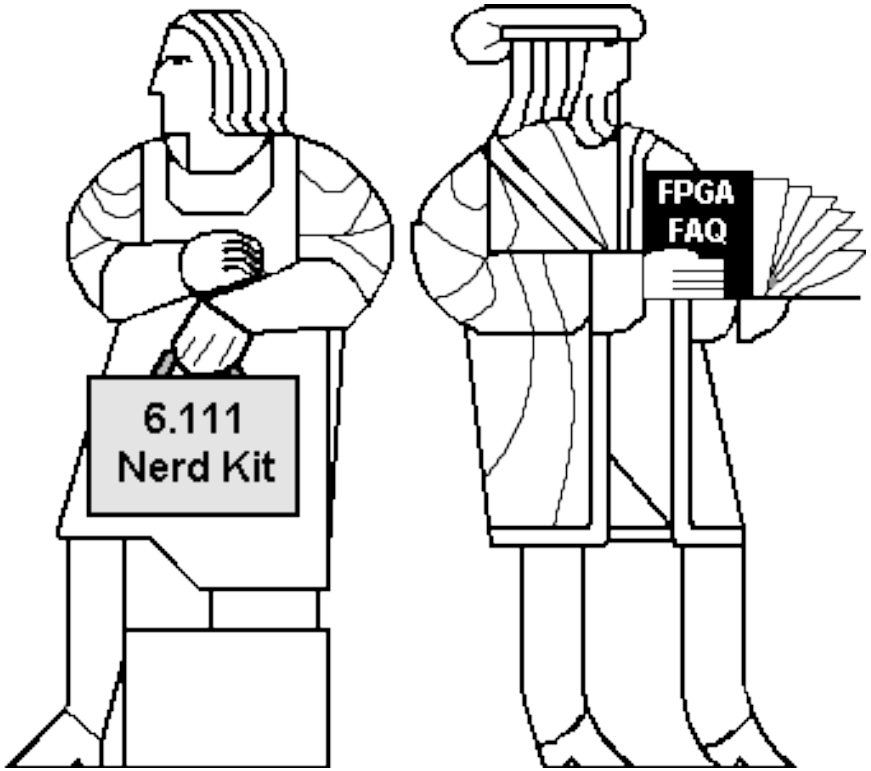
Welcome to 6.111!
Introductory Digital Systems Laboratory



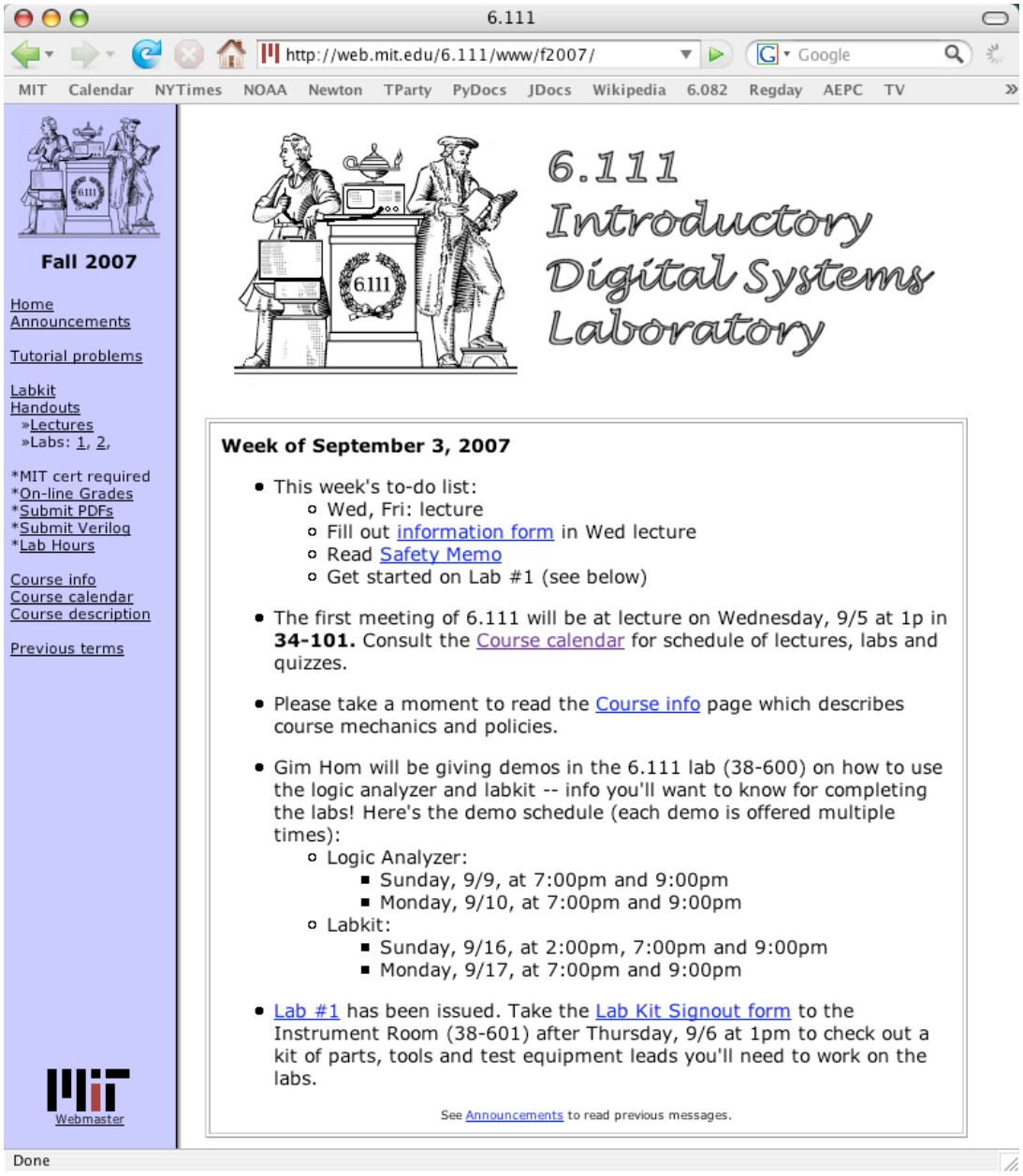
Handouts:
- Info form (yellow)
- Course Calendar
- Lecture slides

Lectures:
- Chris Terman

TAs:
- Roberto Carli
- Brian Schmidt
- Alessandro Yamhure

Lab guru:
- Gim Hom

Course Website (http://mit.edu/6.111)



6.111 Goals

- Fundamentals of logic design
  - combinational and sequential blocks
- System integration with multiple components
  - FPGAs, memories, discrete components, etc.
- Learn a Hardware Description Language (Verilog)
- Interfacing issues with analog components
  - ADC, DAC, sensors, etc.
- Understand different design methodologies
- Understand different design metrics
  - component/gate count and implementation area, switching speed, energy dissipation and power
- Design & implement a substantial digital system
- Have fun!

