

Concrete Encoding of Information

To this point we've discussed encoding information using bits. But where do bits come from?

If we're going to design a machine that manipulates information, how should that information be physically encoded?

What makes a good bit?

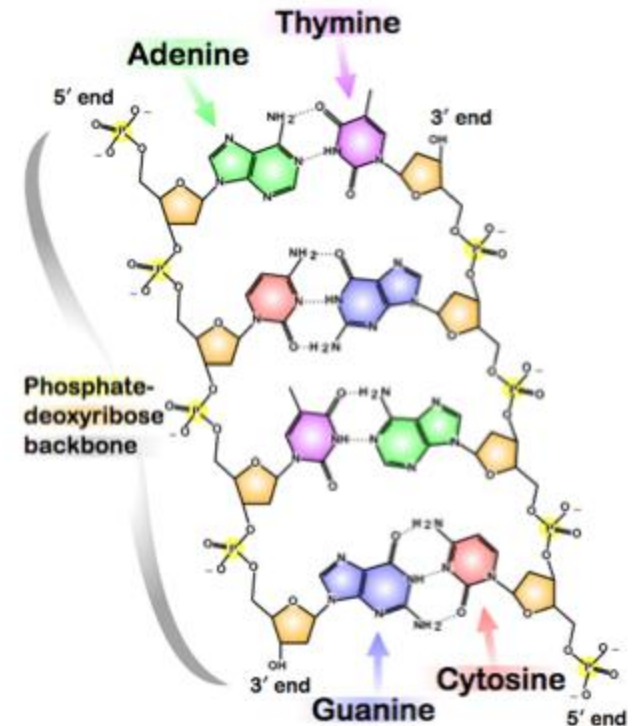
- small, inexpensive (we want a lot of them)
- stable (reliable, repeatable)
- ease and speed of manipulation
(access, transform, combine, transmit, store)

Rosetta Stone



Captmondo (CC BY-SA 3.0)

DNA



Madeleine Price Ball (CC BY-SA 3.0)

Let's Use Electrical Phenomenon...

Consider using phenomenon associated with charged particles:

voltages	phase
currents	frequency

In this course we'll use **voltages** to encode information. But the best choice depends on the intended application...

Voltage pros:

- easy generation, detection
- lots of engineering knowledge
- potentially ~~low~~ power in steady state
zero

Voltage cons:

- easily affected by environment
- DC connectivity required?
- R & C effects slow things down



Representing Information with Voltage

Representation of each (x,y) point on a B&W image:

0 volts: BLACK
1 volt: WHITE
0.37 volts: 37% Gray



John Phelan (CC BY 3.0)

How much information at each point?

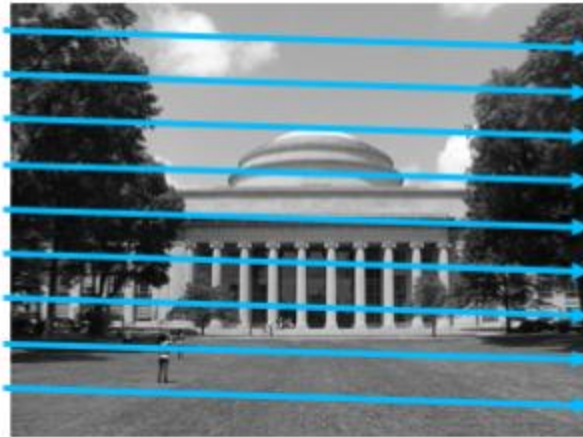
Suppose we can reliably distinguish voltages that differ by $1/2^N$ volts. Then we can represent N bits of information by voltages in the range 0V to 1V. What are realistic values for N ?

Using Voltages to Encode a Picture

Representation of a picture:

Scan points in some prescribed raster order...

Generate voltage waveform:

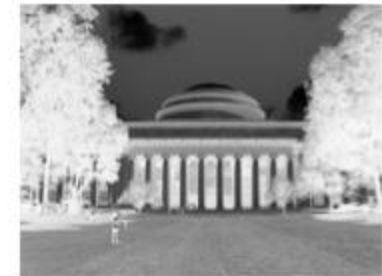


John Phelan (CC BY 3.0)

NTSC
TV
signal



Information Processing = Computation



Why have processing blocks?

- Pre-packaged functionality: rely on behavior without having to be an analog engineer
- Predictable **composition** of functions
→ Tinker-toy assembly
- Guaranteed behavior:
if components work, system will work!

