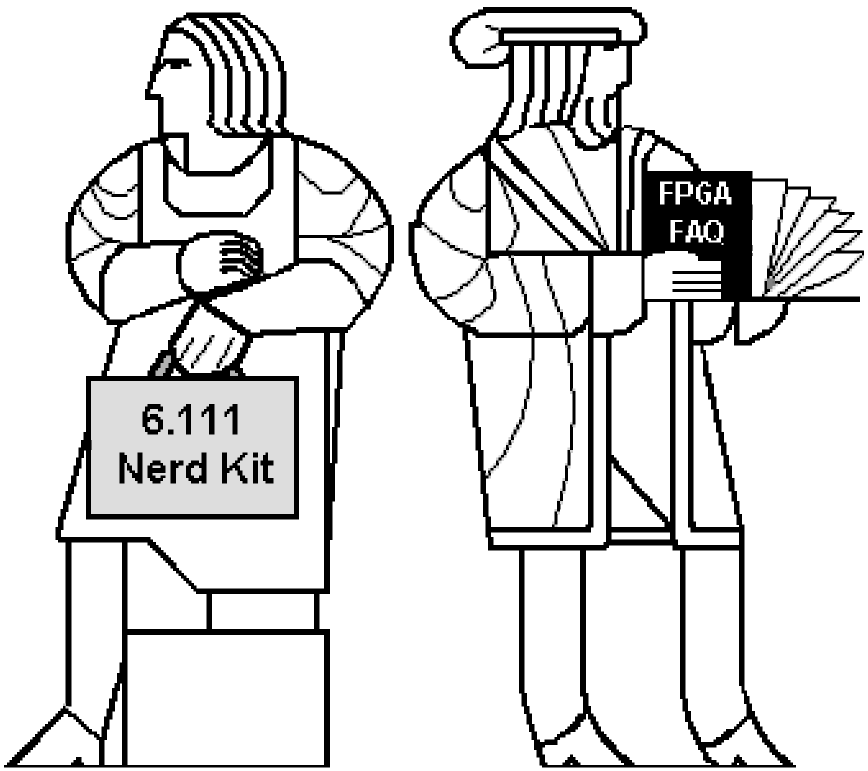
Welcome to 6.111!
Introductory Digital Systems Laboratory

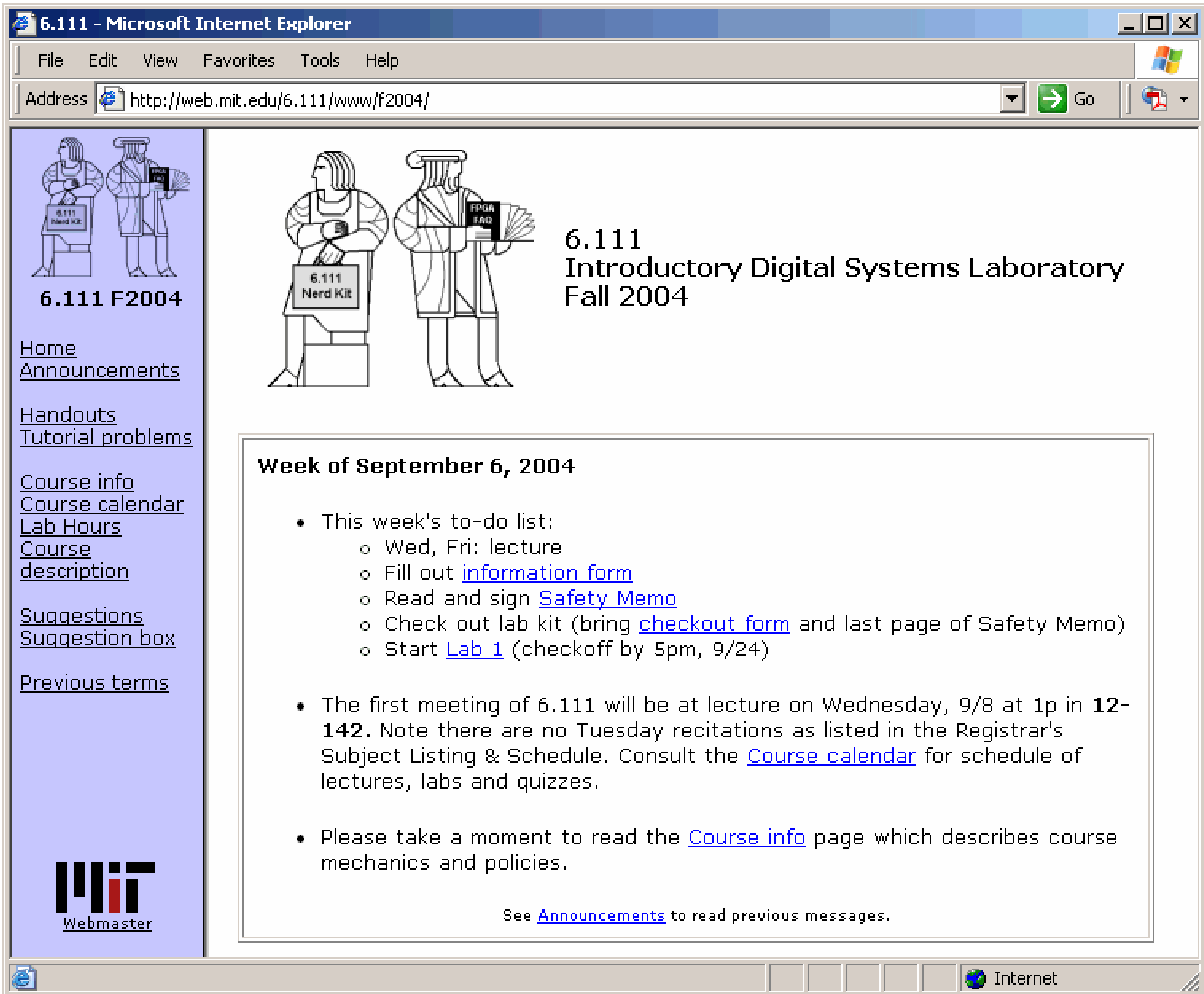


Handouts:
- Info form (yellow)
- Course Calendar
- Safety Memo
- Kit Checkout Form
- Lecture slides

Lectures:
- Chris Terman

TAs:
- Karthik Balakrishnan
- HuangBin Son
- Colin Weltin-Wu

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!