Labs: learning the ropes

- Lab 1
  - Experiment with gates, design & implement some logic
  - Learn about lab equipment in the Digital Lab (38-600): oscilloscopes and logic analyzers
- Lab 2
  - Introduction to Verilog & the labkit
- Lab 3
  - Design and implement a Finite State Machine (FSM)
  - Use Verilog to program an FPGA
  - Report and its revision will be evaluated for CI-M
- Lab 4
  - Design a complicated system with multiple FSMs (Major/Minor FSM)
  - Voice recorder using AC97 codec and SRAMs
- Lab 5
  - Video circuits: a simple Pong game

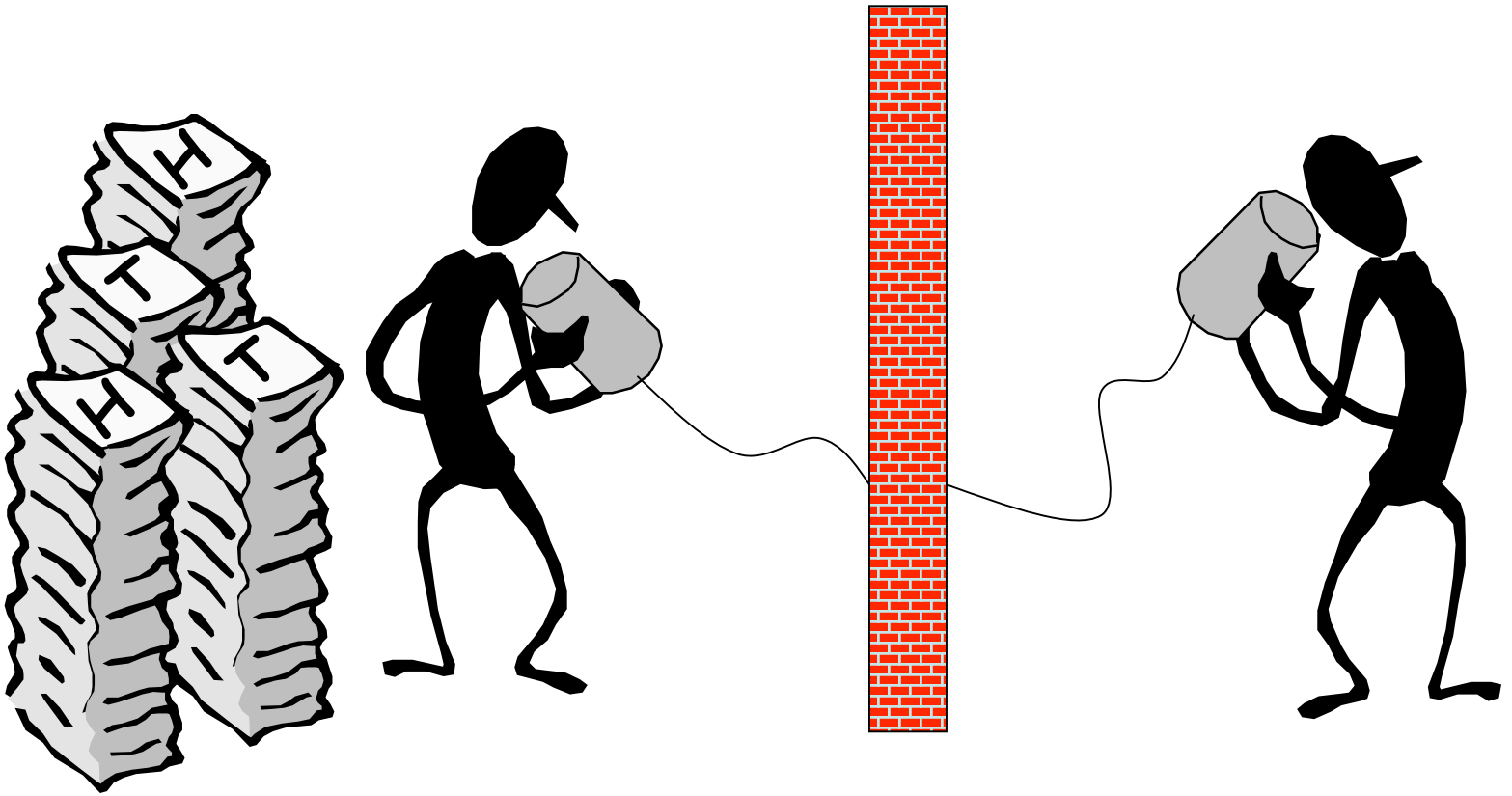
Final Project

- Done in groups of two (or sometimes three)
- Open ended
- You and the staff negotiate a project proposal
  - Must emphasize digital concepts, but inclusion of analog interfaces (e.g., data converters, sensors or motors) common and often desirable
  - Proposal Conference, several Design Reviews
- Design presentation to staff
- Staff will provide help with project definition and scope, design, debugging, and testing
- It is extremely difficult for a student to receive an A without completing the final project