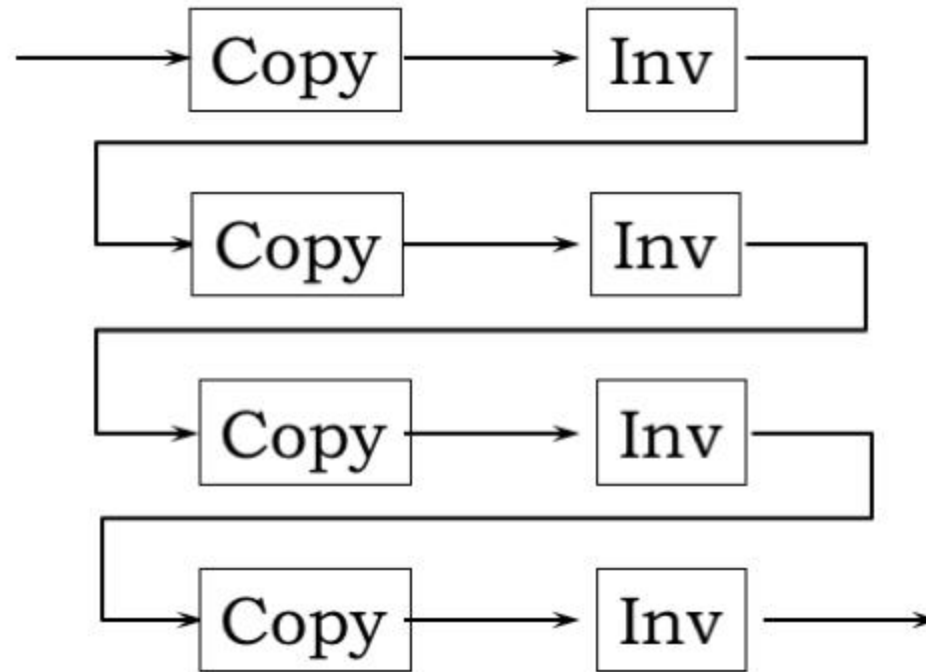


Wow, rules simple enough for a programmer to follow!

Let's Build a System!



input



(Reality)



output

Why Did Our System Fail?

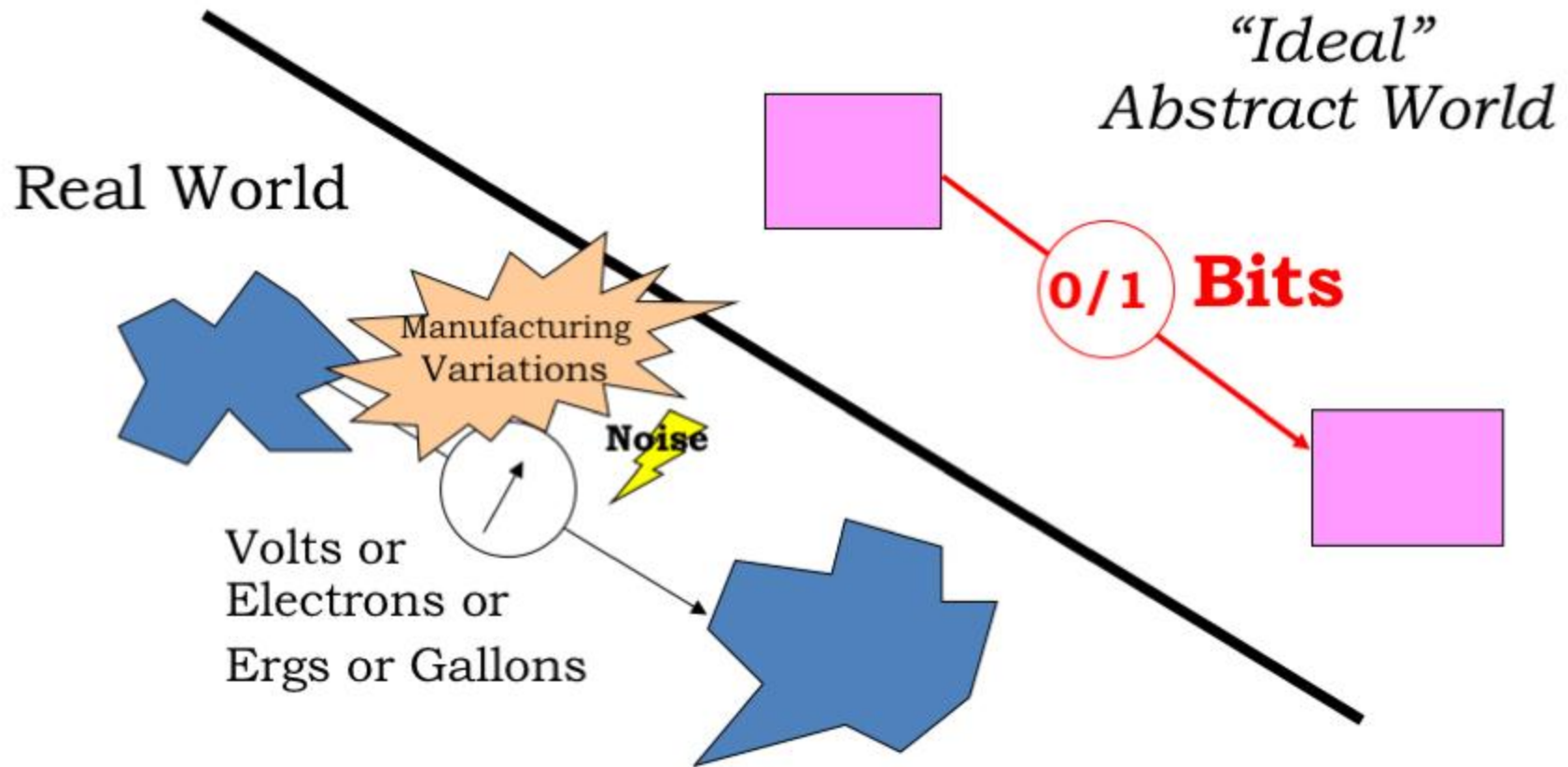
Why doesn't theory match reality?