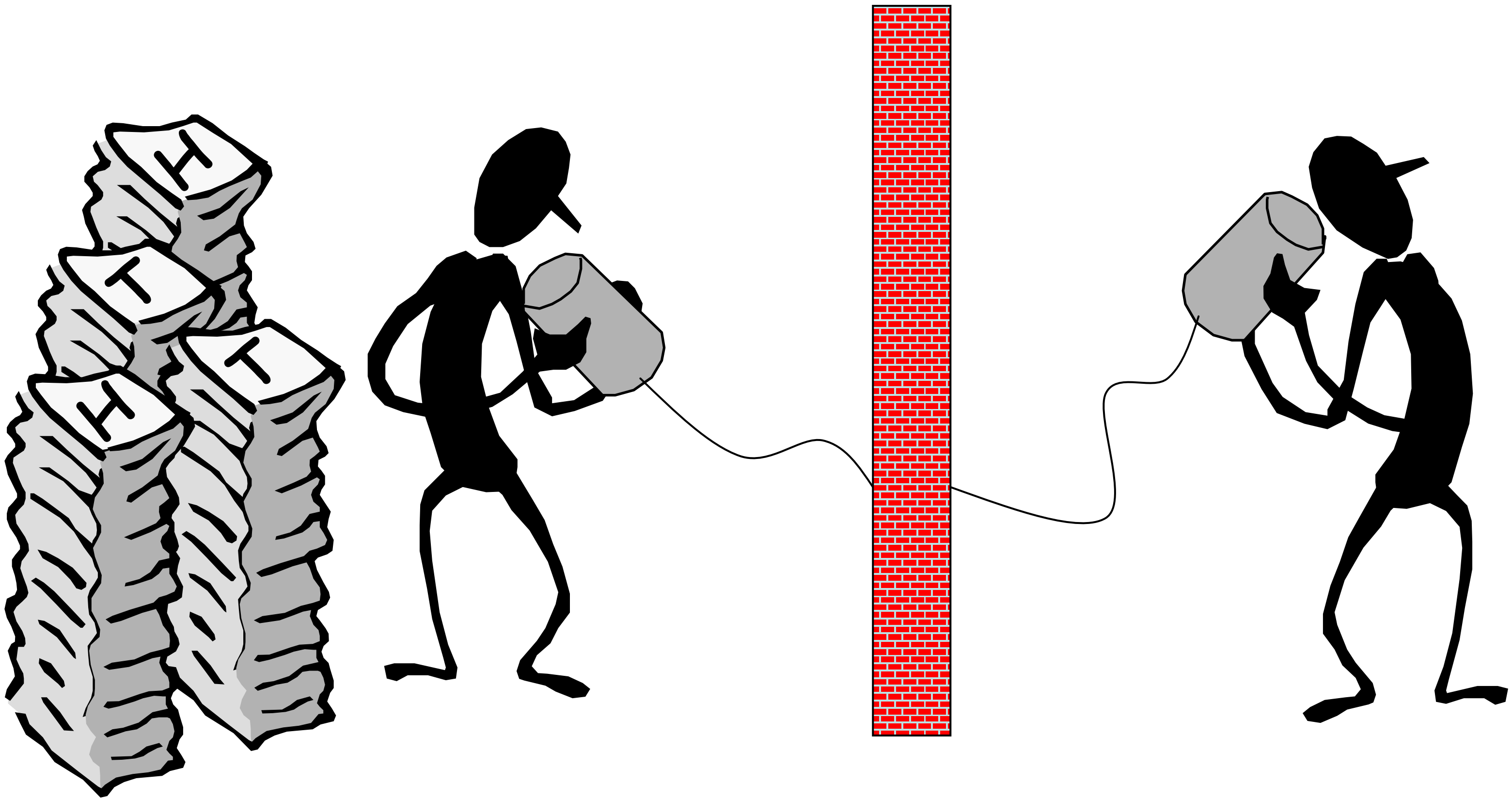
Labs: learning the ropes

- Lab 1
  - Learn about the lab kit and wire something
  - Learn about lab equipment in the Digital Lab (38-600): oscilloscopes and logic analyzers
  - Program and test a PAL (Programmable Array Logic Device)
  - Introduction to Verilog
- Lab 2
  - Design and implement a Finite State Machine (FSM)
  - Use Verilog to program an FPGA
  - Learn how to use an SRAM
  - Report and its revision will be evaluated for CI-M
- Lab 3
  - Design a complicated system with multiple FSMs (Major/Minor FSM)
  - Implement RAMs and ROMs in an FPGA
  - Interfacing to analog components (ADC and DAC)