Evaluation

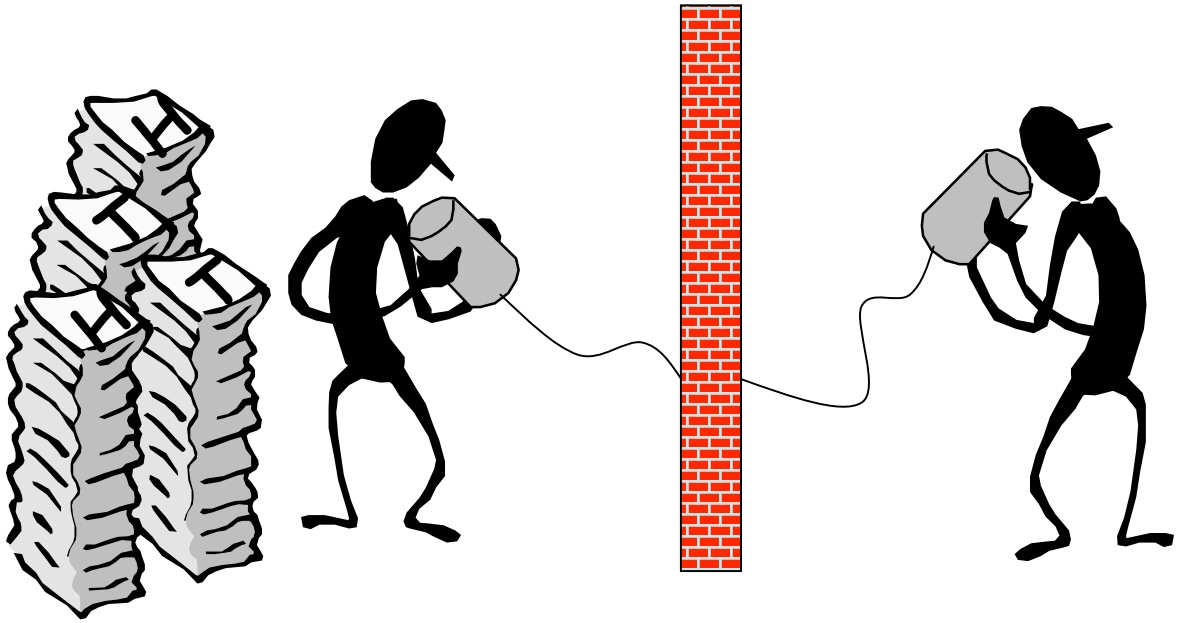
- Midterm (10/31): 20%
- Labs: 30%
  - Labs 1 & 2: 3%, Lab 3: 6%, Labs 4 & 5: 9%
- CI-M paper: 10%
- Final Project: 40%
  - Deadlines and participation: 5%
  - Quality and organization of presentation and report: 5%
  - Complexity, innovation and risk: 10%
  - Problem definition: 2%
  - Architecture: 3%
  - Design (modularity, Verilog): 5%
  - Functionality: 10%
- A large number of students do "A" level work and are, indeed, rewarded with a grade of "A". The corollary to this is that, since average performance levels are so high, punting any part of the subject can lead to a disappointing grade.

Why Digital? A Thought Experiment



Goal: transmit results of 100 coin flips

Experiment #1: Analog Encoding



100 coin flips → 2^{100} possibilities
Transmit voltage $N/2^{100}$ for possibility #N
Required voltage resolution = $1/2^{100}$ = ~8e-31 volts

impossible to reliably transmit/receive voltages with required resolution

Rethink basic system architecture

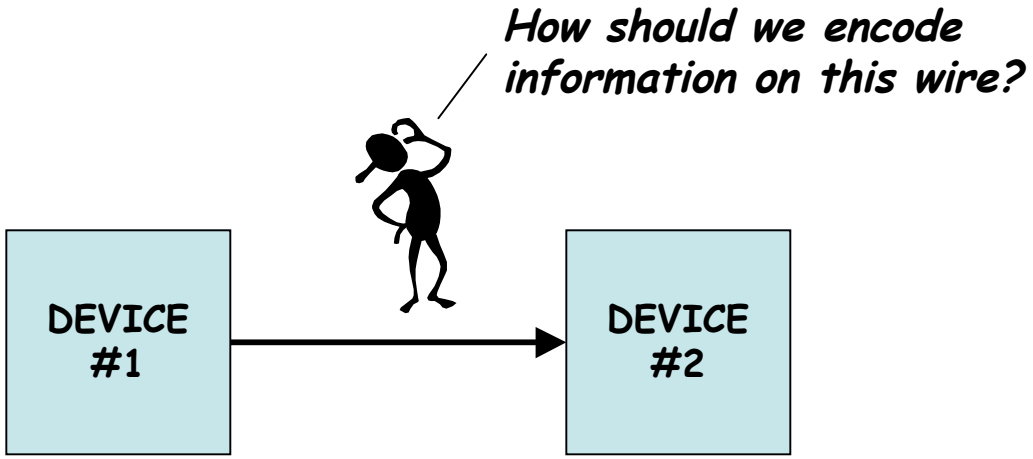
- Noise and inaccuracy are inevitable; we can't reliably transmit/receive/manipulate "infinite" information-- we must design our system to tolerate some amount of error if it is to process information reliably.
- A system is a structure that is guaranteed to exhibit a specified behavior, assuming all of its (imperfect) components obey their specified behaviors.

  CONTRACTS!

  How is this achieved?

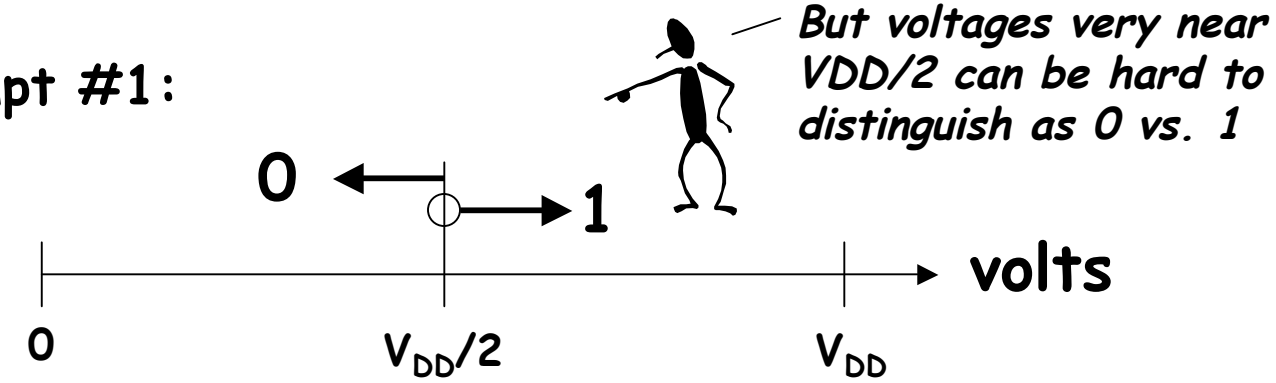
  Every system component will have clear obligations and responsibilities. If contracts are violated all bets are off.