1. COPY block doesn't work right
2. INV block doesn't work right
3. Theory is imperfect
4. Reality is imperfect
5. Our system architecture stinks

ANSWER: all of the above!

Noise and inaccuracy are inevitable; we can't reliably reproduce infinite information – we must **design our system to tolerate some amount of error** if it is to process information reliably.

The **Digital** Abstraction

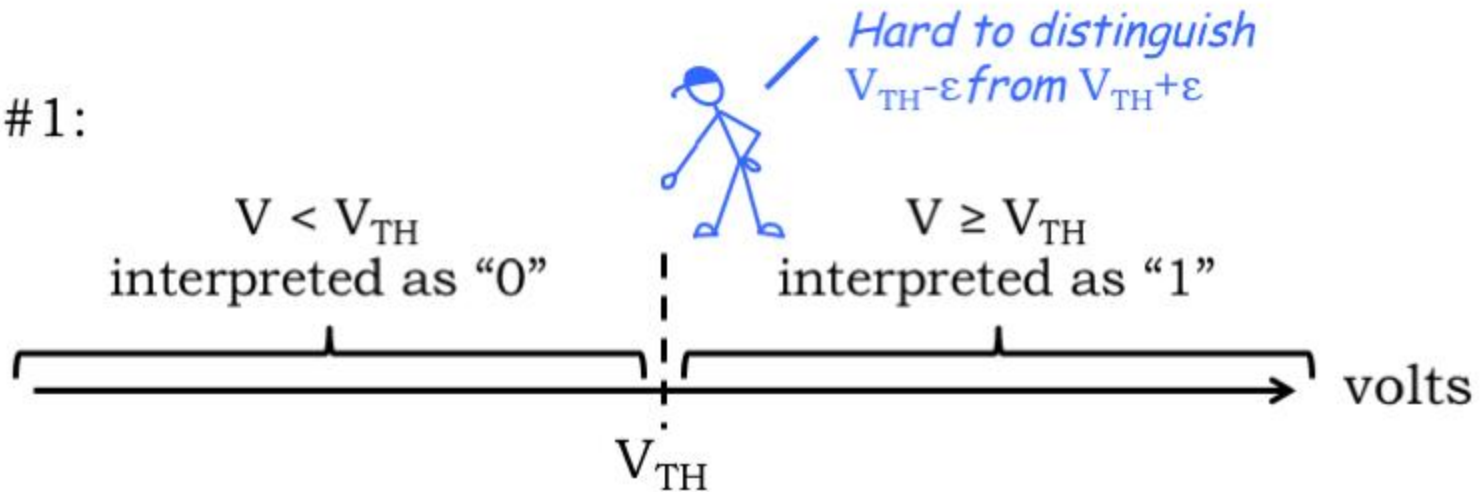


Keep in mind that the world is not digital, we would simply like to engineer it to behave that way. Furthermore, we must use **real physical phenomena** to implement digital designs!

