CS250P: Computer Systems Architecture Circuits Recap – Digital Why And How

Sang-Woo Jun

Fall 2023

UCI

Large amount of material adapted from MIT 6.004, “Computation Structures”,
Morgan Kaufmann “Computer Organization and Design: The Hardware/Software Interface: RISC-V Edition”,
and CS 152 Slides by Isaac Scherson

Course outline

- Part 1: The Hardware-Software Interface
  - What makes a ‘good’ processor?
  - Assembly programming and conventions
- Part 2: Recap of digital design
  - Combinational and sequential circuits
  - How their restrictions influence processor design
- Part 3: Computer Architecture
  - Computer Arithmetic
  - Simple and pipelined processors
  - Caches and the memory hierarchy
- Part 4: Computer Systems
  - Operating systems, Virtual memory

“Complex ISA can slow down the clock”

Why?

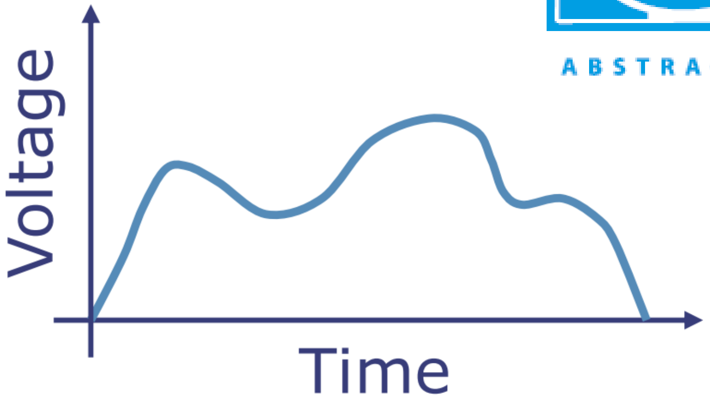
The digital abstraction

"Building Digital Systems in an Analog World"

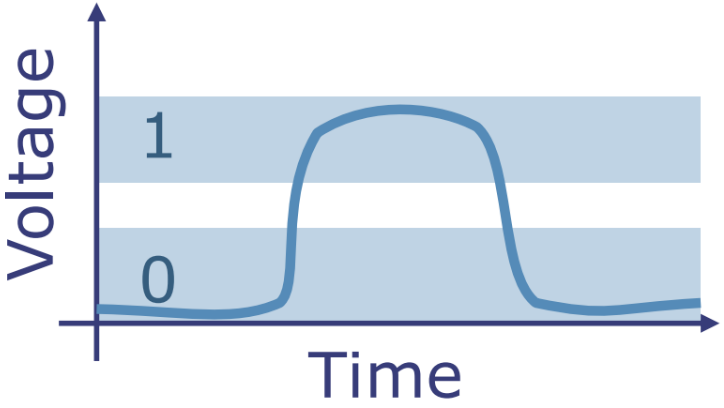
Source: MIT 6.004 2019 L05

The digital abstraction

- Electrical signals in the real world is analog
  - Continuous signals in terms of voltage, current,



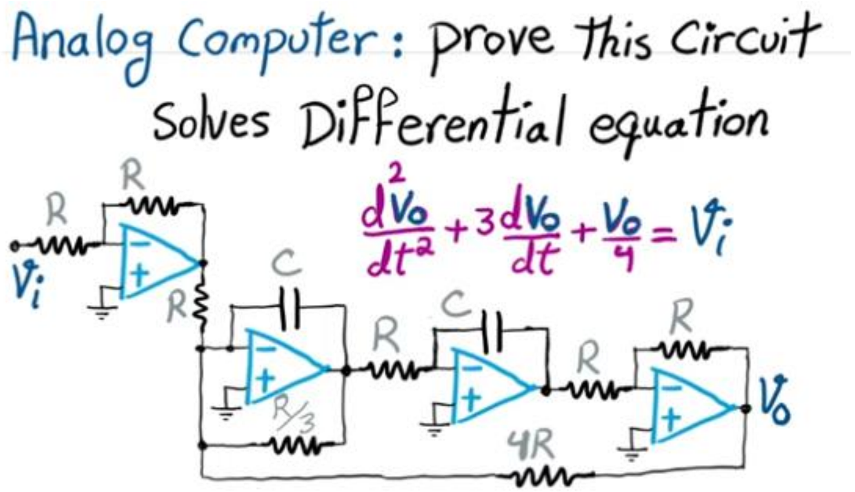
- Modern computers represent and process information using discrete representations
  - Typically binary (bits)
  - Encoded using ranges of physical quantities (typically voltage)



