EECS 373
Design of Microprocessor-Based Systems

Robert Dick
University of Michigan

Lecture 11: ADCs, DACs, prototyping, memory, and PCBs

14 February 2017

Some slides inherited from Mark Brehob.

Outline

- Context and review
- Project and midterm schedule
- ADC and DAC operation and errors
- Prototyping
- Memory
- PCB design

Context and review

- Struct packing.
  - Memory use implications of struct element ordering.
- Implementation details.
  - PWM.
  - Virtual timers.
  - Linked lists.
- ADCs and DACs.
  - Temporal and spatial discretization implications.
  - Oversampling.

Outline

- ~~Context and review~~
- Project and midterm schedule
- ADC and DAC operation and errors
- Prototyping
- Memory
- PCB design

Project schedule

- 13-15 Feb: Project proposal meetings.
- 28 Feb: Written project proposal regrade deadline.
- 6 Mar: Deadline for ordering/ensuring in stock two critically important components.
- 12 Mar: Get started on projects.
- 20 Mar: Labs out, full-time on projects.

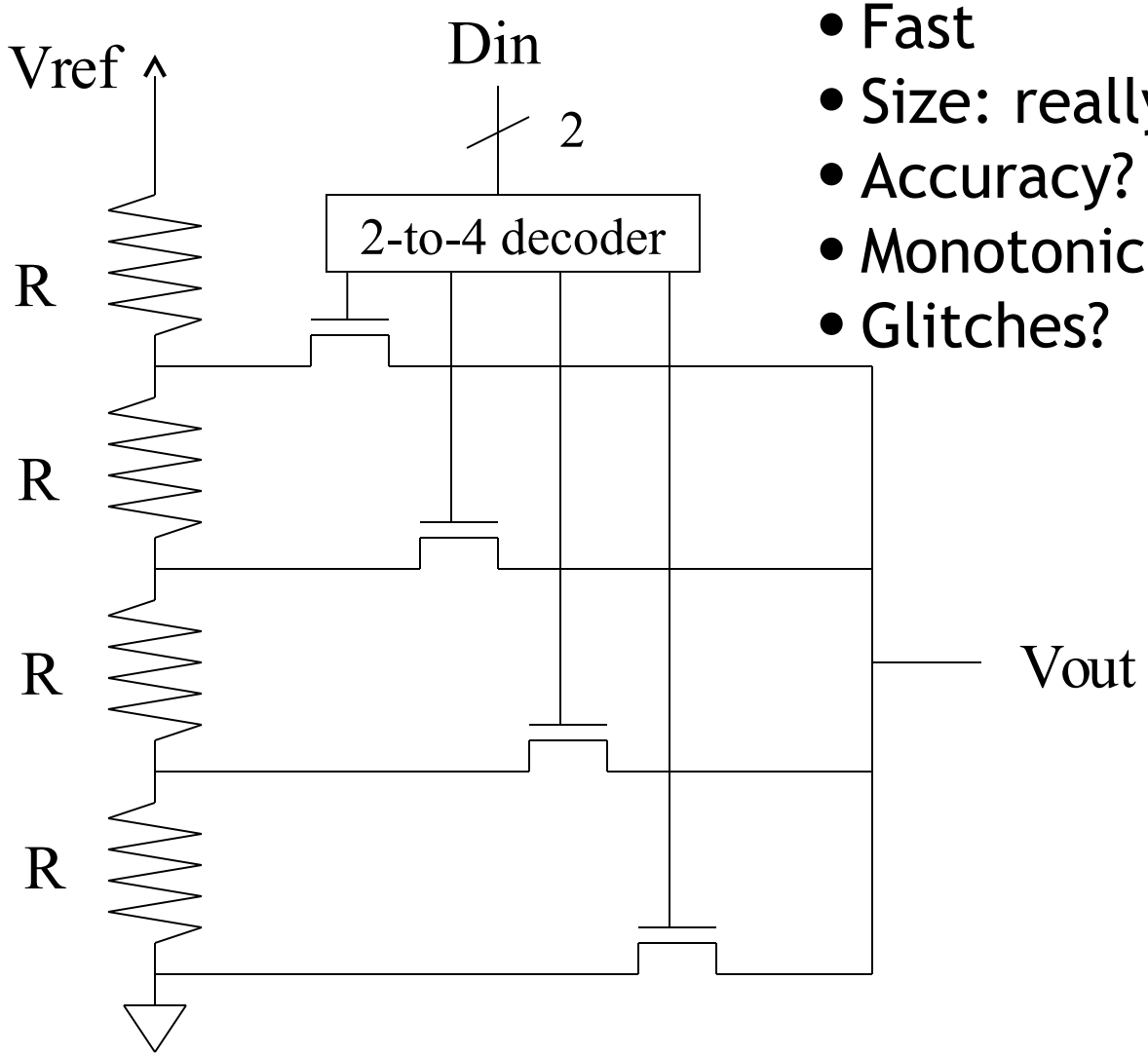
Midterm schedule

- Solutions to practice midterms posted today.
- Can get help from TAs, Matt, me.
- Exam: 8-9:30pm, 22 Feb, 2505 GGBL.
- Review session times and locations posted to Piazza by Thursday.
- In-class review in next two lectures.

Outline

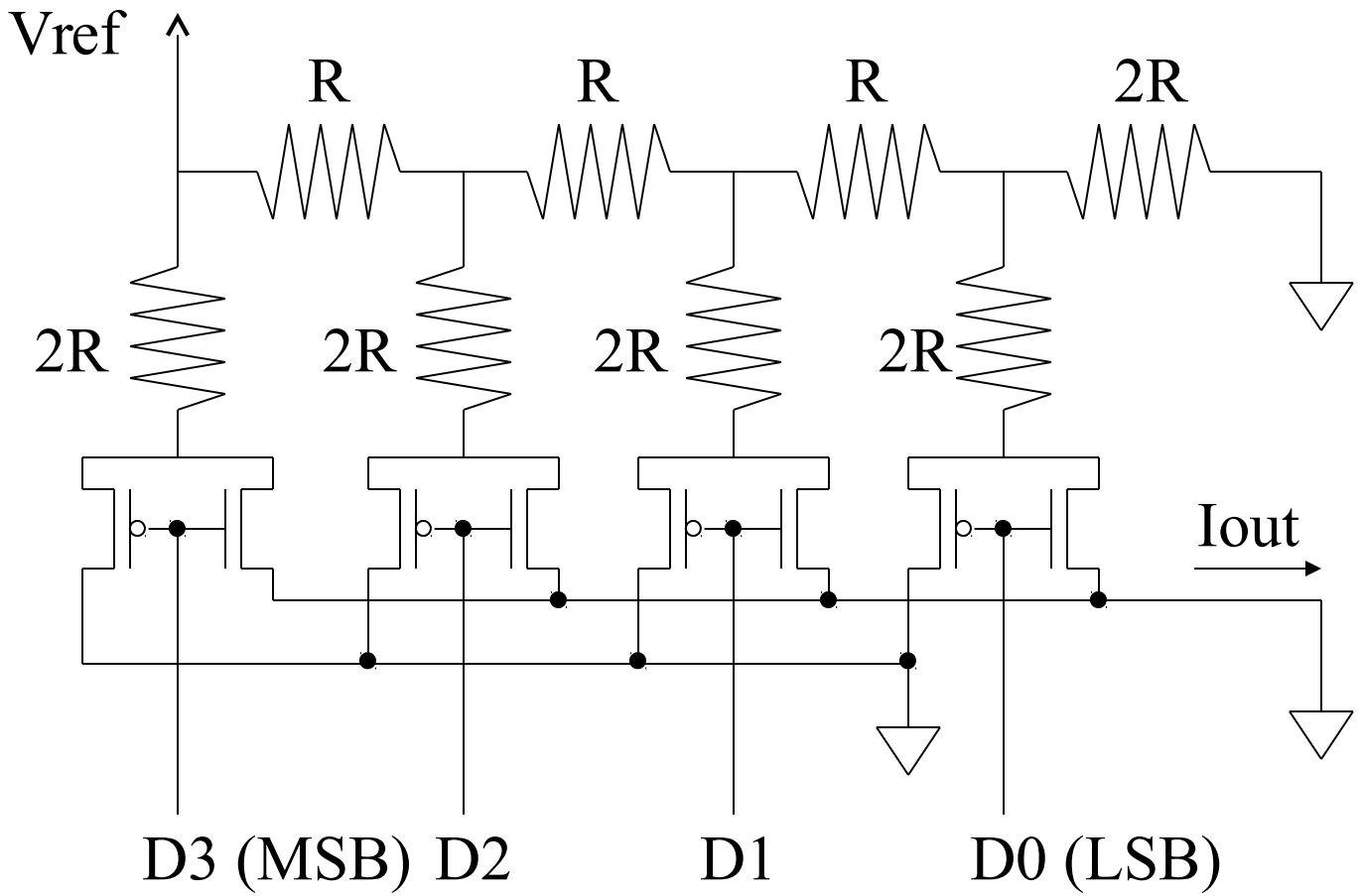
- ~~Context and review~~
- ~~Project and midterm schedule~~
- ADC and DAC operation and errors
- Prototyping
- Memory
- PCB design