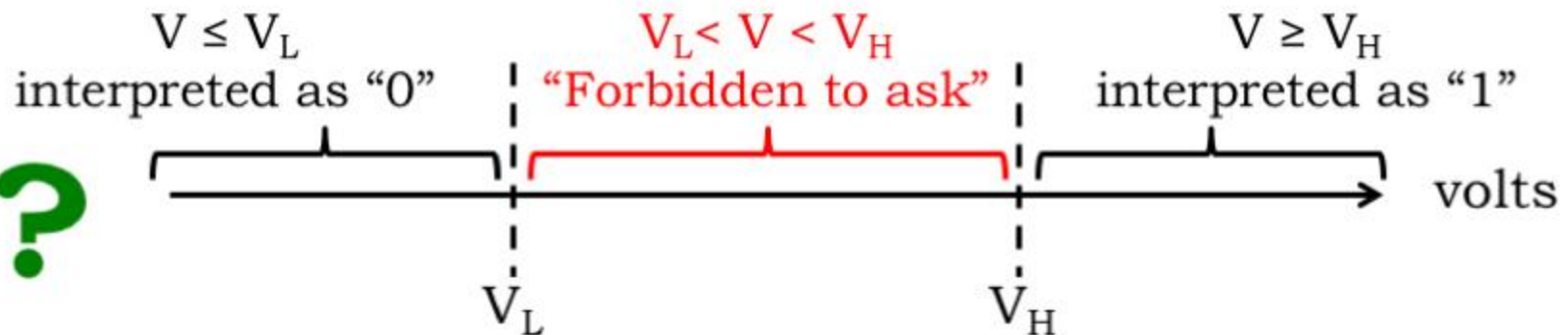
Using Voltages “Digitally”

- Key idea: encode only one bit of information: 2 values “0”, “1”
- Use the same uniform representation convention for *every* component and wire in our digital system

Attempt #1:



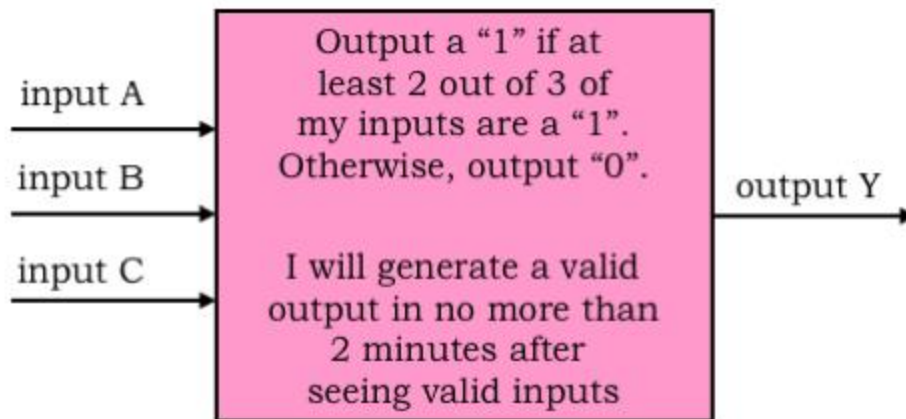
Attempt #2:



A Digital Processing Element

A combinational device is a circuit element that has

- Static discipline
- one or more digital *inputs*
 - one or more digital *outputs*
 - a *functional specification* that details the value of each output for every possible combination of valid input values
 - a *timing specification* consisting (at minimum) of an upper bound t_{PD} on the required time for the device to compute the specified output values from an arbitrary set of stable, valid input values



A Combinational Digital System

A set of interconnected elements is a combinational device if

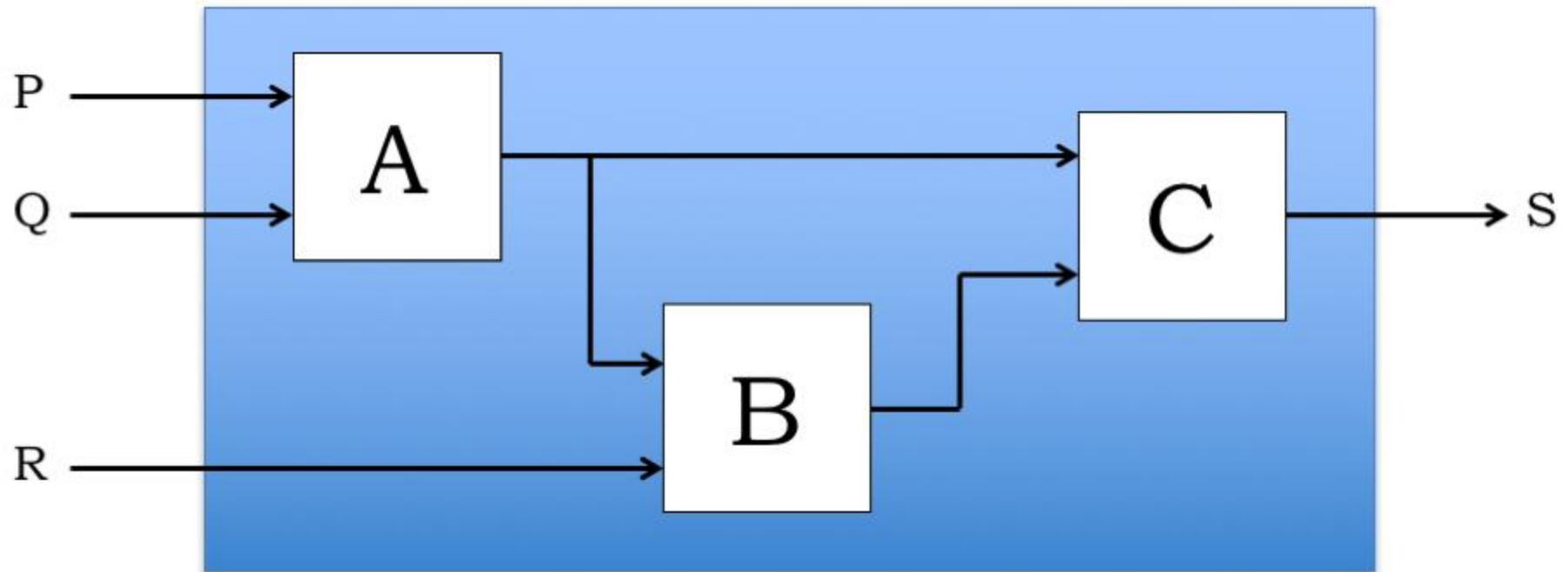
- each circuit element is combinational
- every input is connected to exactly one output or to some vast supply of constant 0's and 1's
- the circuit contains no directed cycles