Final Project

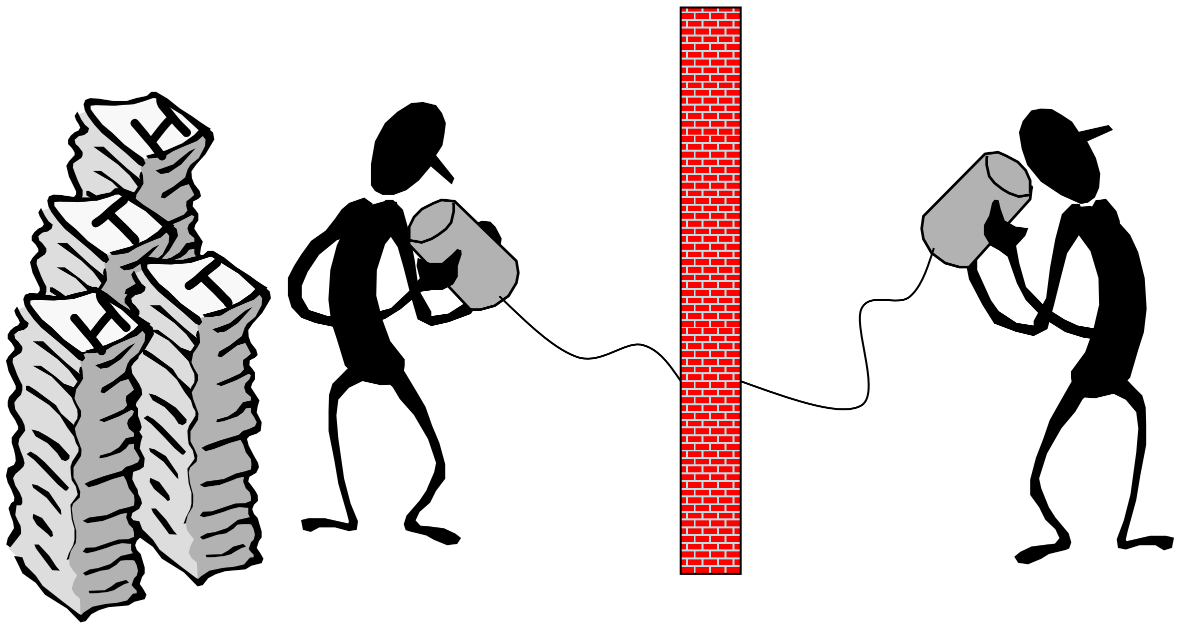
- Done in groups of two or three
- Open ended
- You and the staff negotiate a project proposal
  - Must emphasize digital concepts, but inclusion of analog interfaces (*e.g.*, data converters, sensors or motors) common and often desirable
  - Proposal Conference, several Design Reviews
- Design presentation in class (% of the final grade for the in-class presentation)
- 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

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 that 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 components obey their specified behaviors.

How is this achieved? **CONTRACTS!**

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

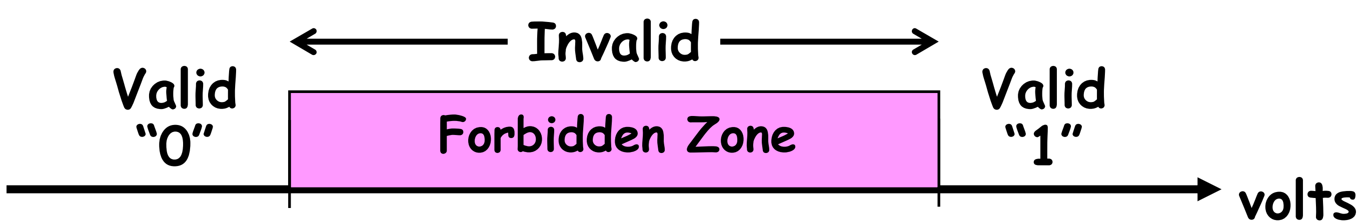
Going Digital

- Digital representation = information encoded as a sequence of symbols chosen from a (small) set.
- Keep in mind that the world is not digital, we will simply engineer it to behave that way. Furthermore, we must use real physical (analog, continuous) phenomena to implement digital designs!
- Common choices
  - Binary symbols (0, 1)
  - If we have DC connectivity (wired): encode using voltages/currents
  - If we don't have DC connectivity (wireless): encode using frequency/phase

- Going digital keeps the contracts simple – limit quantum of information we process in exchange for reliablity

Using Voltages Digitally

- Key idea: don't allow "0" to be mistaken for a "1" or vice versa
- Use the same "uniform representation convention" for *every* component and wire in our digital system
- To implement devices with high reliability, we outlaw "close calls" via a representation convention which forbids a range of voltages between "0" and "1".

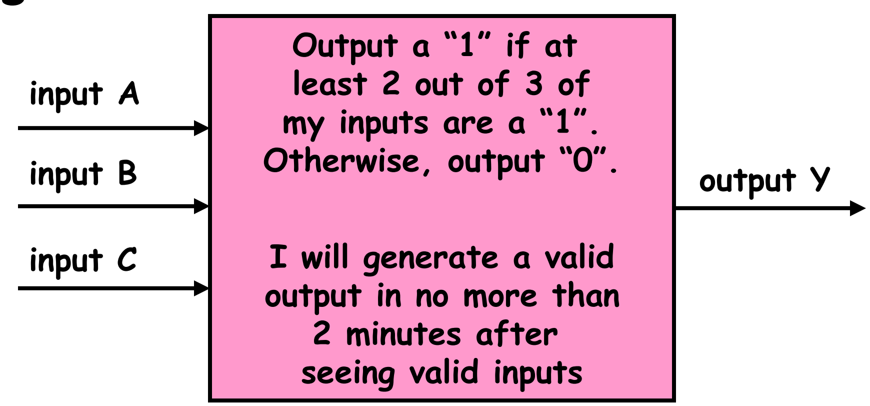


Consequence: notion of valid and invalid signals

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

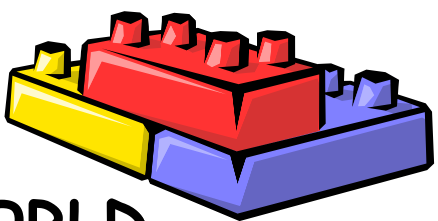


Why have processing blocks?

