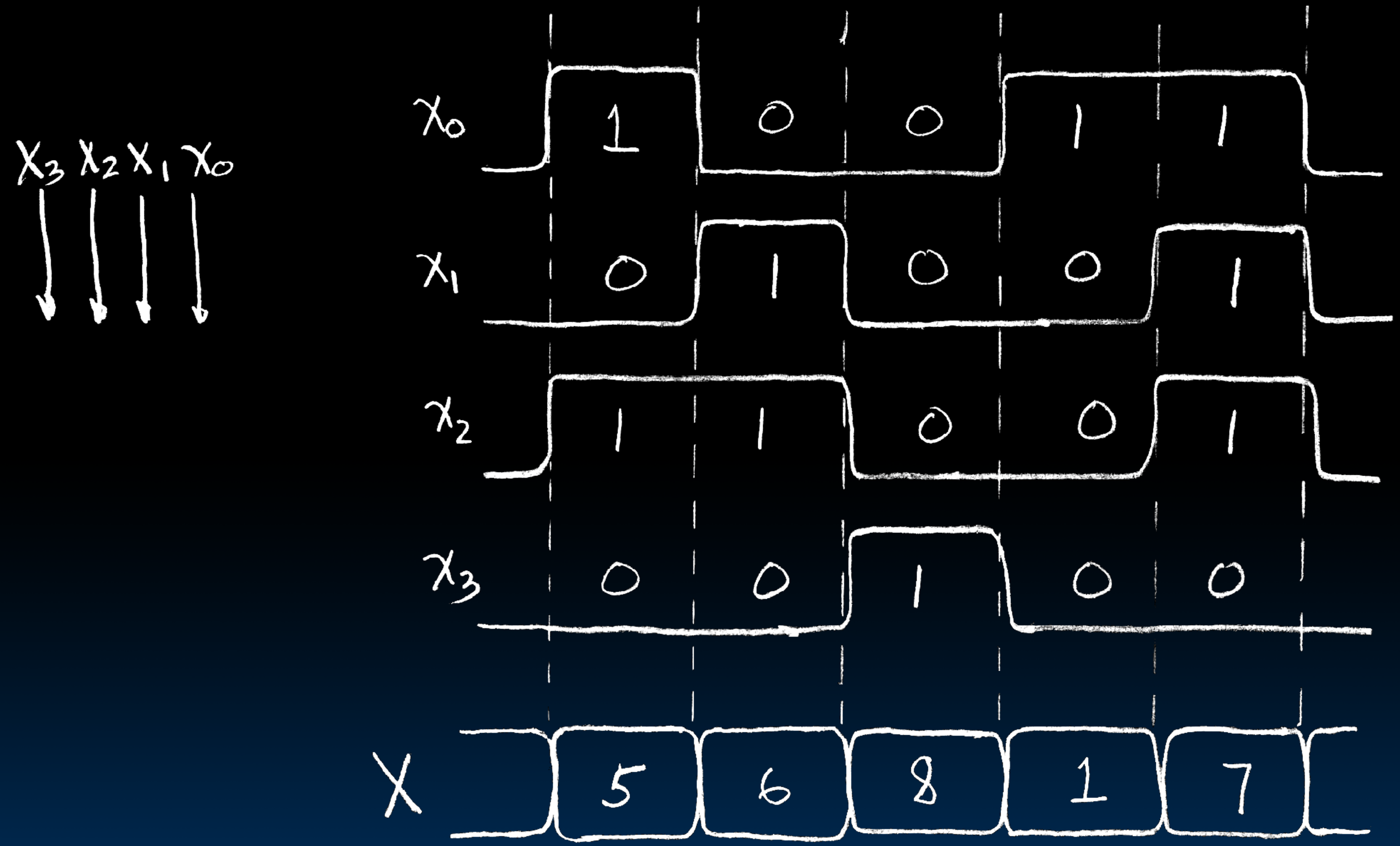
- When digital is only treated as 1 or 0
- Is transmitted over wires continuously
- Transmission is effectively instant
- Implies that a wire contains 1 value at a time