Why is this true?

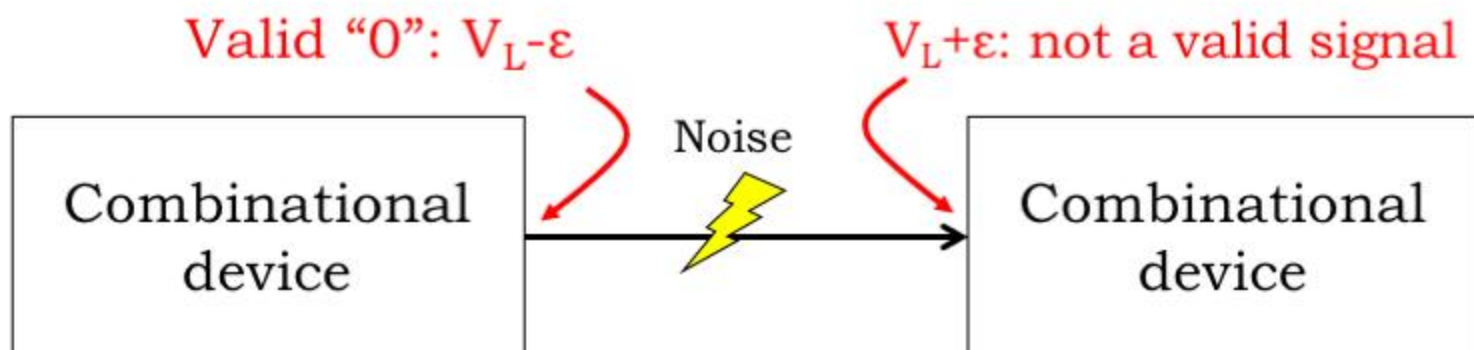
Is This a Combinational Device?

A, B and C are combinational devices. Is the following circuit a combinational device?



- Does it have digital inputs?
- Does it have digital outputs?
- Can you derive a functional description?
- Can you derive a t_{PD} ?

Will This System Work?



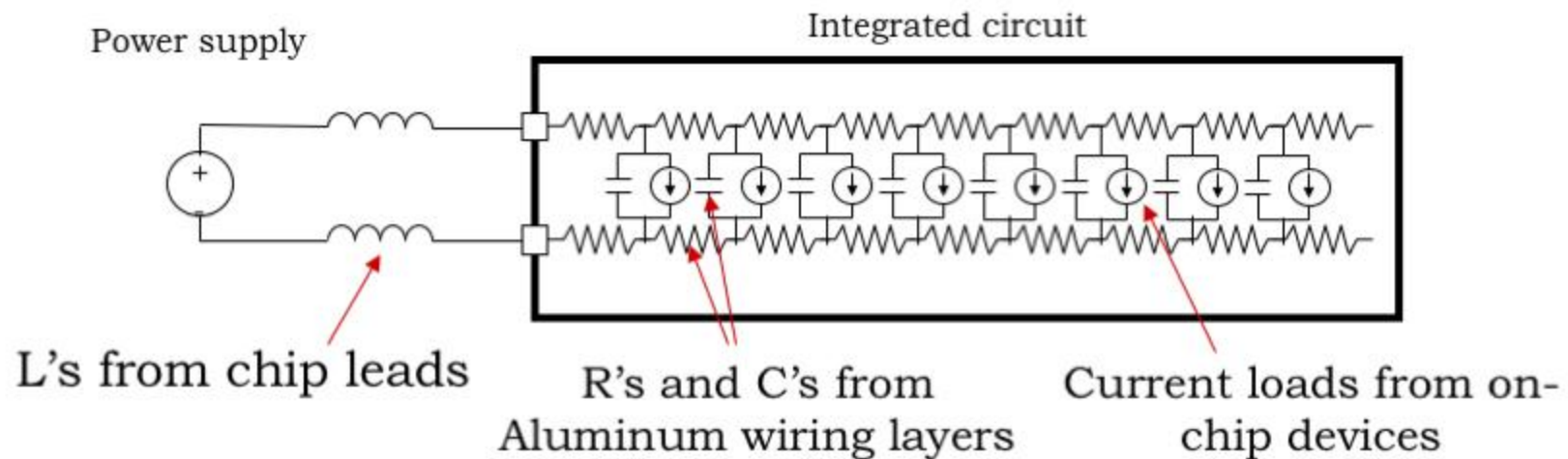
Upstream device transmits a signal at $V_L - \epsilon$, a valid "0". Noise on the connecting wire causes the downstream device to receive $V_L + \epsilon$, a signal in the forbidden zone.