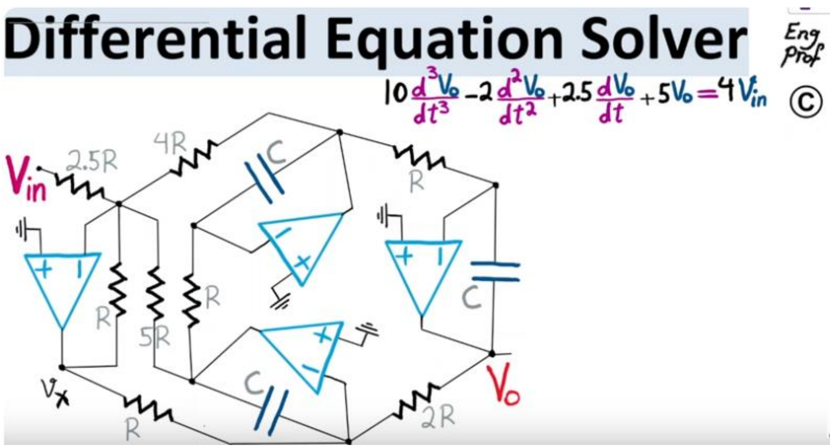
Source: MIT 6.004 2019 L05

Aside: Historical analog computers

- Computers based on analog principles have existed
  - Uses analog characteristics of capacitors, inductors, resistors, etc to model complex mathematical formulas
    - Very fast differential equation solutions!
    - Example: Solving circuit simulation would be very easy if we had the circuit and was measuring it



Engineering Prof., “Analog Computer with Op Amp to solve differential equation”
https://www.youtube.com/watch?app=desktop&v=HeZRtnRXpEI



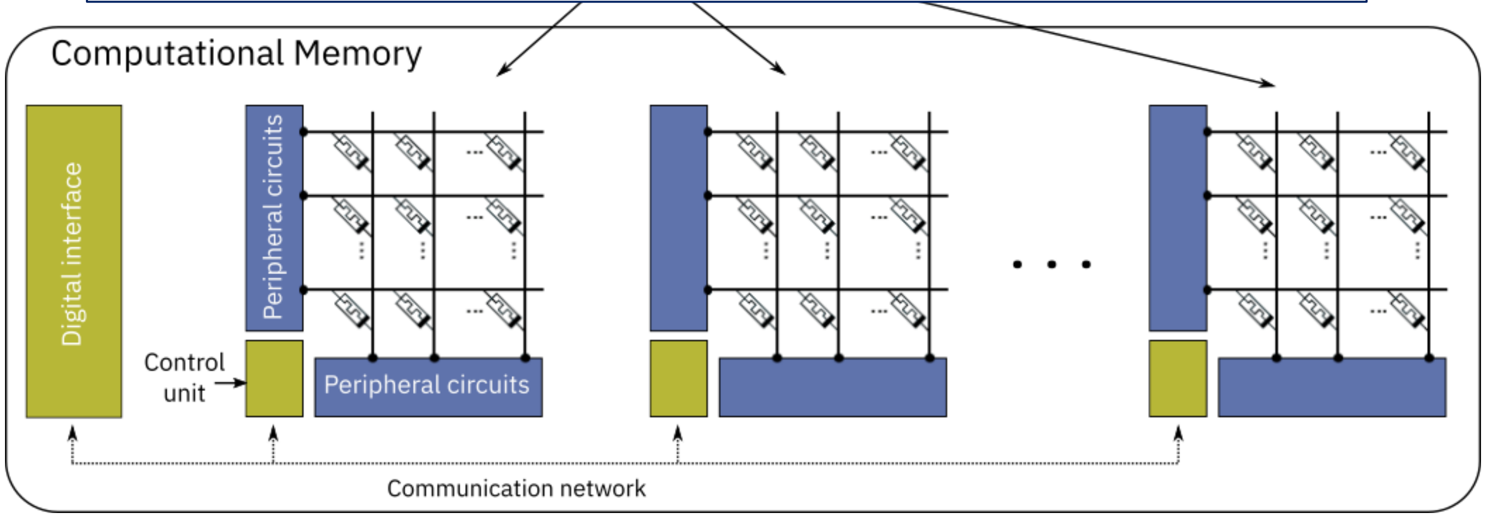
Engineering Prof., “Differential Equation Solver Circuit (Analog Computer)”
https://www.youtube.com/watch?app=desktop&v=HeZRtnRXpEI