Signals and Waveforms



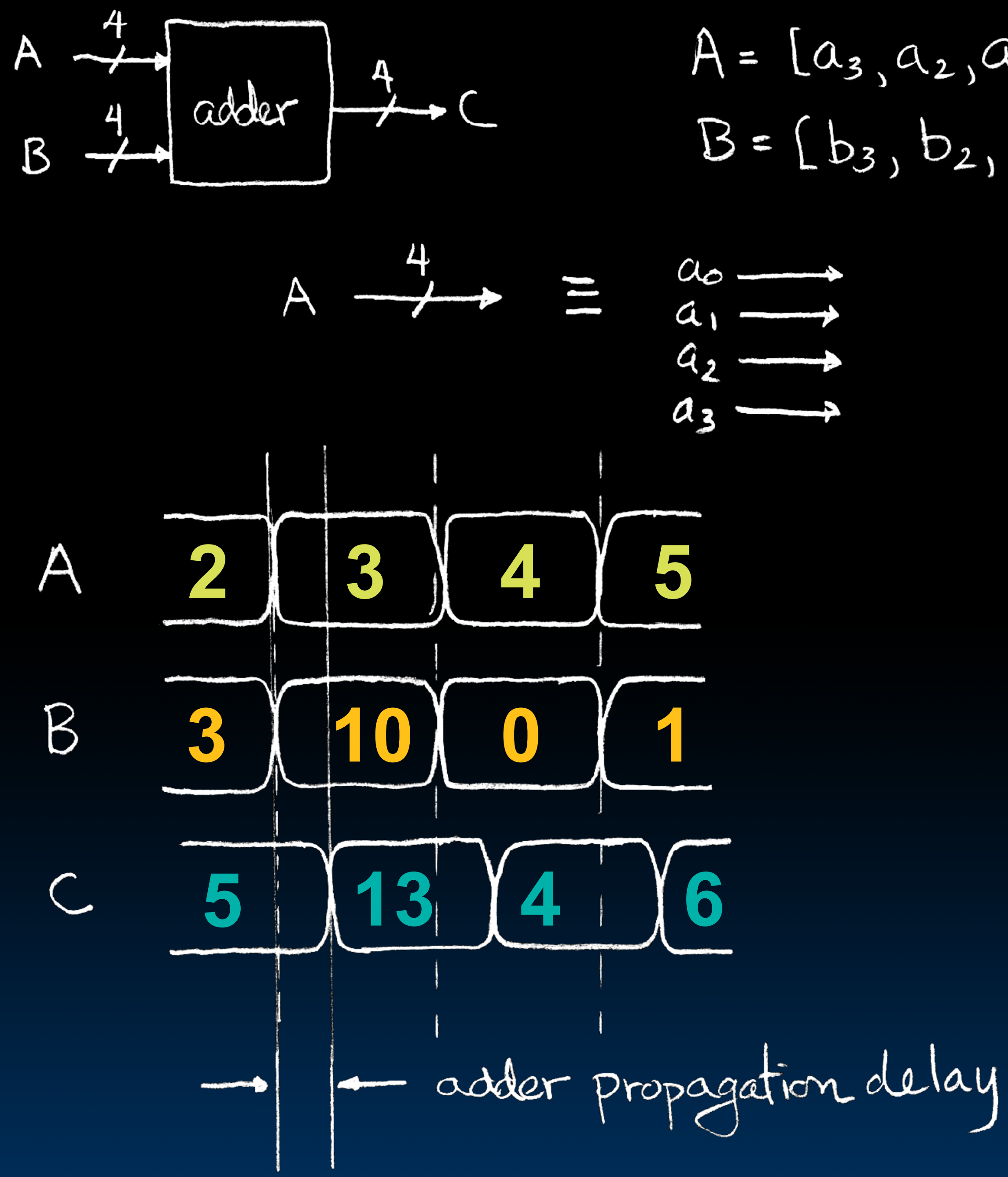
Garcia, Kao

Signals and Waveforms: Grouping

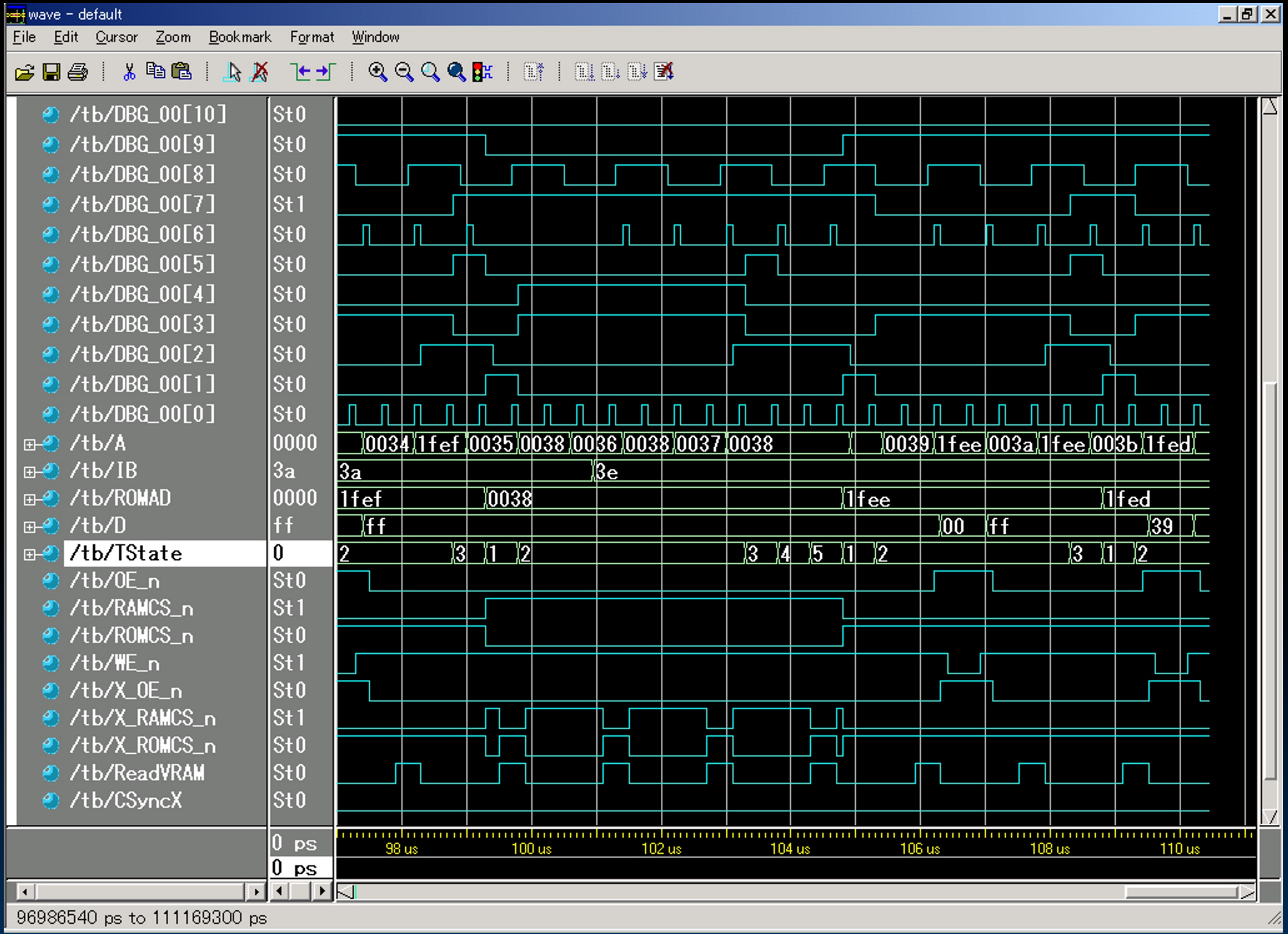


Garcia, Kao

Signals and Waveforms: Circuit Delay



Sample Debugging Waveform



Berkeley UNIVERSITY OF CALIFORNIA

Garcia, Kao

Type of Circuits

Synchronous Digital Systems are made up of two basic types of circuits:

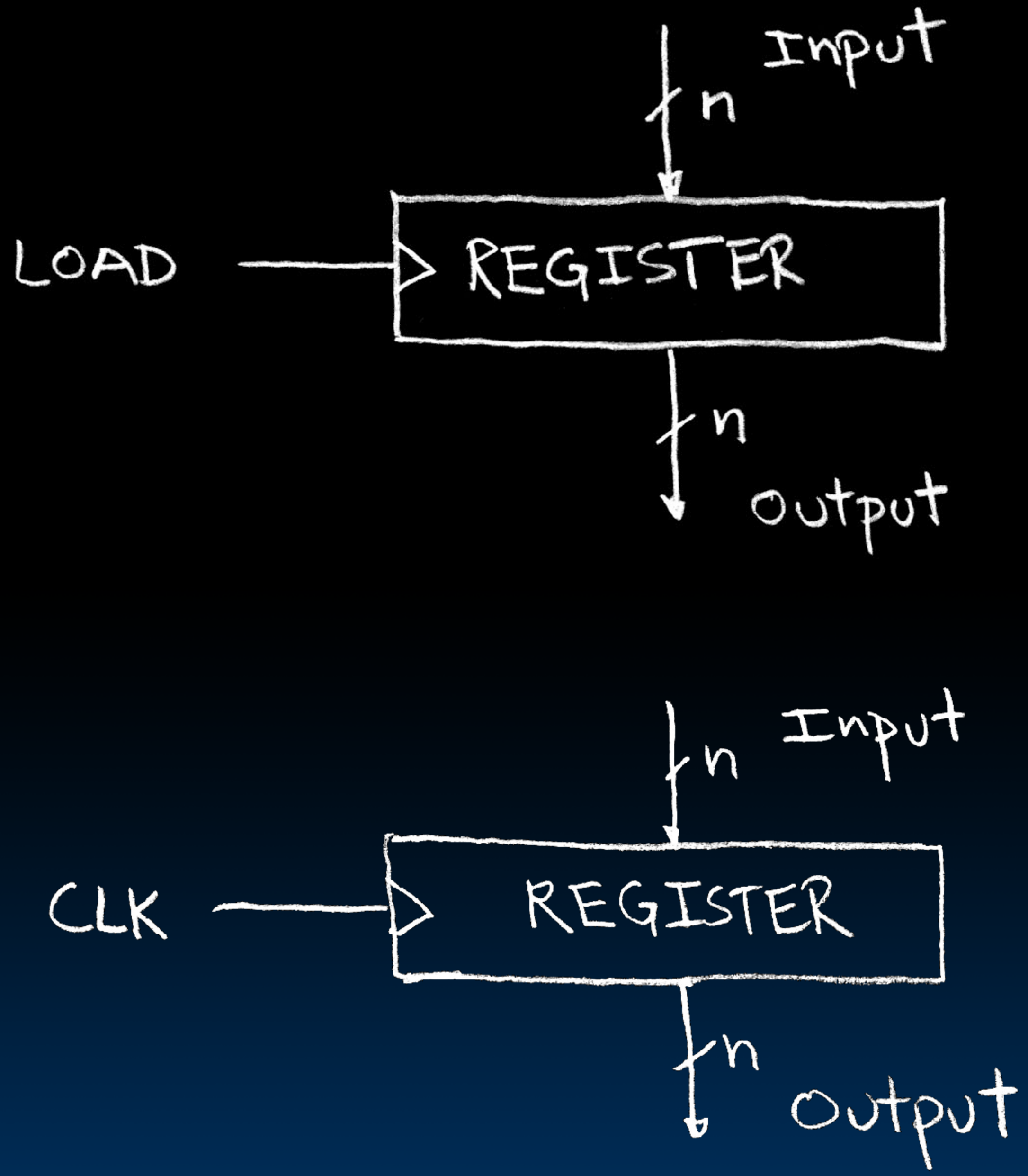
Combinational Logic (CL) circuits

- Our previous adder circuit is an example.
- Output is a function of the inputs only.
- Similar to a pure function in mathematics, y = f(x). (No way to store information from one invocation to the next, no side effects)

State Elements

- circuits that store information.

Circuits with STATE (e.g., register)



L14 SW can peek at HW (past ISA abstraction boundary) for optimizations | SW can depend on particular HW implementation of ISA | Timing diagrams serve as a critical debugging tool in the EE toolkit

F F F	0%
F F T	0%
F T F	0%
F T T	0%
T F F	0%
T F T	0%
T T F	0%
T T T	0%

And in conclusion…

Clocks control pulse of our circuits

Voltages are analog, quantized to 0/1

Circuit delays are fact of life

Two types of circuits:

- Stateless Combinational Logic (&, |, ~)
- State circuits (e.g., registers)

Garcia, Kao