Hmm. Looks like the output voltage needs to be adjusted so that a signal will still be valid when it reaches an input even if there's noise.

Where Does Noise Come From?

- Parasitic resistance, inductance, capacitance
 - IR drop, $L(di/dt)$ drop, LC ringing from current steps

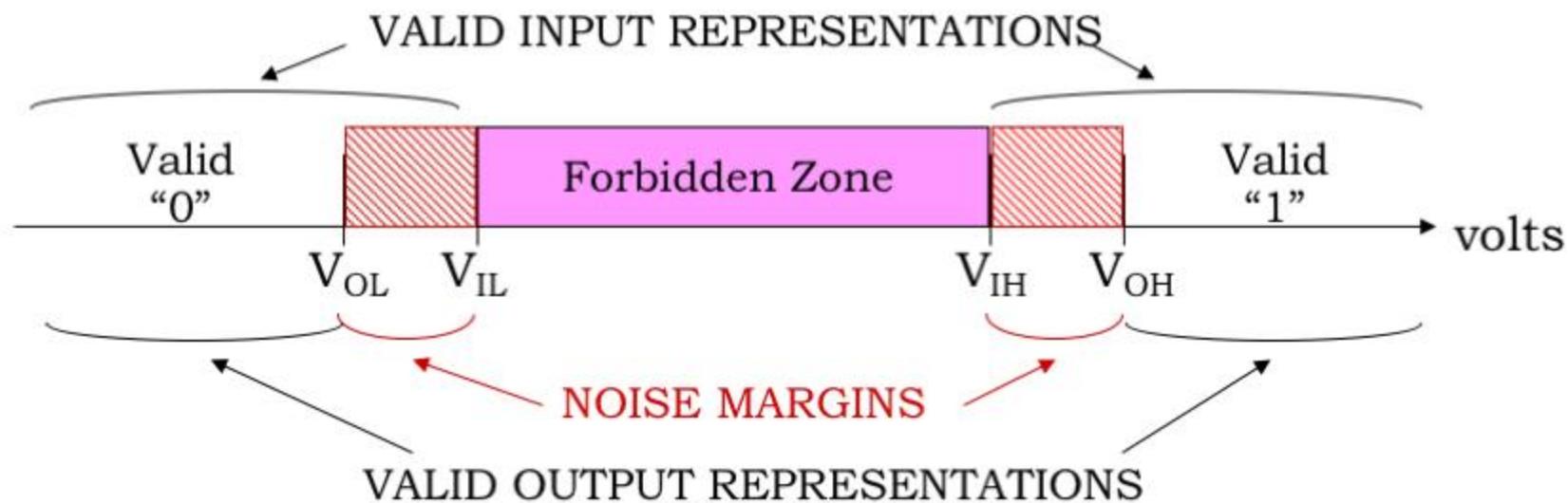


- Imprecision of component values
 - Manufacturing variations, allowable tolerances
- Environmental effects
 - External EM fields, temperature variations, etc.
- ...

Needed: Noise Margins!