Digital Signaling



We can encode information using voltages, currents, frequency, phase, etc. A common choice is to use a voltage encoding and a binary (0/1) signaling scheme

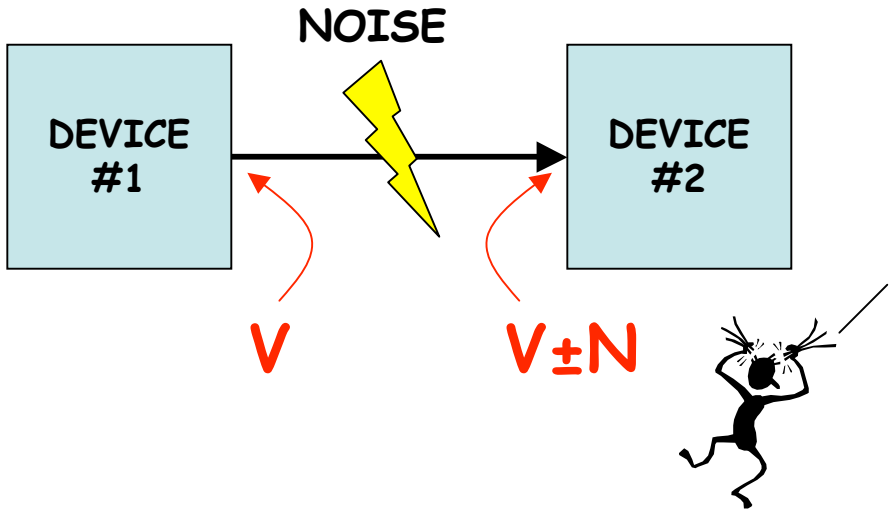
Encoding Attempt #1:



Digital Signaling II

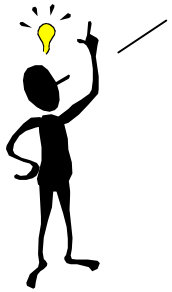
Encoding Attempt #2:

This avoids "close calls", but now we have to consider noise (i.e. unavoidable perturbations to our signaling voltage)

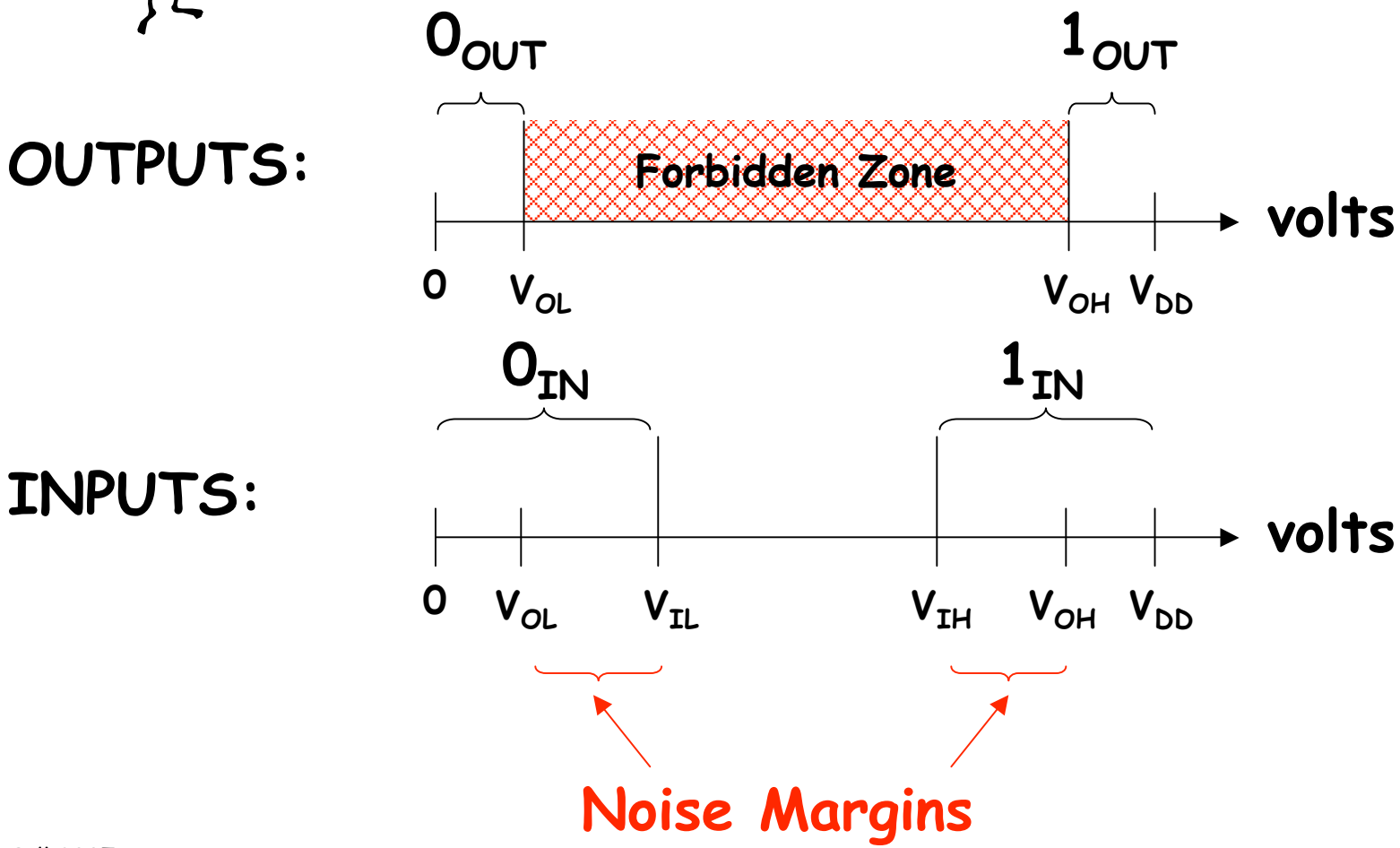


So an output voltage just below V_L might become an <u>illegal</u> input voltage in the forbidden zone!

Big Idea: Noise Margins



Let's leave room for bad things to happen! So we'll design devices restore marginally valid input signals. They must accept marginal inputs and provide unquestionable outputs (i.e., to leave room for noise).



Using Voltages Digitally

Digital input: $V_{IN} < V_{IL}$ or $V_{IN} > V_{IH}$

Digital output: $V_{OUT} < V_{OL}$ or $V_{OUT} > V_{OH}$

Noise margins: $V_{OL} < V_{IL} < V_{IH} < V_{OH}$

Where V_{OL}, V_{IL}, V_{IH} and V_{OH} are part of the specification for a particular family of digital components.