Polish analog computer AKAT-1 (1959)
Source: Topory

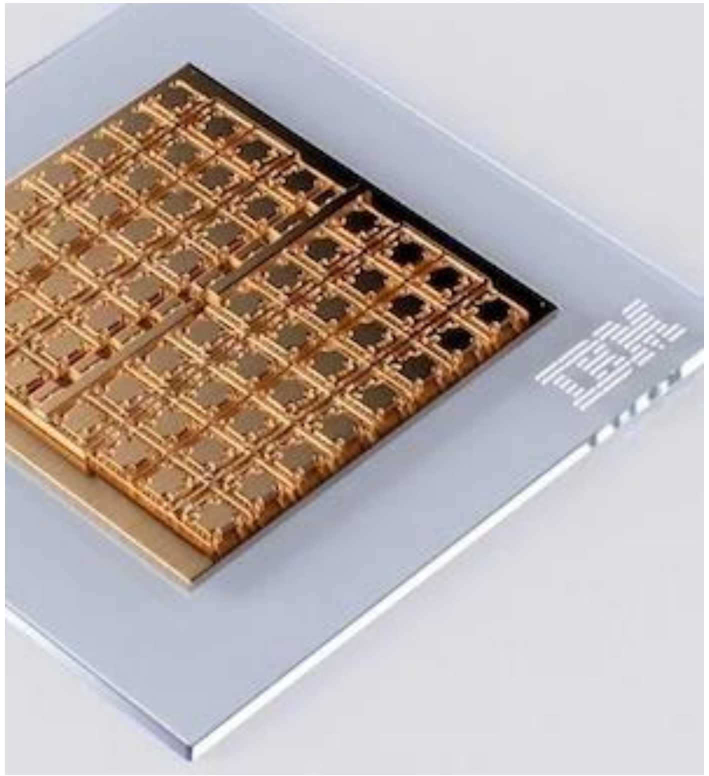
Aside: Historical analog computers

- Some modern resurgence as well!
  - Research on sub-modules performing fast non-linear computation using analog circuitry

"A neural network layer can be implemented on (at least) one crossbar, in which the weights of that layer are stored in the charge or conductance state of the memory devices at the crosspoints. "



IBM Analog Hardware Acceleration Kit, "What is analog AI and an analog chip?"



"The IBM AI analog chip includes 64 analog tiles, each of which can be used as a layer in a neural network"

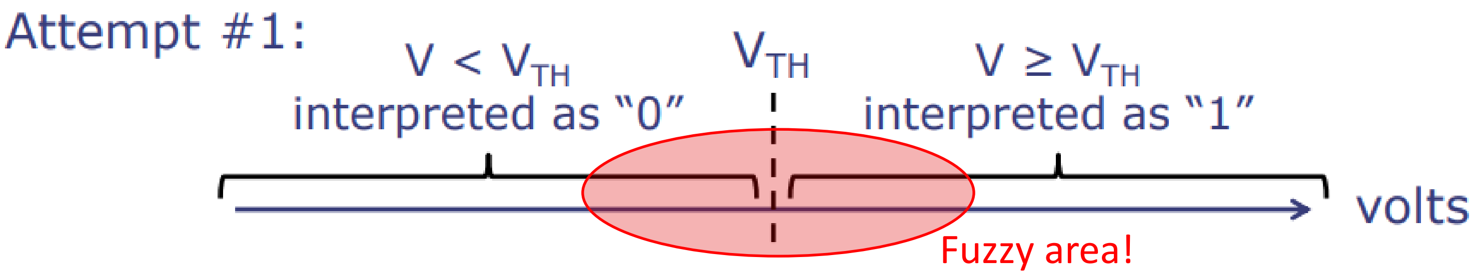
Aside: Historical analog computers

Why are digital systems desirable?

Emphasis: NOISE!!

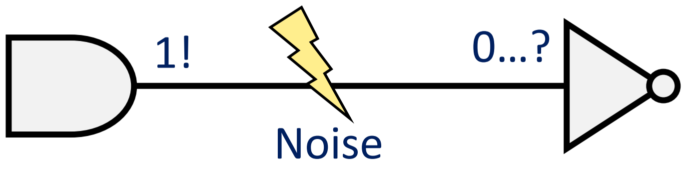
Using voltage digitally

- Key idea
  - Encode two symbols, “0” and “1” (1 bit) in an analog space
  - And use the same convention for every component and wire in system



Problem: There is always noise between transmitter and receiver

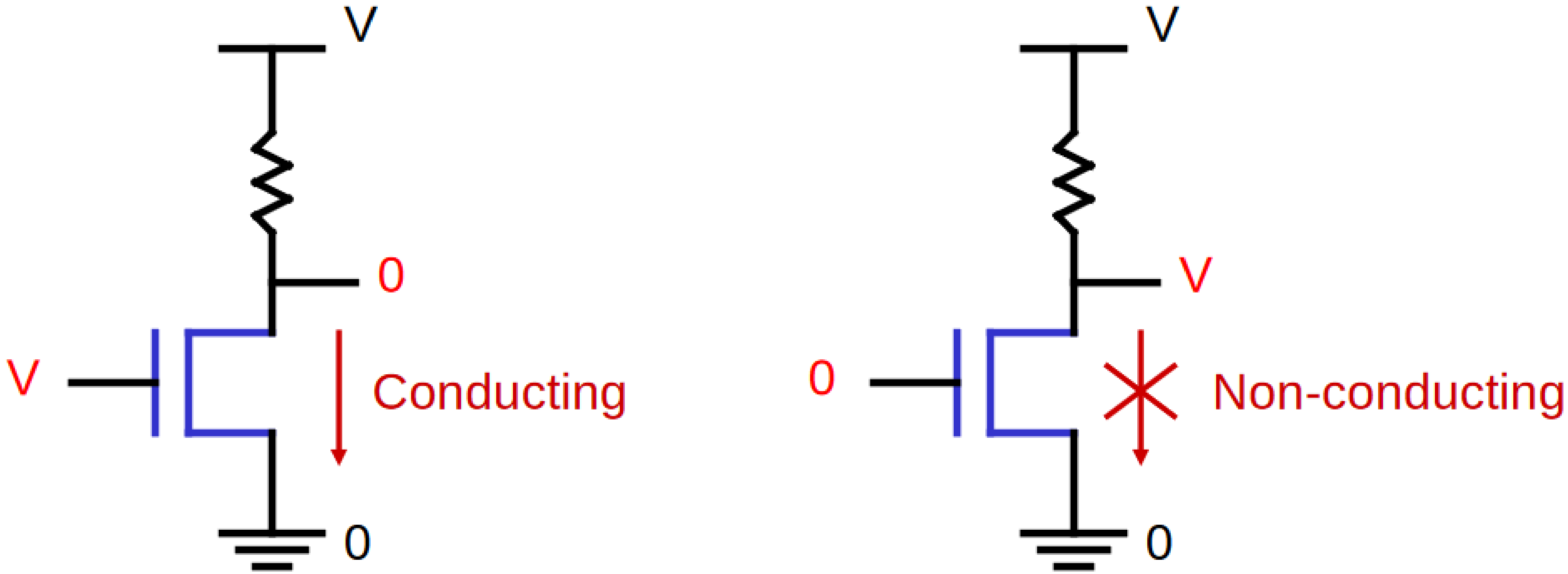
Also, noise can accumulate as we pass through more gates