DAC #1: voltage divider



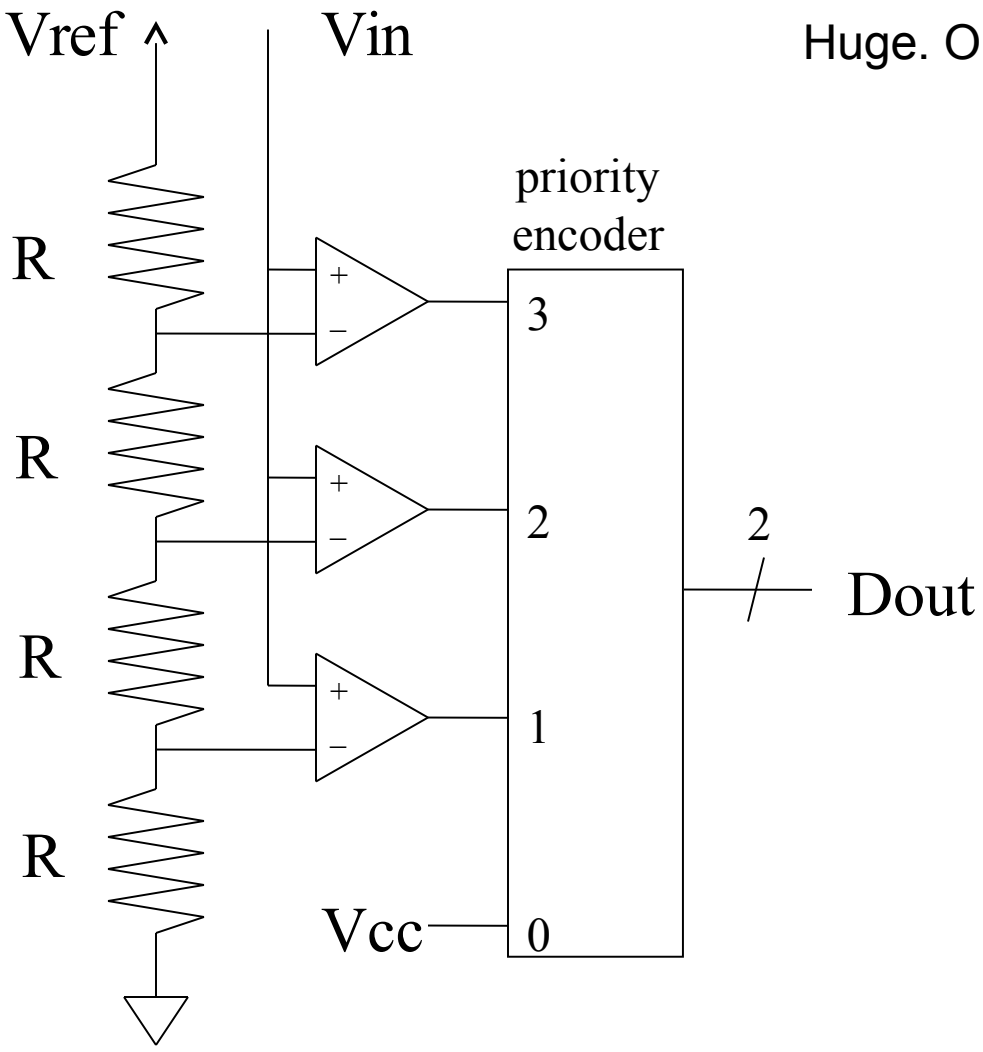
- Fast
- Size: really big: $O(2^n)$
- Accuracy?
- Monotonicity?
- Glitches?