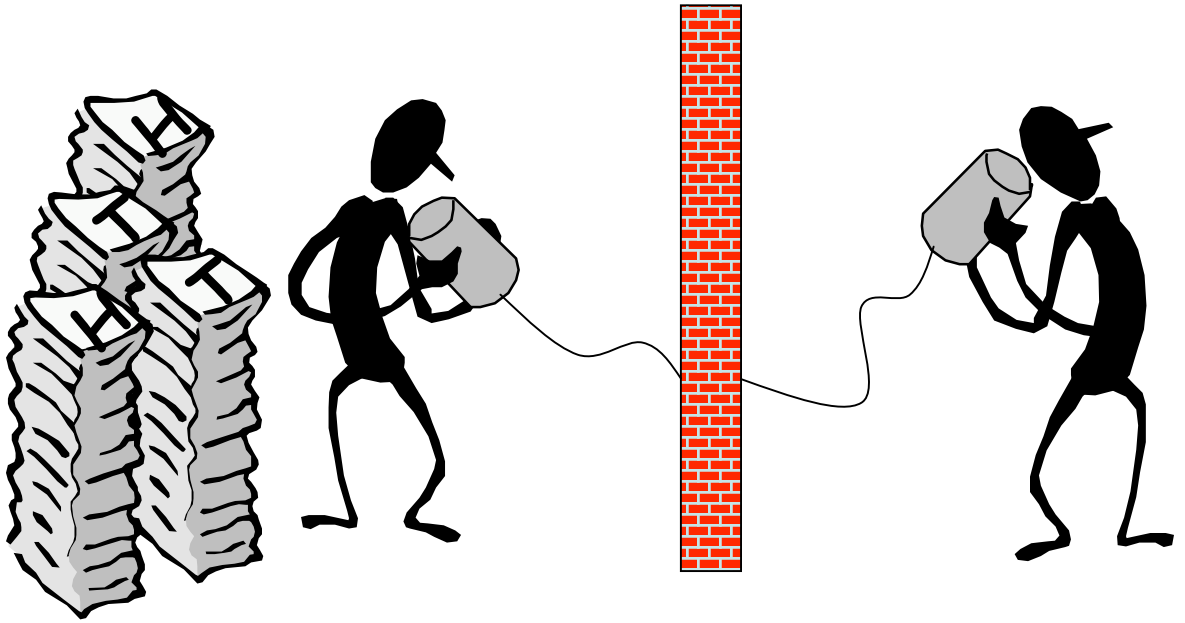
Challenges: if one lowers the supply voltage to save power and increase speed of transitions, the noise margin also decreases.

Sample DC (signaling) Specification

I/O Standard	V_{IL}		V_{IH}		V_{OL}	V_{OH}	I_{OL}	I_{OH}
	V, Min	V, Max	V, Min	V, Max	V, Max	V, Min	mA	mA
LVTTL	–0.3	0.8	2.0	3.45	0.4	2.4	Note(3)	Note(3)
LVCMOS33, LVDCI33	–0.3	0.8	2.0	3.45	0.4	V_{CCO} – 0.4	Note(3)	Note(3)
LVCMOS25, LVDCI25	–0.3	0.7	1.7	V_{CCO} + 0.3	0.4	V_{CCO} – 0.4	Note(3)	Note(3)
LVCMOS18, LVDCI18	–0.3	35% V_{CCO}	65% V_{CCO}	V_{CCO} + 0.3	0.45	V_{CCO} – 0.45	Note(4)	Note(4)
LVCMOS15, LVDCI15	–0.3	35% V_{CCO}	65% V_{CCO}	V_{CCO} + 0.3	25% V_{CCO}	75% V_{CCO}	Note(4)	Note(4)
LVCMOS12	–0.3	35% V_{CCO}	65% V_{CCO}	V_{CCO} + 0.3	25% V_{CCO}	75% V_{CCO}	Note(6)	Note(6)
PCI33_3[5]	–0.2	30% V_{CCO}	50% V_{CCO}	V_{CCO}	10% V_{CCO}	90% V_{CCO}	Note(5)	Note(5)
PCI66_3[5]	–0.2	30% V_{CCO}	50% V_{CCO}	V_{CCO}	10% V_{CCO}	90% V_{CCO}	Note(5)	Note(5)
PCI-X[5]	–0.2	35% V_{CCO}	50% V_{CCO}	V_{CCO}	10% V_{CCO}	90% V_{CCO}	Note(5)	Note(5)

Source: Xilinx Virtex 5 Datasheet

Experiment #2: Digital Encoding



Transmit:
- < .2V for "tails",
- > .8V for "heads"

Receive:
- < .4V is "tails",
- > .6V is "heads"

100 coin flips → one transmission for each flip

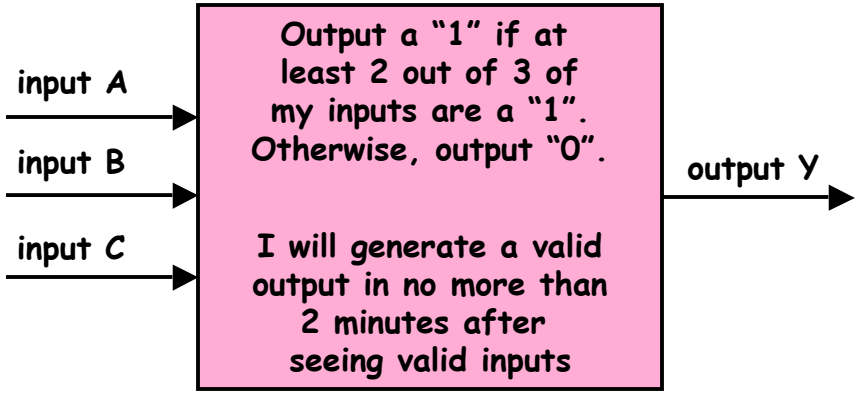
But when does receiver make measurements?

Is  "HT" or "HHHTTT"?

A Digital Processing Element

- *A combinational device* is a circuit element that has
  - 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

Static discipline

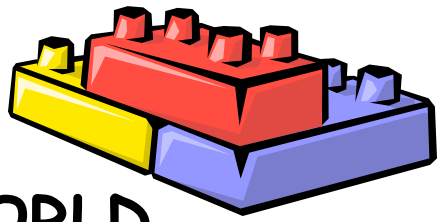


Why have processing blocks?