Source: MIT 6.004 2019 L05

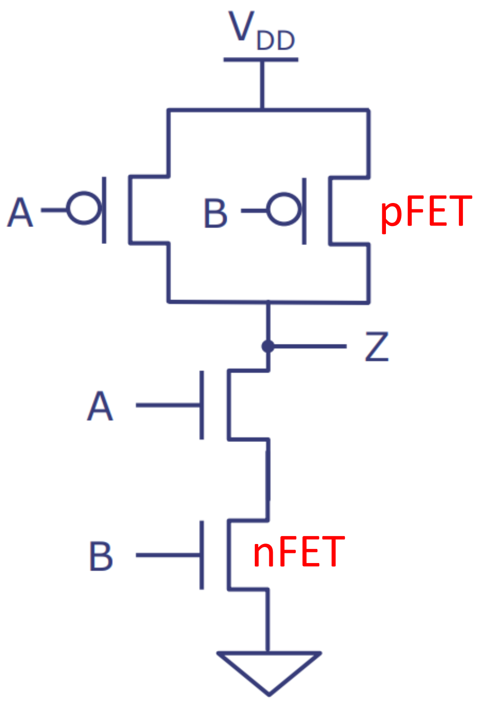
Building block of digital design: Transistors

- A 3-terminal design which works as a switch



Building block of digital design: Transistors

- Composed to create digital logic



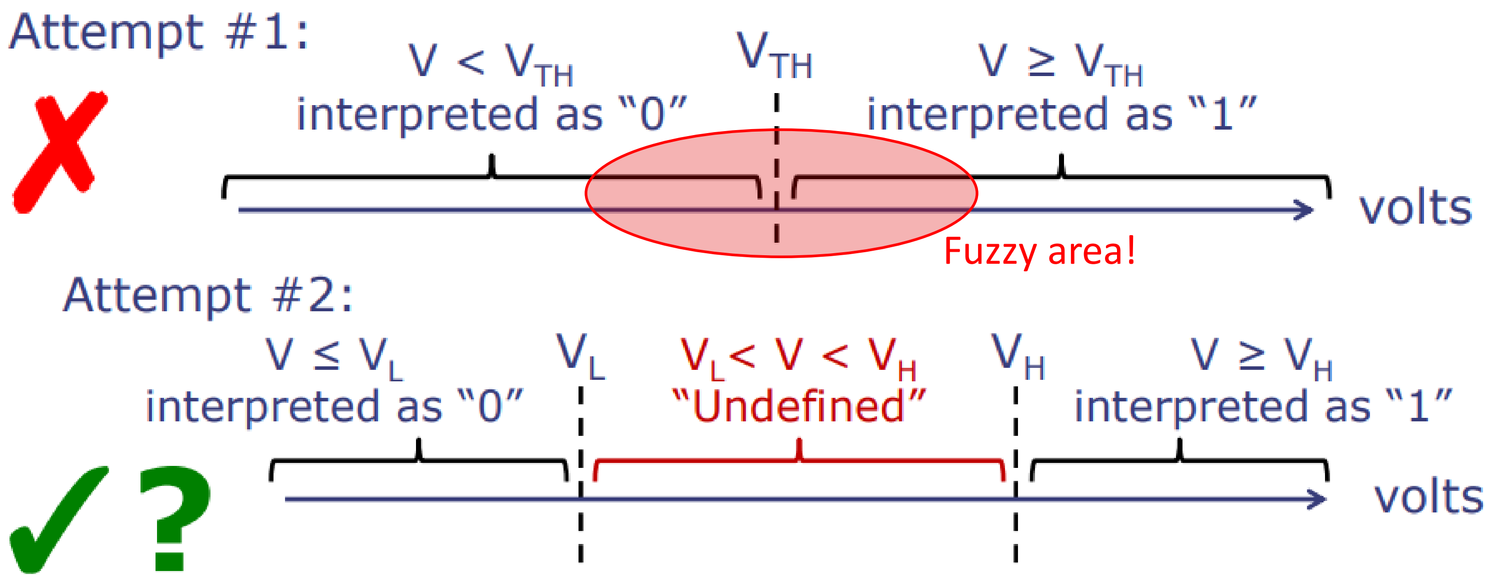
A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