DAC #2: R/2R ladder



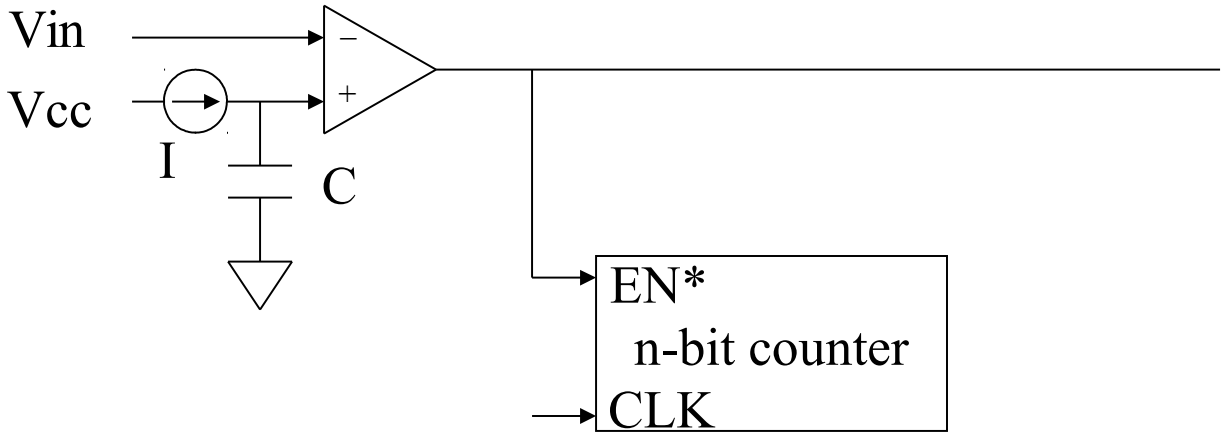
- Size: small, O(n)
- Accuracy?
- Monotonicity? (Consider 0111 -> 1000)
- Glitches?

ADC #1: flash



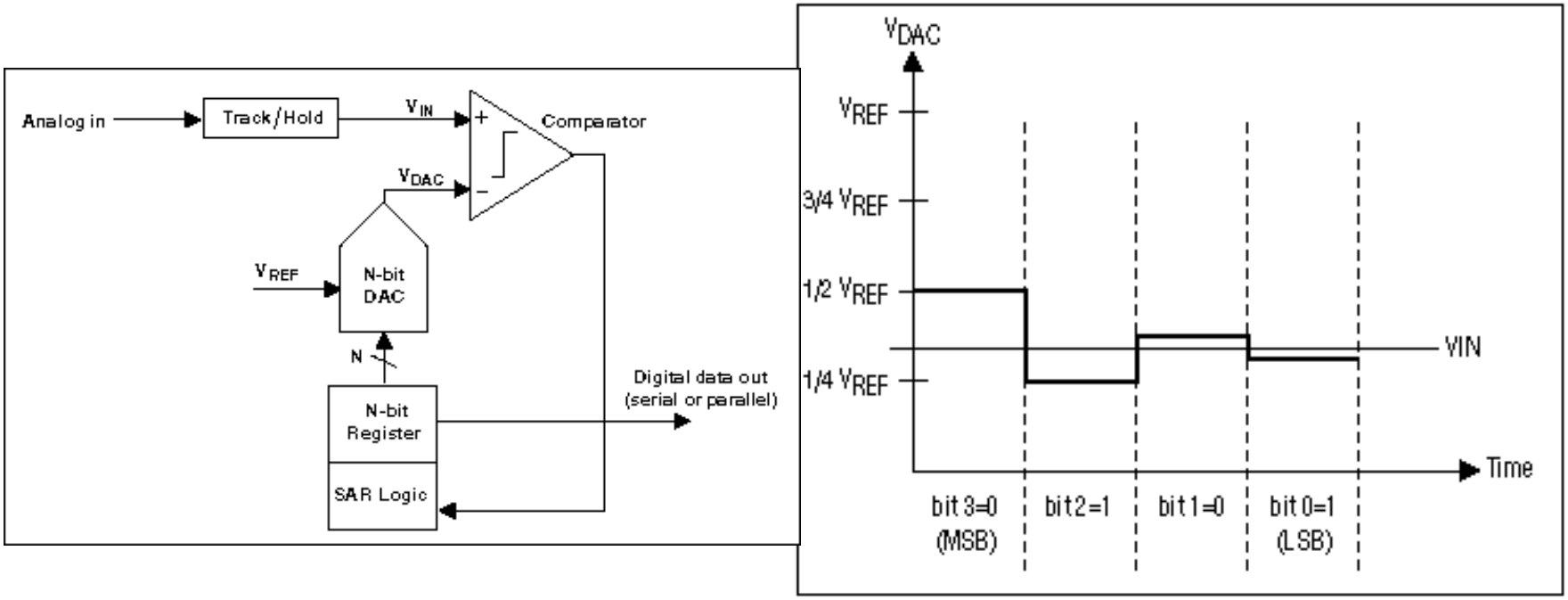
Huge. $O(2^n)$.