- The goal of modular design:

ABSTRACTION

- What does that mean anyway:
  - Rules simple enough for a 6-3 to follow…
  - Understanding BEHAVIOR without knowing IMPLEMENTATION
  - Predictable composition of functions
  - Tinker-toy assembly
  - Guaranteed behavior under REAL WORLD circumstances

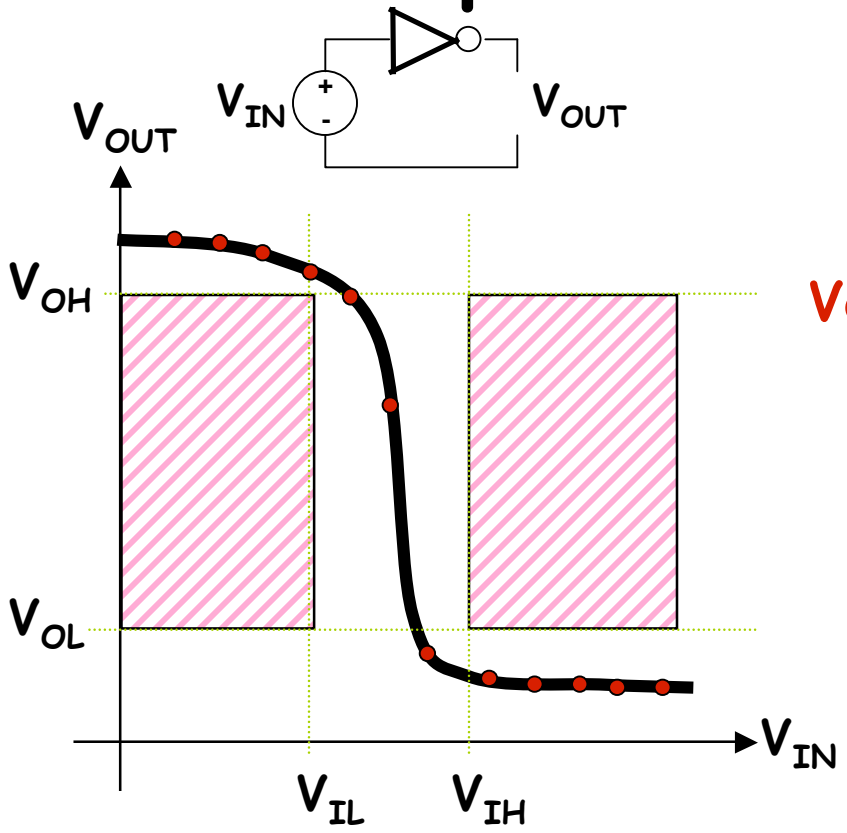


A Combinational Digital System

- A set of interconnected elements is a *combinational device* if
  - each circuit element is a combinational device
  - every input is connected to exactly one output or a constant (eg, some vast supply of 0's and 1's)
  - the circuit contains no directed cycles

- Why is this true?
  - Given an acyclic circuit meeting the above constraints, we can derive functional and timing specs for the input/output behavior from the specs of its components!
  - We'll see lots of examples soon. But first, we need to build some combinational devices to work with…

Example device: An Inverter



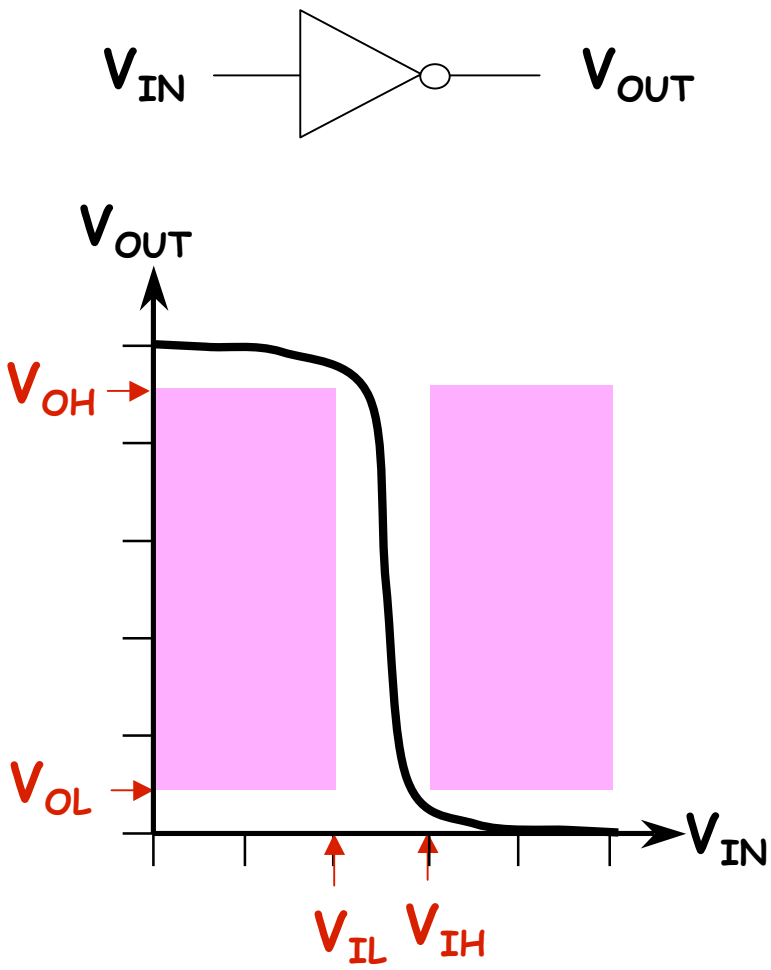
Voltage Transfer Characteristic:
Plot of V_{OUT} vs. V_{IN} where each measurement is taken after any transients have died out.

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

Static Discipline requires that we avoid the shaded regions aka "forbidden zones"), which correspond to *valid* inputs but *invalid* outputs. Net result: combinational devices must have GAIN > 1 and be NONLINEAR.

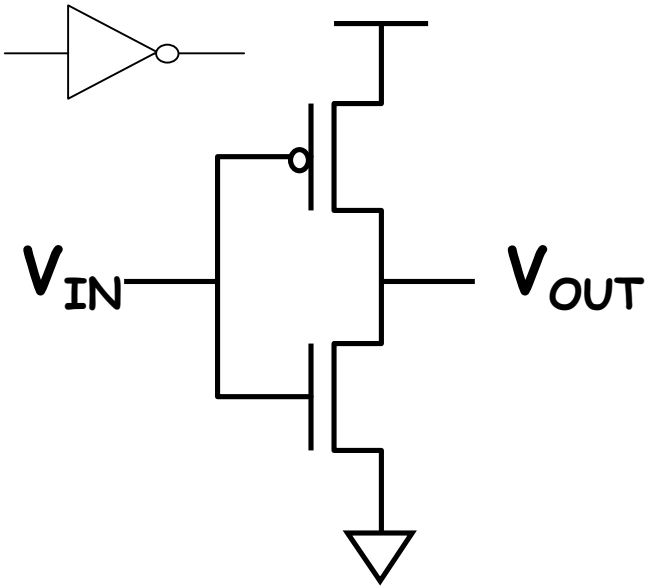
$$\frac{\partial V_{OUT}}{\partial V_{IN}}$$