CMOS NAND Gate

Source: MIT 6.004 2019 L09

Using voltage digitally

- Key idea
  - Encode two symbols, “0” and “1” (1 bit) in an analog space
  - And use the same convention for every component and wire in system

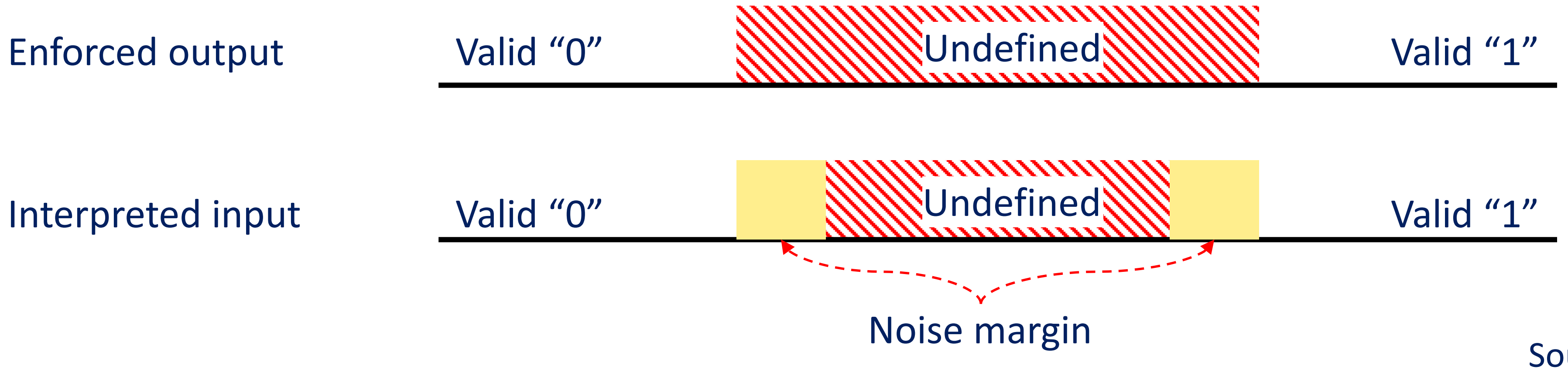


V_L and V_H of output are enforced
during component design and manufacture

Source: MIT 6.004 2019 L05

Handling noise

- When a signal travels between two entities, **there will be noise**
  - Temperature, electromagnetic fields, interaction with surrounding modules, …
- What if V_{out} is barely lower than V_L, or barely higher than V_H?
  - Noise may push the signal into invalid range
  - Rest of the system runs into undefined state!
- Solution: Output signals use a stricter range than input