ADC #2: single-slope integration



- Start: Reset counter, discharge C.
- Charge C at fixed current I until Vc > Vin . How should C, I, n, and CLK be related?
- Final counter value is Dout.
- Slow: conversion may take several milliseconds.
  - $O(2^n)$
- Good differential linearity (dI/dO)
- Absolute linearity depends on precision of C, I, and clock.

ADC #3: successive approximation

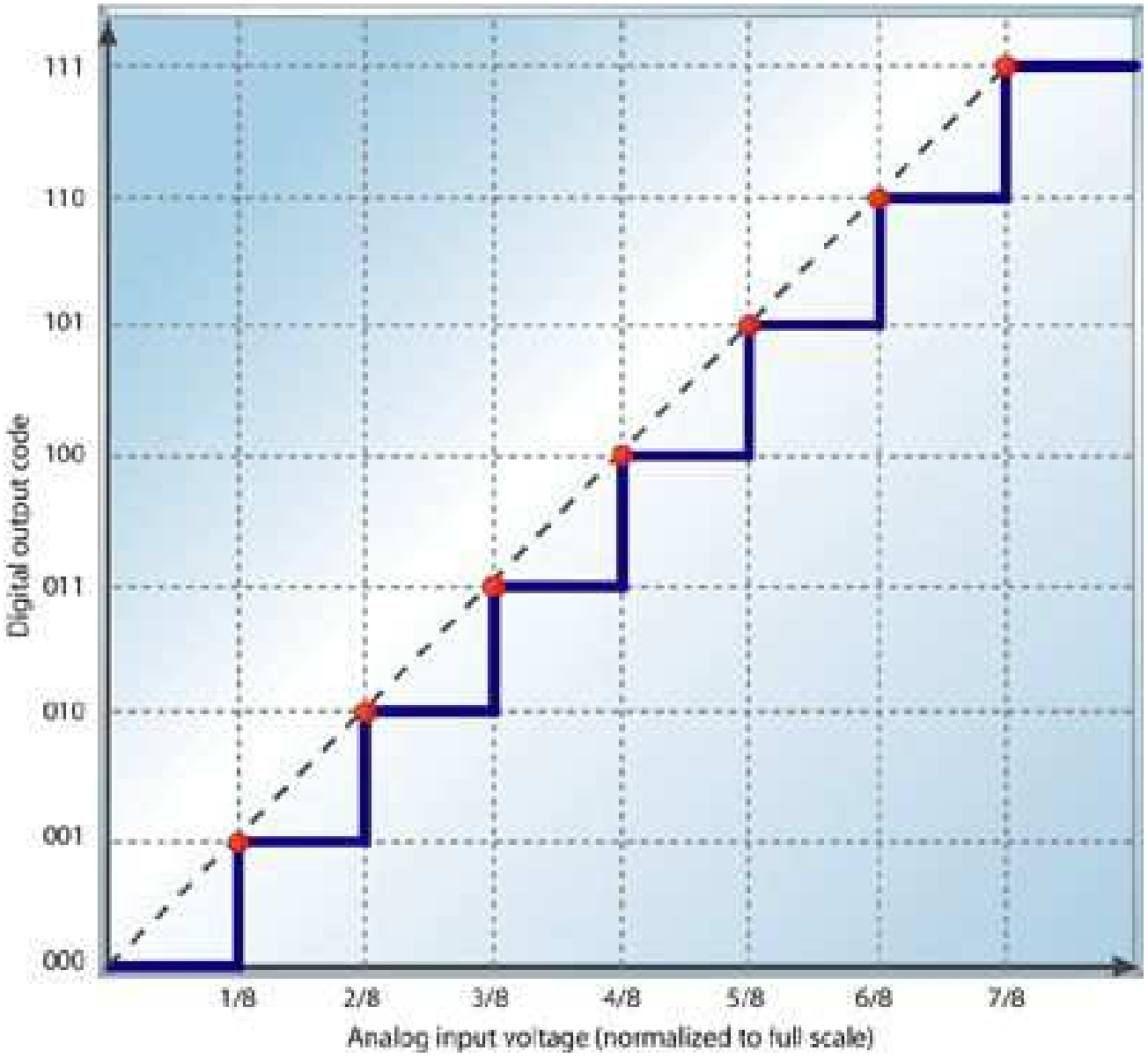


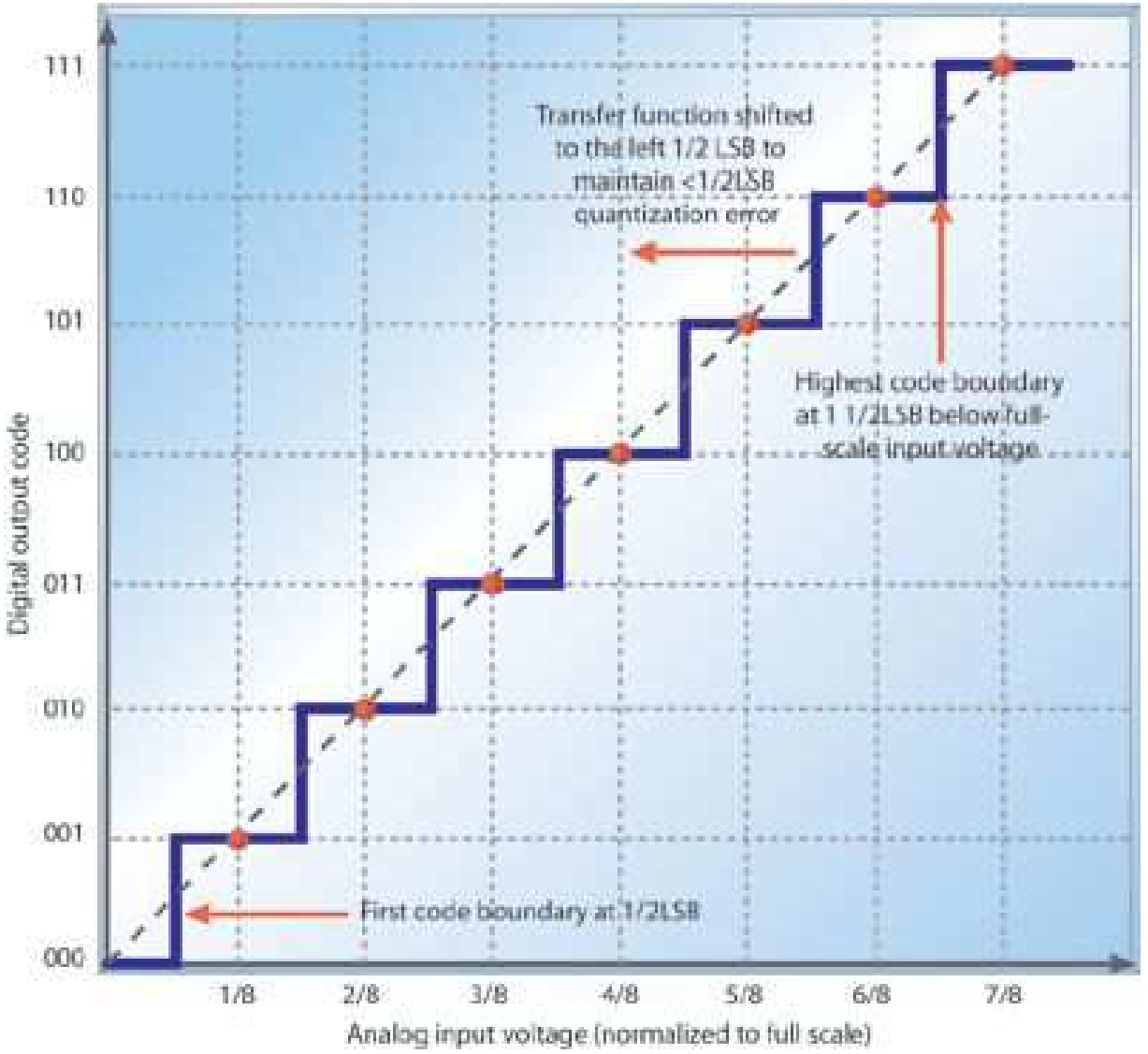
1 Sample → Multiple cycles

- Uses DAC for guessing.
- Faster: O(n)
- Goes from MSB to LSB.
- Not good for high-speed ADCs.