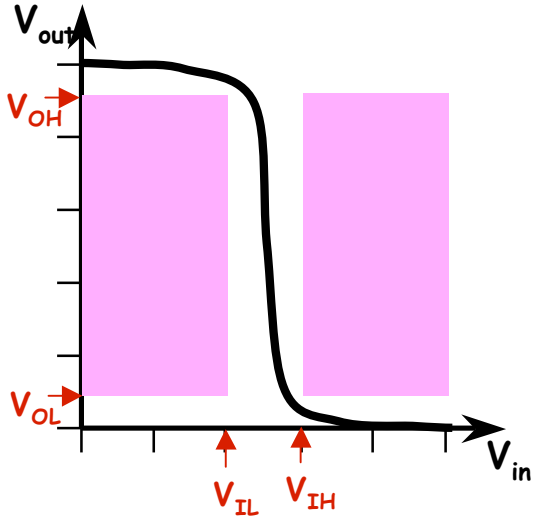
Combinational Device Wish List

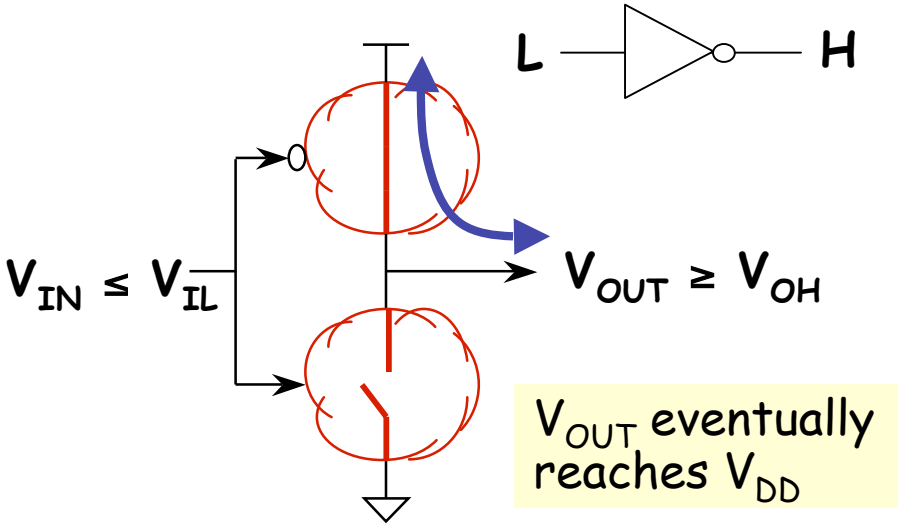


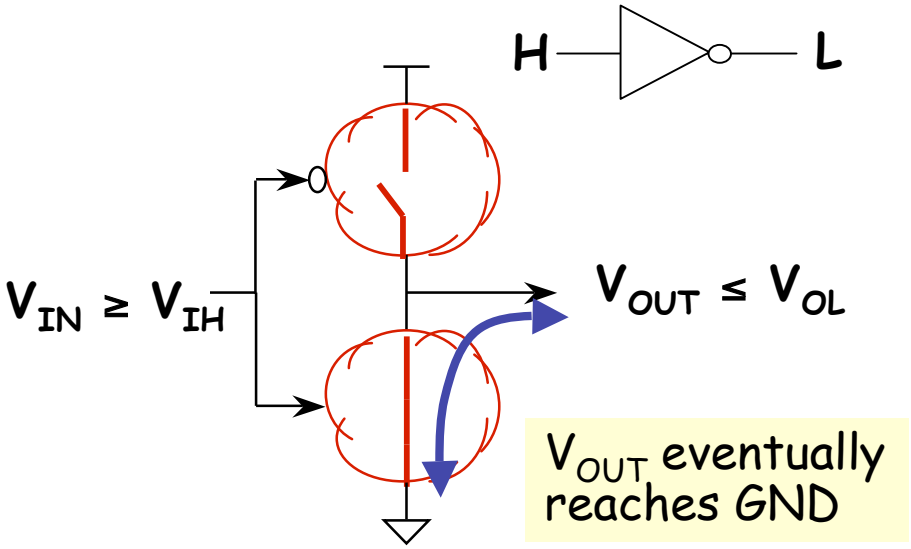
- ✓ Design our system to tolerate some amount of error
  ⇒ Add positive noise margins
  ⇒ VTC: gain>1 & nonlinearity
- ✓ Lots of gain ⇒ big noise margin
- ✓ Cheap, small
- ✓ Changing voltages will require us to dissipate power, but if no voltages are changing, we'd like zero power dissipation
- ✓ Want to build devices with useful functionality (what sort of operations do we want to perform?)