- 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

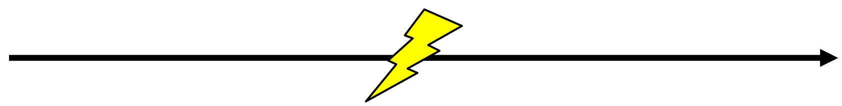
- **A set of interconnected elements is a *combinational device* if**
  - **each circuit element is a combinational device**
  - **every input is connected to exactly one output or a constant (eg, some vast supply of 0's and 1's)**
  - **the circuit contains no directed cycles**
- **Why is this true?**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

Wires: theory vs. practice

Does a wire obey the static discipline?

Noise: changes voltage…

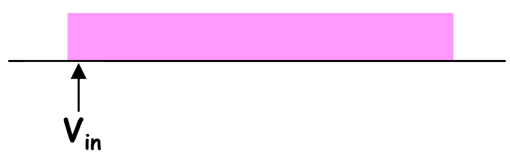
V_{in}



V_{out}

(voltage close to boundary with forbidden zone)

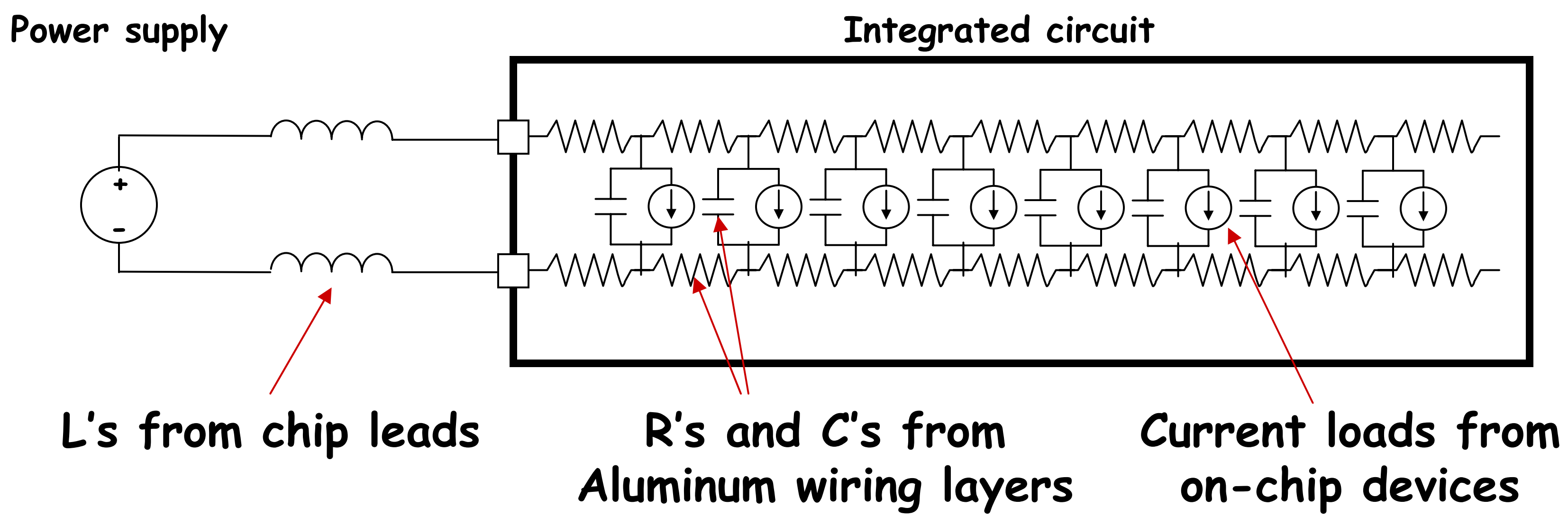
(voltage in forbidden zone: **Oops, not a valid voltage!**)



Questions to ask ourselves:

In digital systems, where does noise come from?
How big an effect are we talking about?

Power Supply Noise



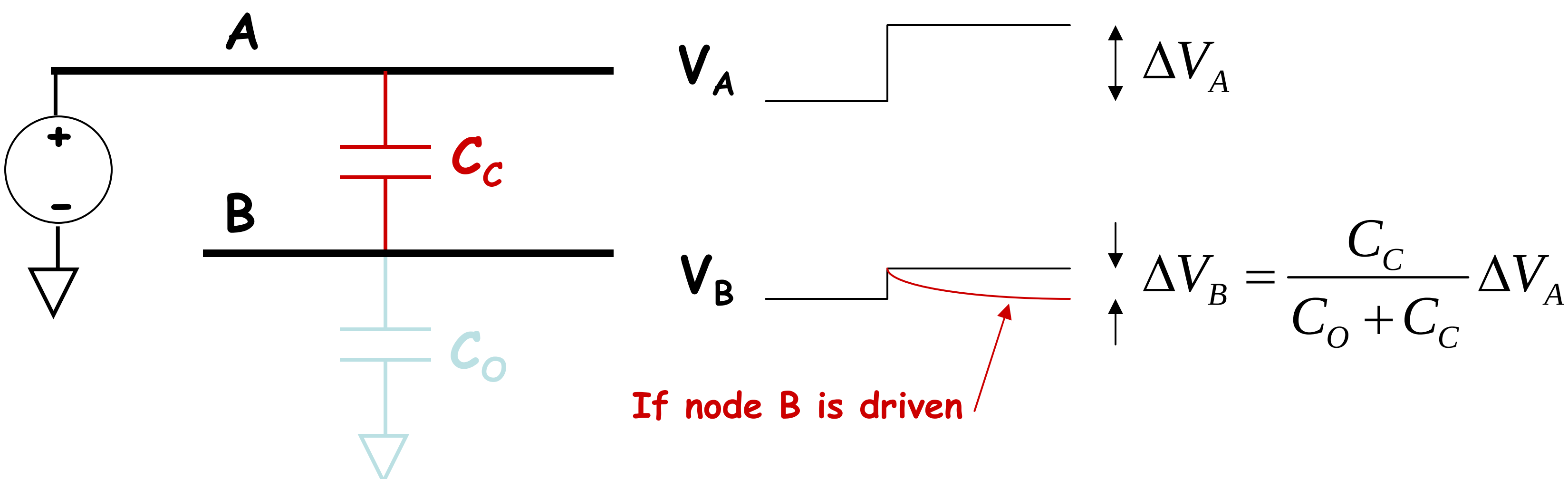
ΔV from:

- IR drop

  (between gates: 30mV, within module: 50mV, across chip: 350mV)
- L(dI/dt) drop

  (use extra pins and bypass caps to keep within 250mV)
- LC ringing triggered by current "steps"

Crosstalk

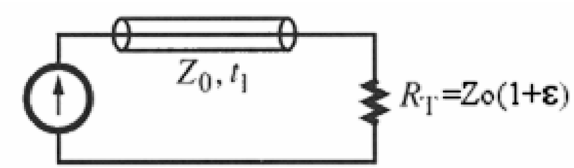


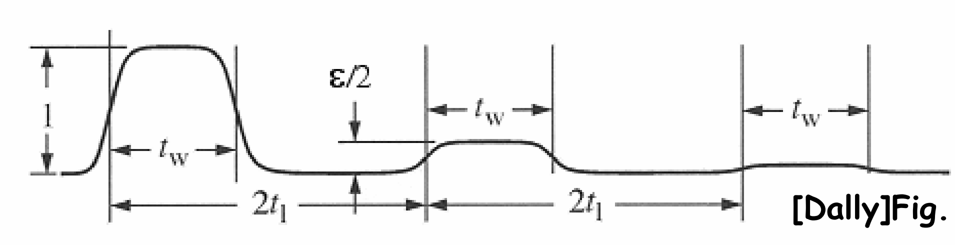
This situation frequently happens on integrated circuits where there are many overlapping wiring layers. In a modern integrated circuit ΔV_A might be 2.5V, C_O = 20fF and C_C = 10fF → ΔV_B = 0.83V! Designers often try to avoid these really bad cases by careful routing of signals, but some crosstalk is unavoidable.

Intersymbol Interference

ΔV from energy storage left over from earlier signaling on the wire:

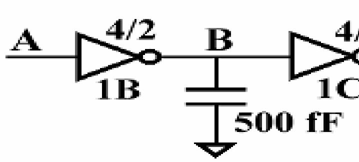
- transmission line discontinuities
  (reflections off of impedance mismatches and terminations)



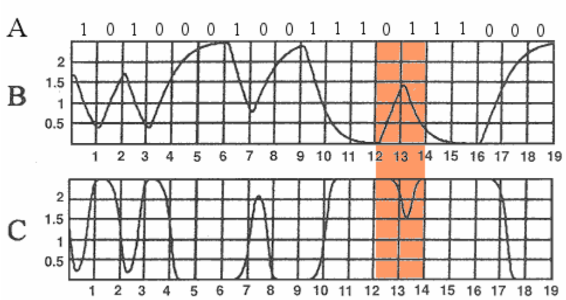


[Dally]Fig. 6-17

- charge storage in RC circuit
  (narrow pulses are lost due to incomplete transitions)

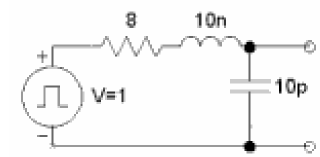


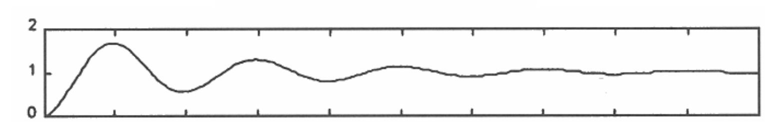
[Dally]Fig. 6-19



[Dally]Fig. 6-20

- RLC ringing
  (triggered by voltage "steps")