Proposed fix: separate specifications for inputs and outputs

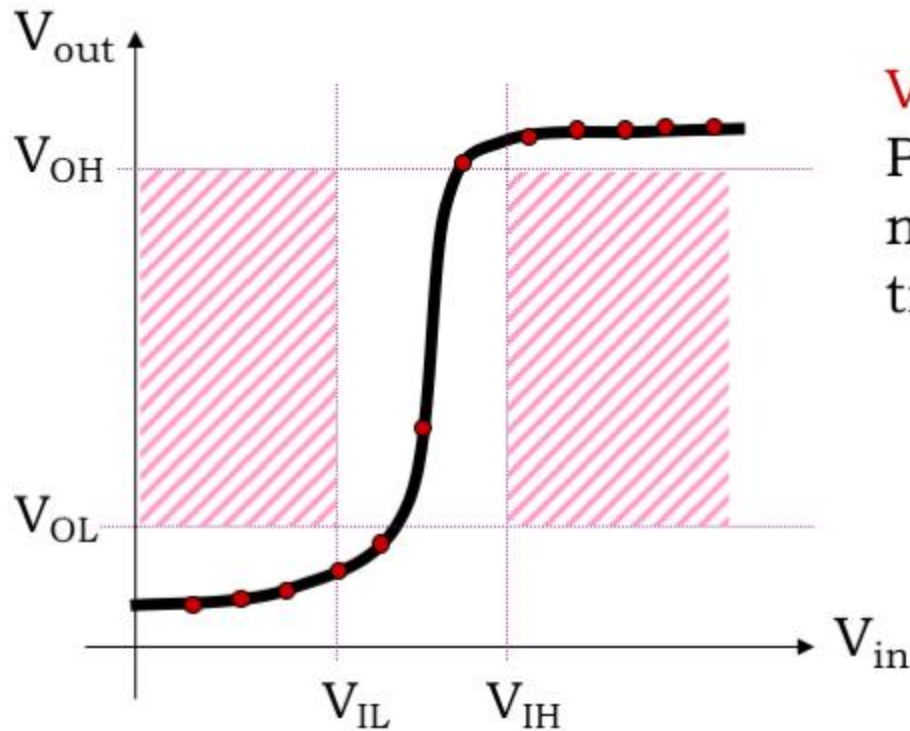
- digital output: “0” $\leq V_{OL}$, “1” $\geq V_{OH}$
- digital input: “0” $\leq V_{IL}$, “1” $\geq V_{IH}$
- $V_{OL} < V_{IL} < V_{IH} < V_{OH}$



A combinational device accepts marginal inputs and provides unquestionable outputs (to leave room for noise).

A Buffer

A simple *combinational device*: $0 \rightarrow 0$ $1 \rightarrow 1$

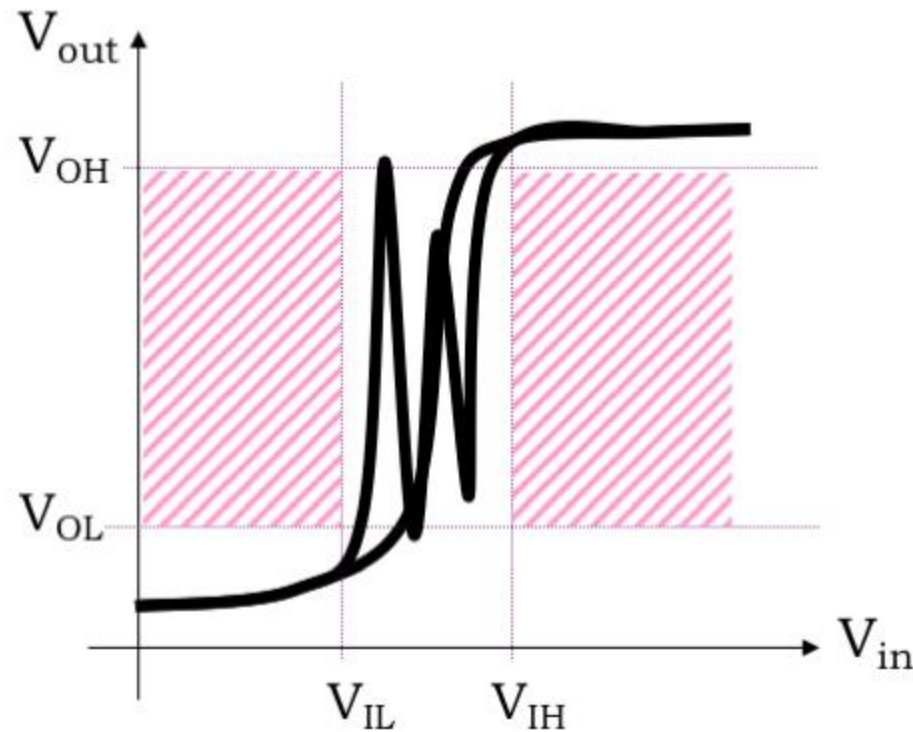


Voltage Transfer Characteristic (VTC):
Plot of V_{out} vs. V_{in} where each measurement is taken after any transients have died out.