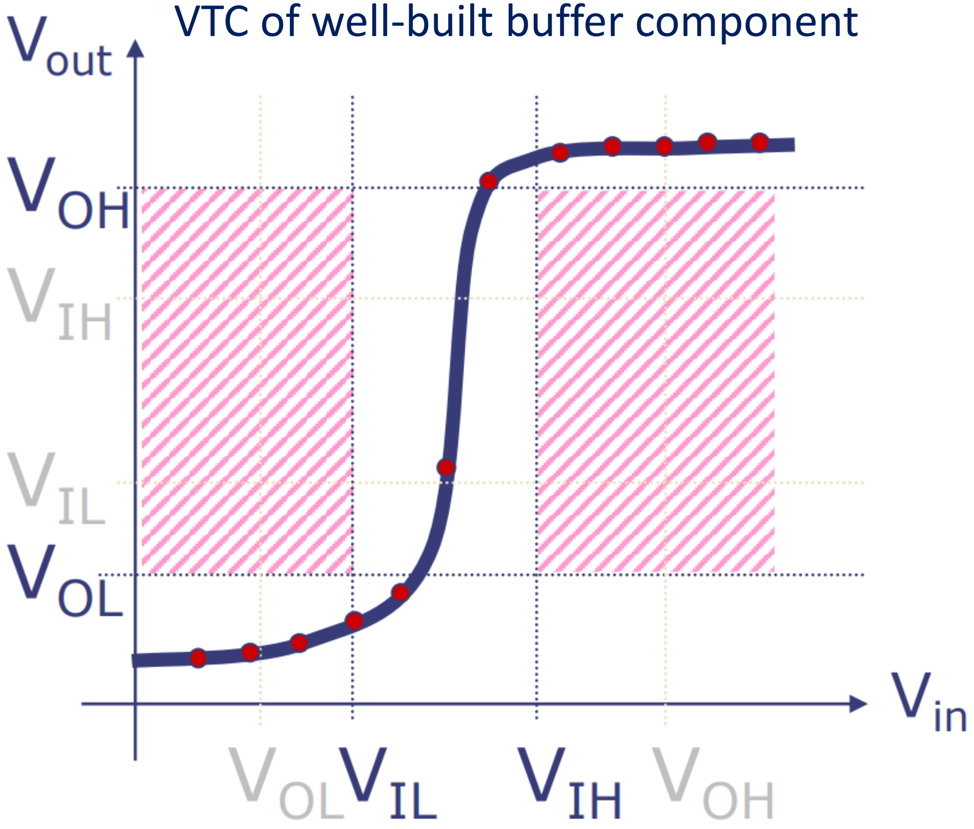


Source: MIT 6.004 2019 L05

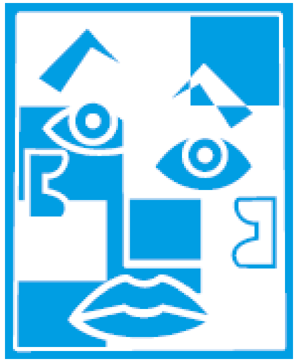
Voltage Transfer Characteristic

- Example component: Buffer
  - A simple digital device that copies its input value to its output
- Voltage Transfer Characteristic (VTC):
  - Plot of V_{out} vs. V_{in} where each measurement is taken after any transients have died out.
  - Not a measure of circuit speed!
    - Only determines behavior under static input
- Each component generates a new, "clean" signal!
  - Noise from previous component corrected

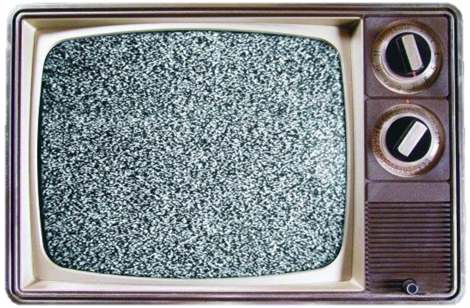
VTC of well-built buffer component



Benefits of digital systems



- Digital components are “restorative”
  - Noise is cancelled at each digital component
  - Very complex designs can be constructed on the abstraction of digital behavior
- Compare to analog components
  - Noise is accumulated at each component
  - Lay example: Analog television signals! (Before 2000s)
    - Limitation in range, resolution due to transmission noise and noise accumulation
    - Contrary: digital signals use repeaters and buffers to maintain clean signals



Source: “Does TV static have anything to do with the Big Bang?” How it works, 2012