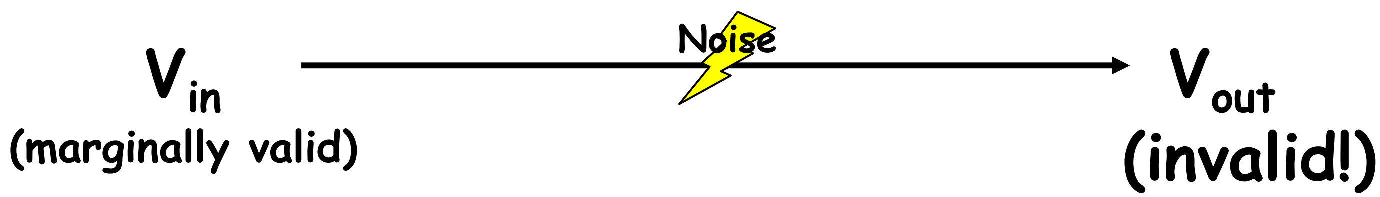




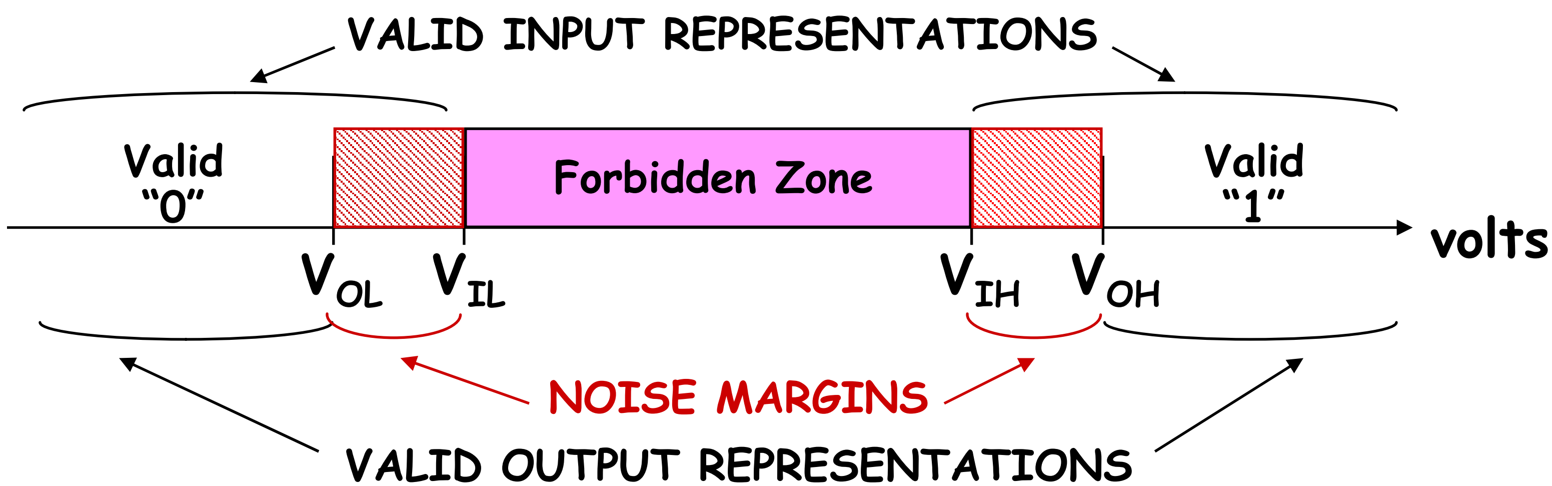
Fix: slower operation, limiting voltage swings and slew rates

Needed: Noise Margins!

Does a wire obey the static discipline?



No! A combinational device must restore marginally valid signals. It must accept marginal inputs and provide unquestionable outputs (i.e., to leave room for noise).



Sample DC (signalling) Specification

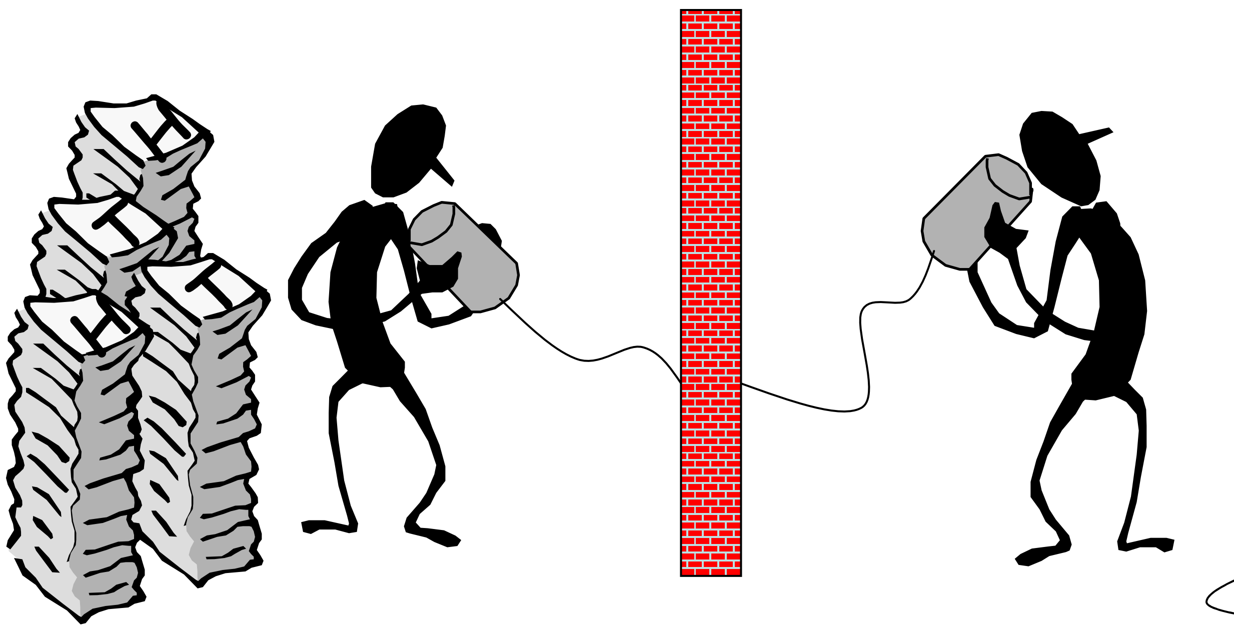
HCMOS family characteristics — FAMILY SPECIFICATIONS

DC CHARACTERISTICS FOR 74HCT

Voltages are referenced to GND (ground = 0 V)

SYMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITIONS		
		74HCT								V_{CC} (V)	V_I	OTHER
		+25			−40 to +85		−40 to +125					
		min.	typ.	max.	min.	max.	min.	max.				
V_{IH}	HIGH level input voltage	2.0	1.6		2.0		2.0		V	4.5 to 5.5		
V_{IL}	LOW level input voltage		1.2	0.8		0.8		0.8	V	4.5 to 5.5		
V_{OH}	HIGH level output voltage all outputs	4.4	4.5		4.4		4.4		V	4.5	V_{IH} or V_{IL}	$-I_O = 20\ \mu A$
V_{OH}	HIGH level output voltage standard outputs	3.98	4.32		3.84		3.7		V	4.5	V_{IH} or V_{IL}	$-I_O = 4.0$ mA
V_{OH}	HIGH level output voltage bus driver outputs	3.98	4.32		3.84		3.7		V	4.5	V_{IH} or V_{IL}	$-I_O = 6.0$ mA
V_{OL}	LOW level output voltage all outputs		0	0.1		0.1		0.1	V	4.5	V_{IH} or V_{IL}	$I_O = 20\ \mu A$
V_{OL}	LOW level output voltage standard outputs		0.15	0.26		0.33		0.4	V	4.5	V_{IH} or V_{IL}	$I_O = 4.0$ mA
V	LOW level output		0.16	0.26		0.33		0.4	V	4.5	V	I = 6.0 mA

Experiment #2: Digital Encoding



0.2V Noise Margins

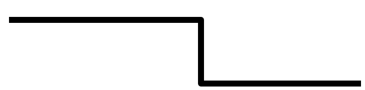
Transmit:

<.2V for "tails",
>.8V for "heads"

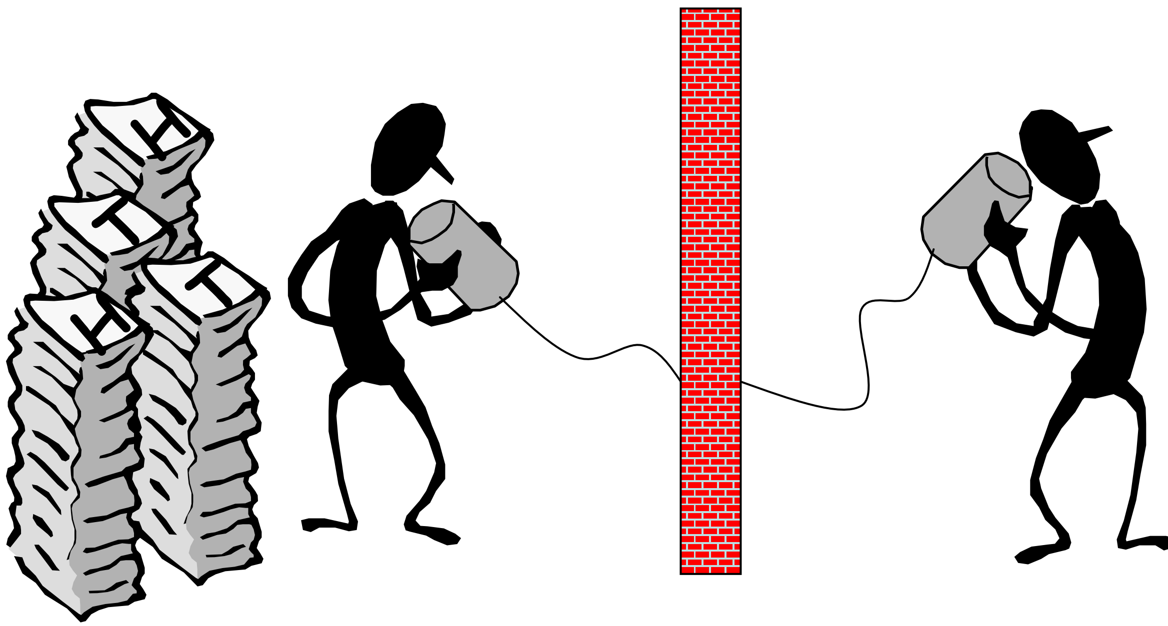
Receive:

<.4V is "tails",
>.6V is "heads"

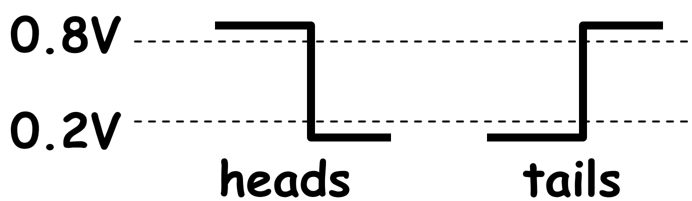
100 coin flips → one transmission for each flip

✓ **But when does receiver make measurements?**
Is  **"HT" or "HHHTTT"?**

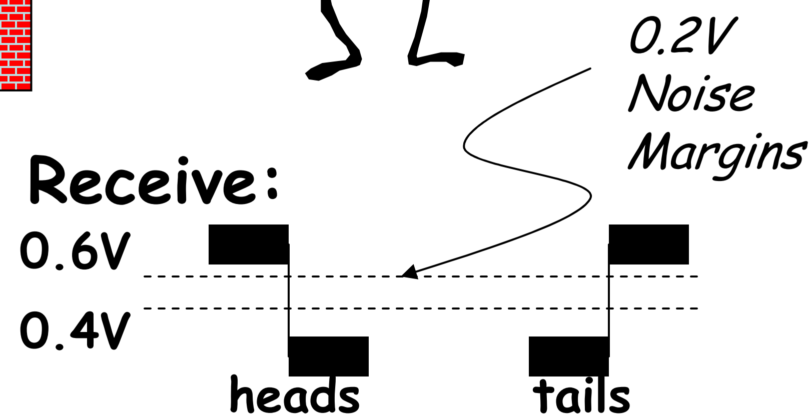
Experiment #3: Manchester Encoding



Transmit:

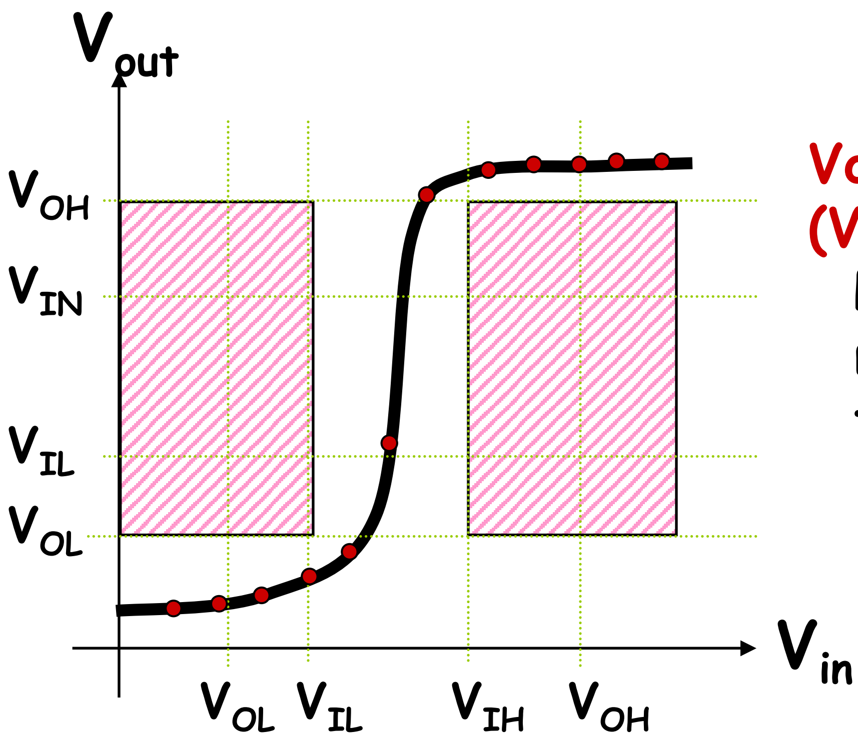


Receive:



100 coin flips → one transmission for each flip
Receiver can tell when new information is present.

Example device: A Buffer



Voltage Transfer Characteristic (VTC):

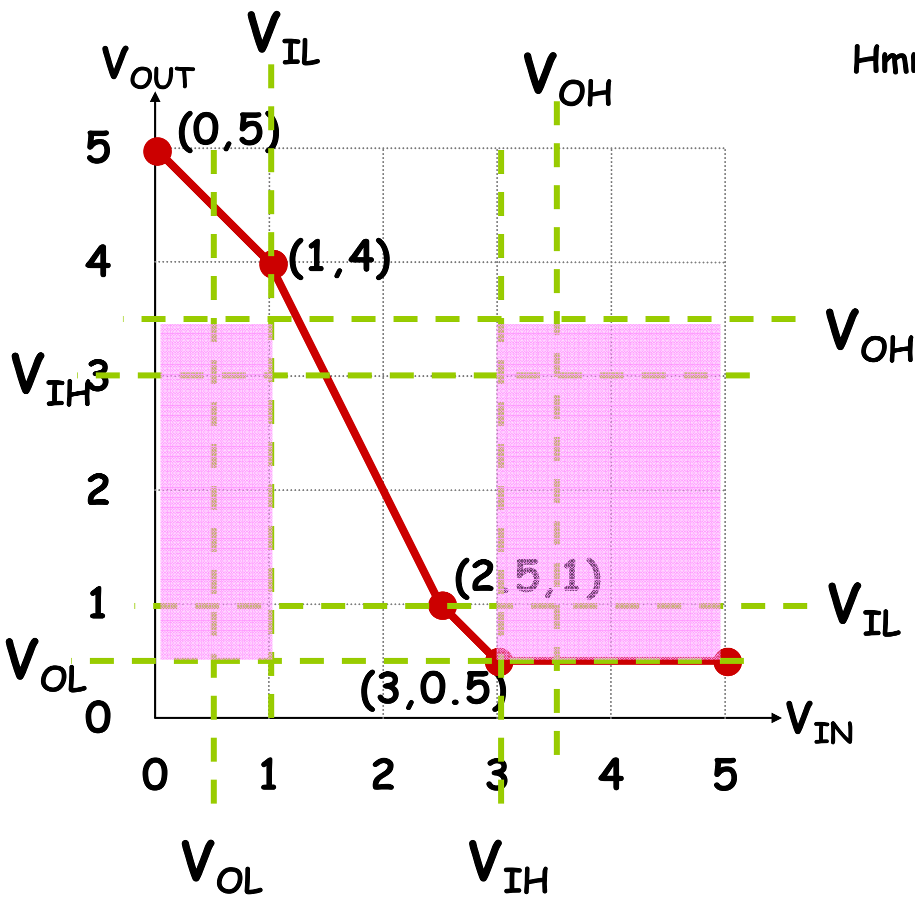
Plot of V_{out} vs. V_{in} where each measurement is taken after any transients have died out.

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

Static Discipline requires that 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**.

Can this be a combinational device?

Suppose that you measured the voltage transfer curve of the device shown below. Could we build a logic family using it as a single-input combinational device?



Hmmm, it had better be an *INVERTER*…

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

Try V_{OL} = 0.5 V

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

Try V_{IH} = 3 V

Now, choose noise margins – find an N and set

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

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

Such that

V_{IH} IN generates V_{OL} or less out; AND

V_{IL} IN generates V_{OH} or more out.

Try N = 0.5 V

Summary

- **We'll 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**