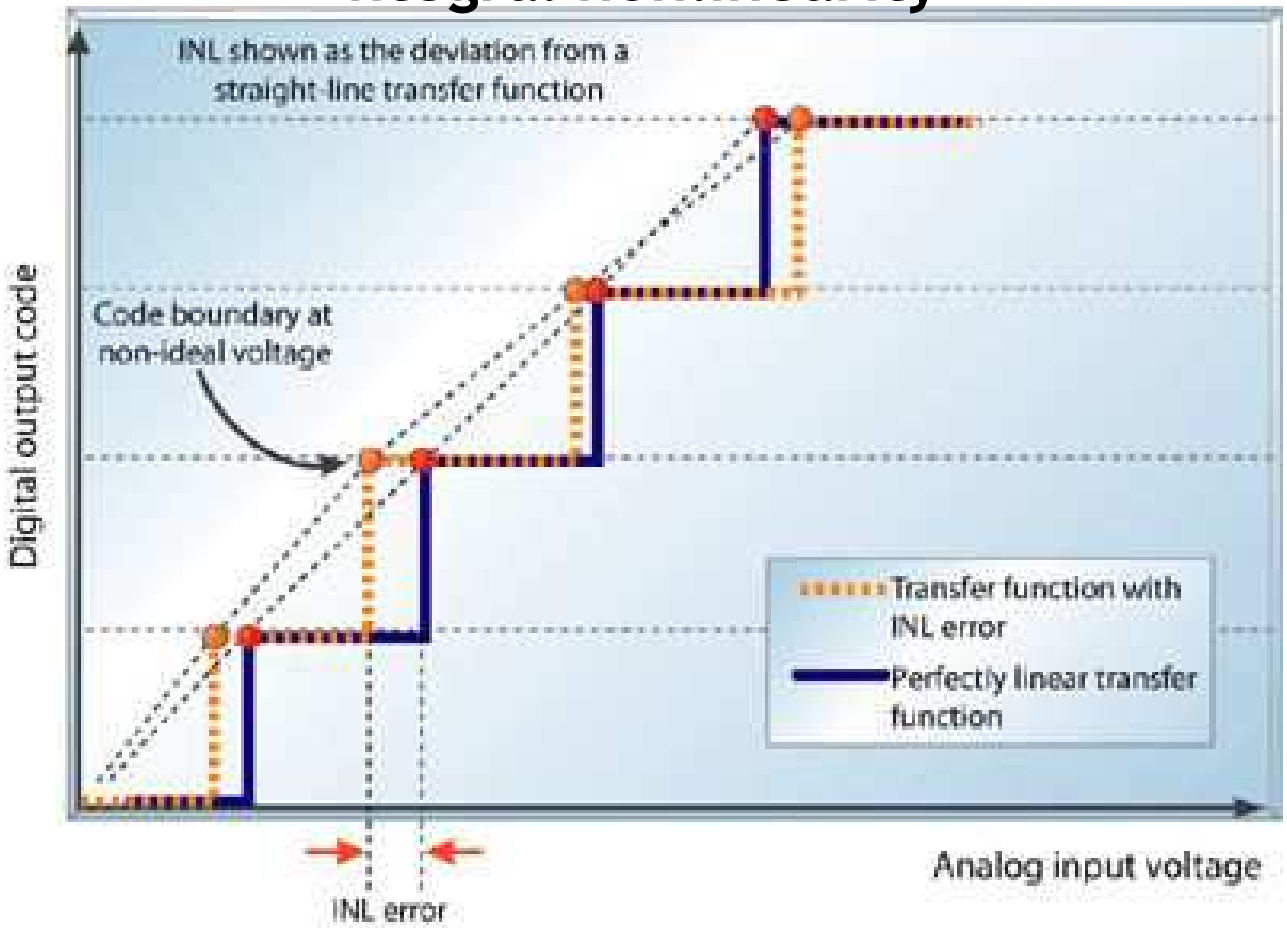
Errors and ADCs

- Figures and some text from:
  - Understanding analog to digital converter specifications. By Len Staller
  - http://www.embedded.com/showArticle.jhtml?articleID=60403334
- Key concept here is that the specification provides *worst case* values.