CS250P: Computer Systems Architecture
Digital Circuit Design Recap

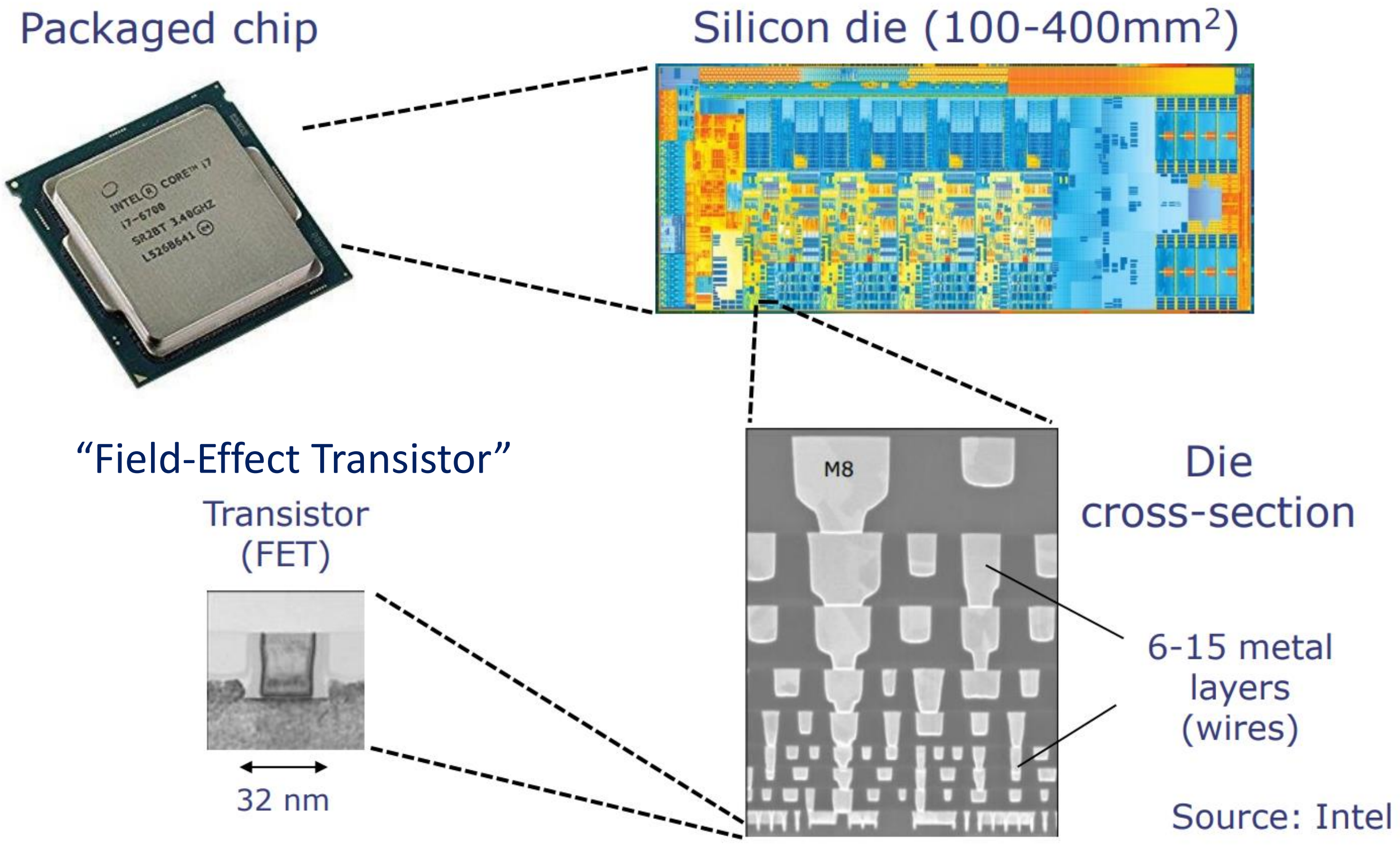
Sang-Woo Jun

Fall 2023

UCI

Large amount of material adapted from MIT 6.004, “Computation Structures”,
Morgan Kaufmann “Computer Organization and Design: The Hardware/Software Interface: RISC-V Edition”,
and CS 152 Slides by Isaac Scherson

The basic building block: CMOS transistors (“Complementary Metal–Oxide–Semiconductor”)

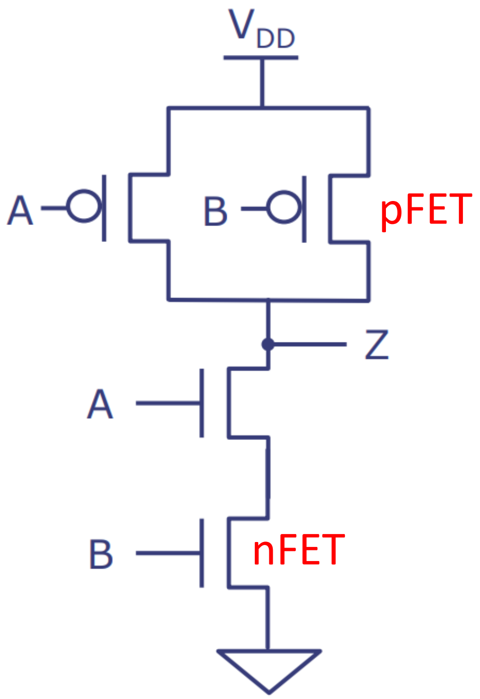


Everything is built as a network of transistors!

Source: MIT 6.004 2019 L09

The basic building block: CMOS FETs

- Remember CS151 – FETs come in two varieties, and are composed to create Boolean logic