Wishes Granted: CMOS



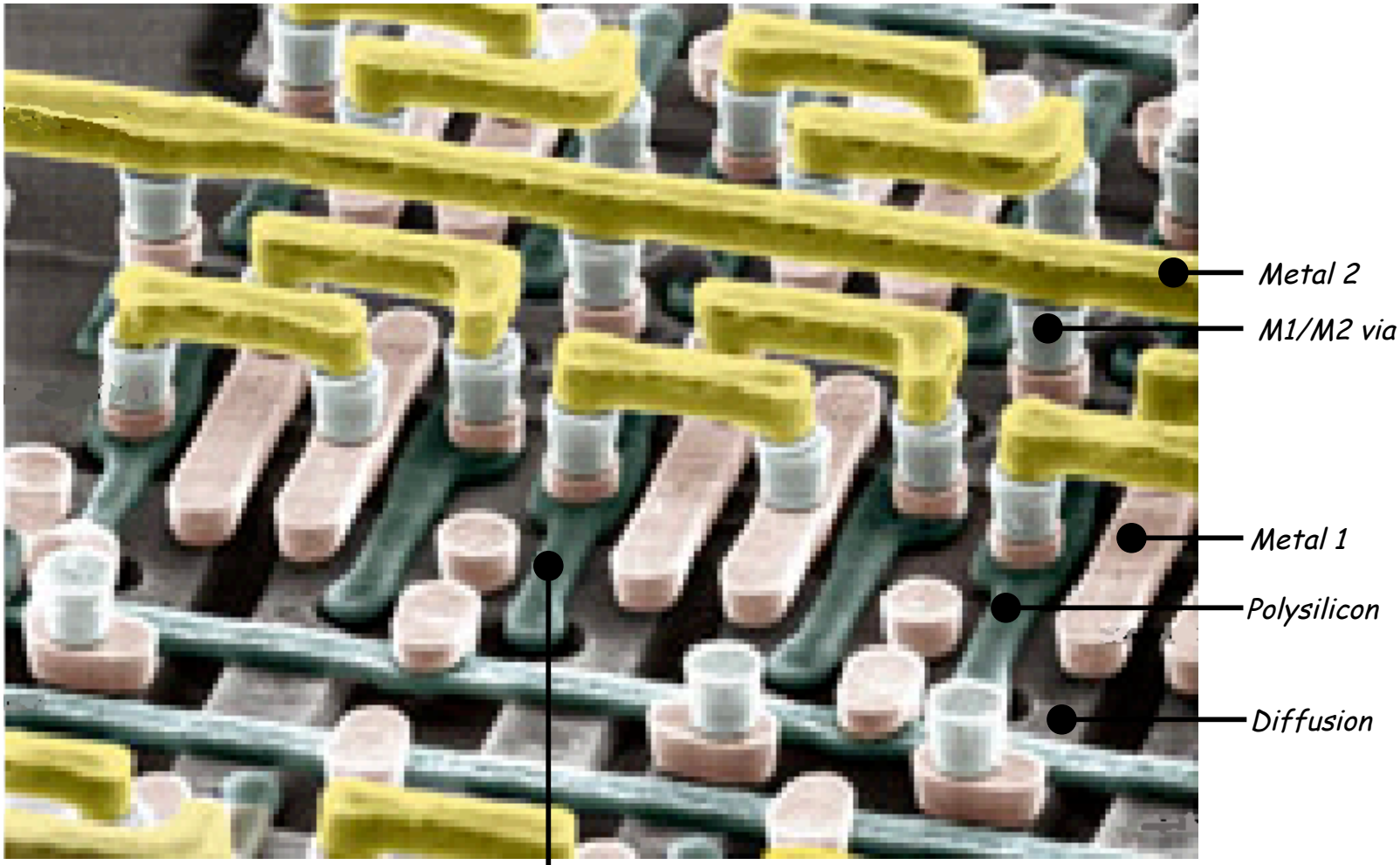


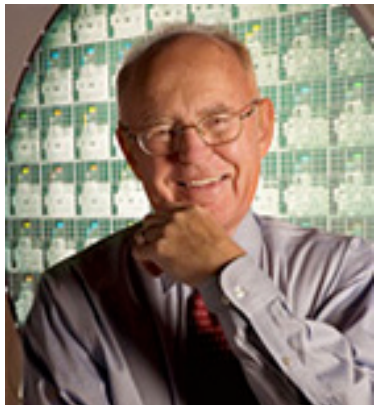




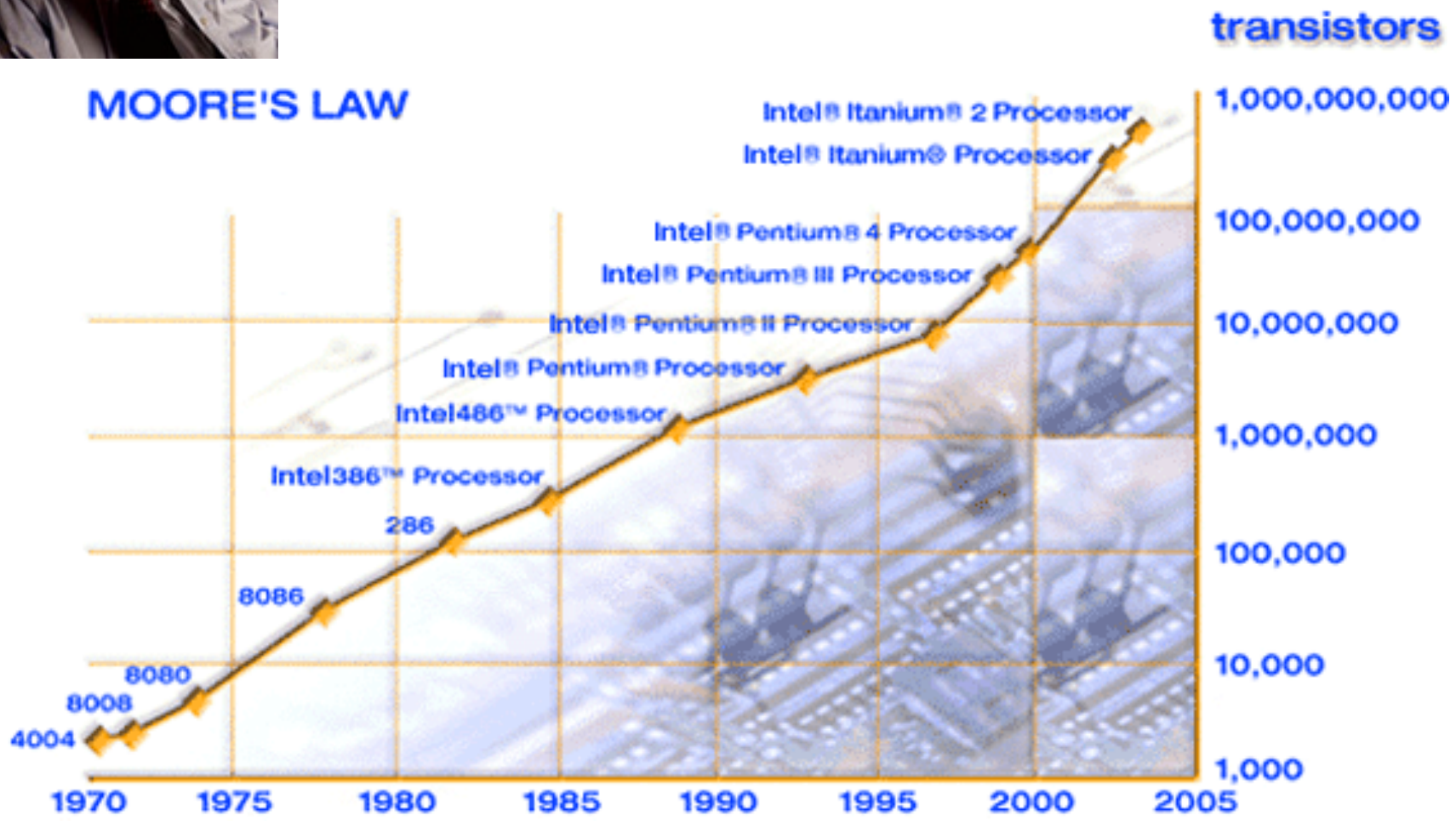
Digital Integrated Circuits

IBM photomicrograph (SiO_2 has been removed!)





CMOS Forever!?



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
- 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