Note: VTC does not tell you anything about how fast a device is — it measures static behavior not dynamic behavior

Static Discipline requires that the VTC avoid the shaded regions (aka “*forbidden zones*”), which correspond to *valid* inputs but *invalid* outputs.

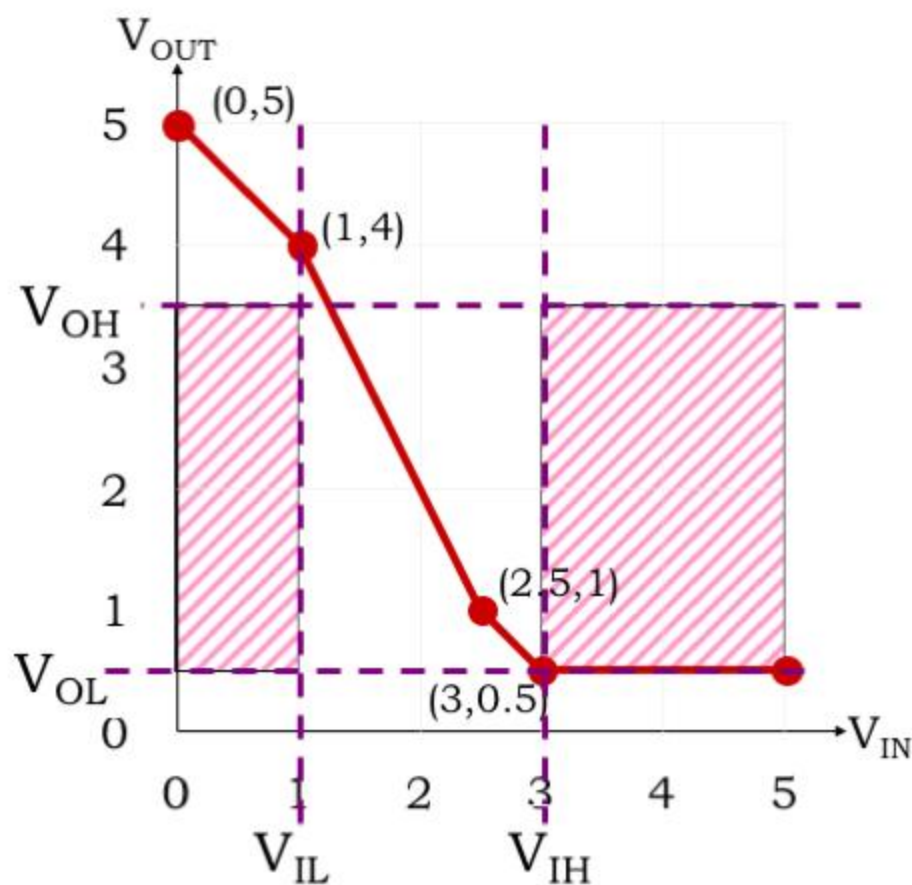
Voltage Transfer Characteristic



- 1) Note the VTC can do anything when $V_{IL} < V_{IN} < V_{IH}$.
- 2) Note that the center white region is taller than it is wide ($V_{OH} - V_{OL} > V_{IH} - V_{IL}$). Net result: combinational devices must have **GAIN > 1** and be **NONLINEAR**.

Can This Be a Combinational Inverter?

Suppose that you measured the voltage transfer curve of the device shown below. Can we find a signaling specification that would allow this device to be a combinational inverter?



The device must be able to actually produce the desired output level. Thus, V_{OL} can be no lower than 0.5 V.

$$\text{Try } V_{OL} = 0.5 \text{ V}$$

V_{IH} must be high enough to produce V_{OL}

$$\text{Try } V_{IH} = 3 \text{ V}$$

Now, find noise margin N and compute

$$V_{OH} = V_{IH} + N$$

$$V_{IL} = V_{OL} + N$$

Then verify that $V_{OUT} \geq V_{OH}$ when $V_{IN} \leq V_{IL}$.

$$\text{Try } N = 0.5 \text{ V}$$

Device is a combinational inverter when $V_{OL}=0.5$, $V_{IL}=1$, $V_{IH}=3$, $V_{OH}=3.5$

Summary

- Use voltages to encode information
- “Digital” encoding
 - valid voltage levels for representing “0” and “1”
 - forbidden zone avoids mistaking “0” for “1” and vice versa
 - gives rise to notion of signal VALIDITY.
- Noise
 - Want to tolerate real-world conditions: NOISE.
 - Key: tougher standards for output than for input
 - devices must have gain and have a non-linear VTC
- Combinational devices
 - Each logic family has Tinkertoy-set simplicity, modularity
 - predictable composition: “parts work → whole thing works”
 - static discipline
 - digital inputs, outputs; restore marginal input voltages
 - complete functional spec
 - valid inputs lead to valid outputs in bounded time