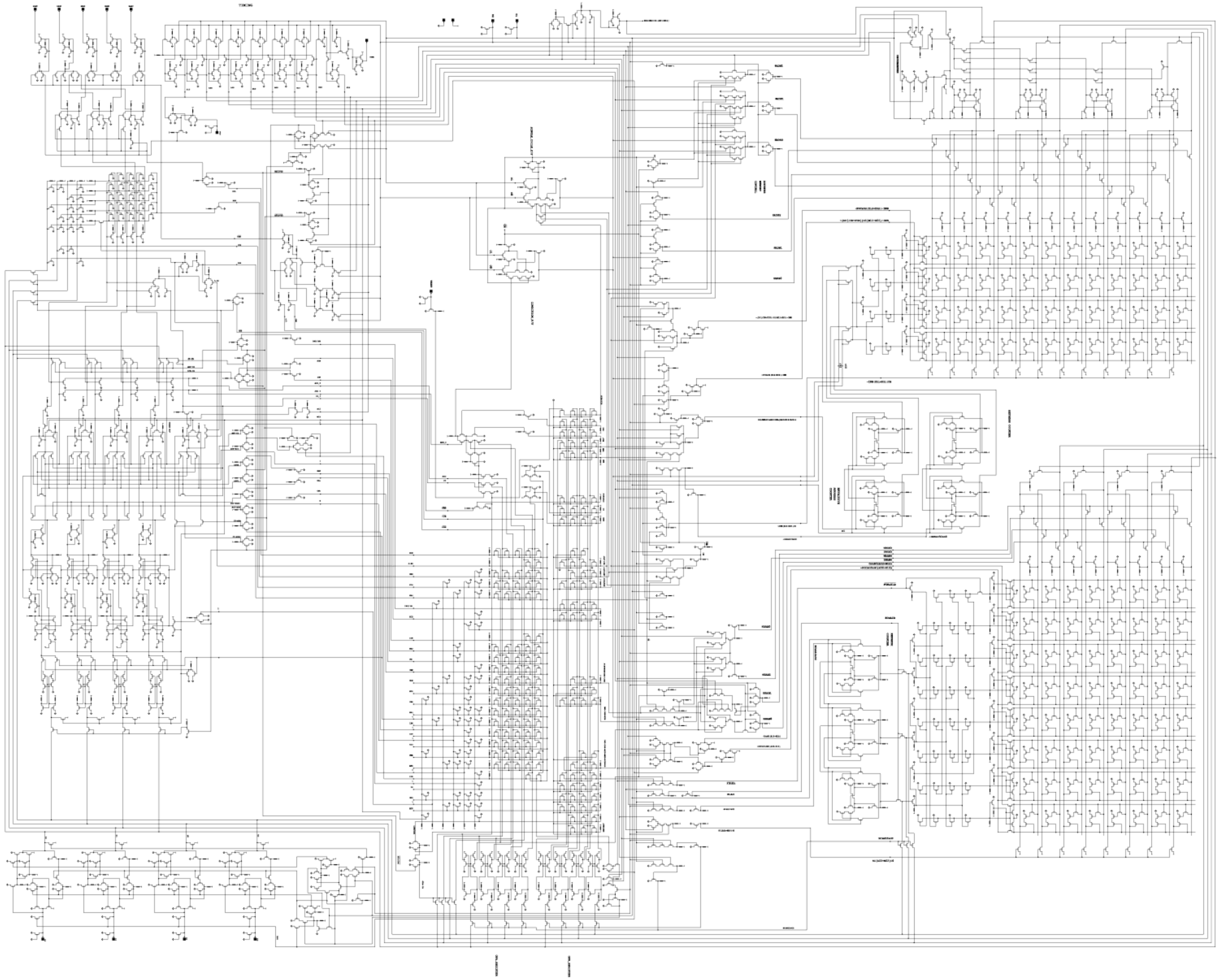


A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

CMOS NAND Gate

Source: MIT 6.004 2019 L09

Making chips out of transistors…?



Intel 4004 Schematics
drawn by Lajos Kintli and Fred Huettig
for the Intel 4004 50th anniversary project

The basic building block 2: Standard cell library

- Standard cell
  - Group of transistor and interconnect structures that provides a boolean logic function
    - Inverter, buffer, AND, OR, XOR, …
  - For a specific implementation technology/vendor/etc…
  - Also includes physical characteristic information
- Eventually, chips designs are expressed as a group of standard cells networked via wires
  - Among what is sent to a fab plant

Example:

Gate	Delay (ps)	Area (μ^2)
Inverter	20	10
Buffer	40	20
AND2	50	25
NAND2	30	15
OR2	55	26
NOR2	35	16
AND4	90	40
NAND4	70	30
OR4	100	42
NOR4	80	32

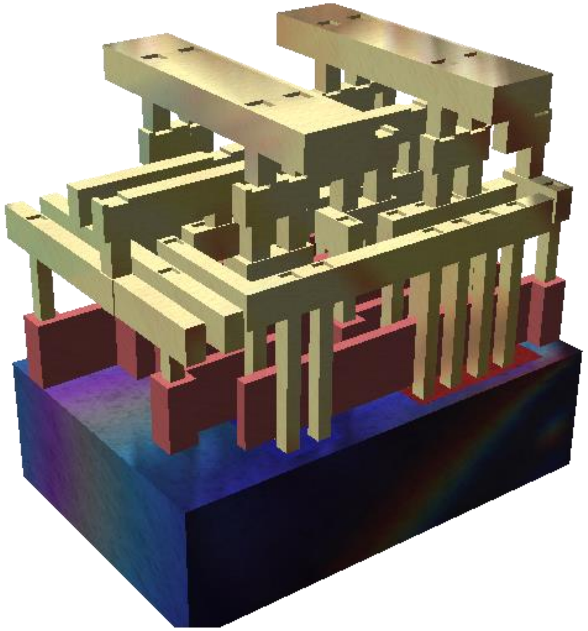
Various components have different delays and area!

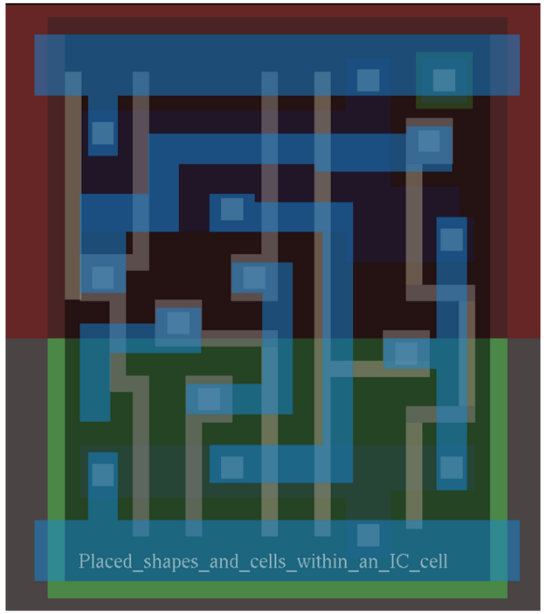
The actual numbers are not important right now

Source: MIT 6.004 2019 L06

Aside: Describing chips for foundries

- GDSII, OASIS file formats
- Depicts many standard cells connected via multiple wire layers





Source: File:Silicon_chip_3d.png, Tgrebinski, File:Wikipediaoasisimage 2.png (Wikipedia)