111
110
101
100
011
010
001
000
Digital output code
1/8
2/8
3/8
4/8
5/8
6/8
7/8
Analog input voltage (normalized to full scale)

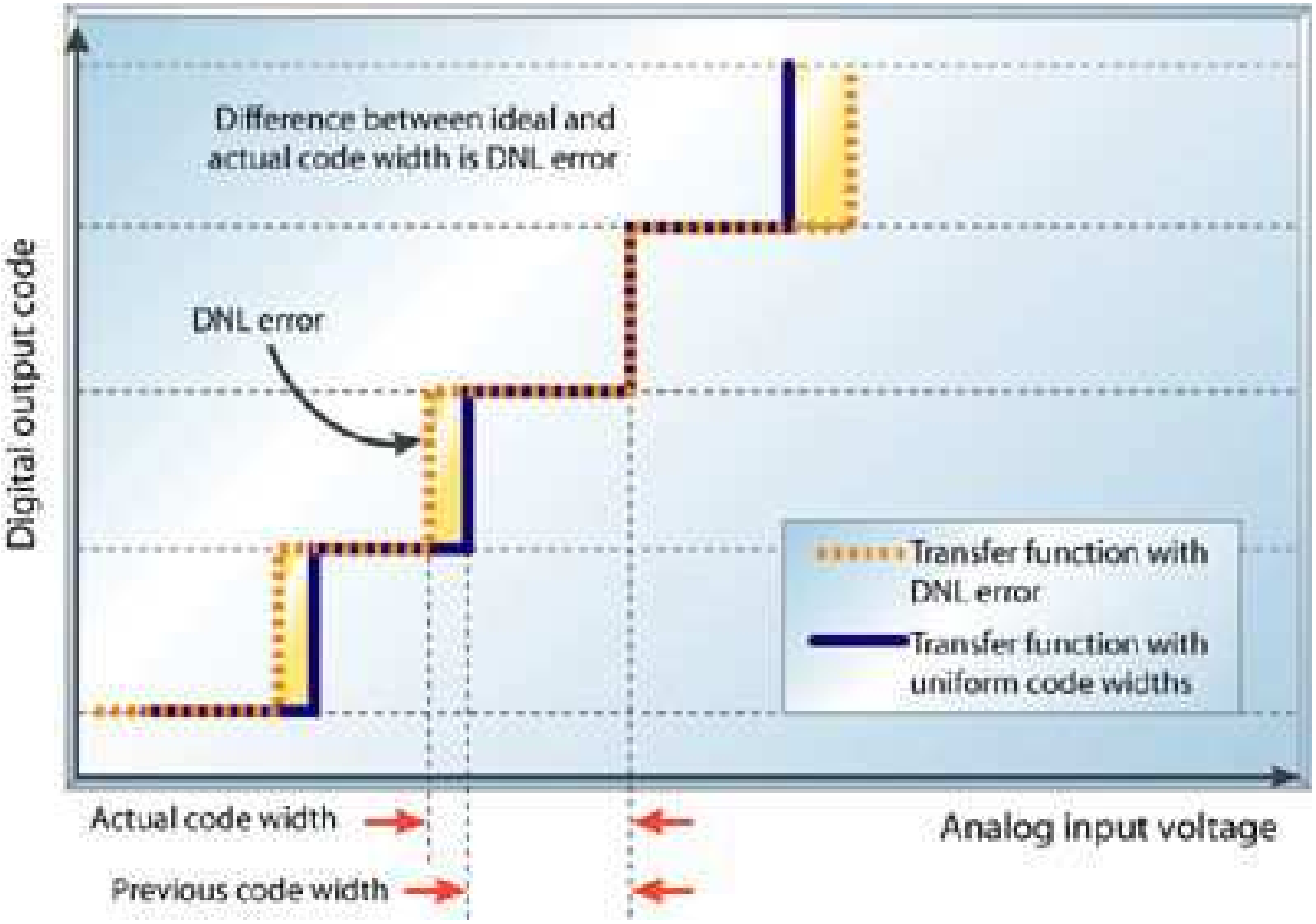

111
110
101
100
011
010
001
000
Digital output code
Transfer function shifted to the left 1/2 LSB to maintain <1/2LSB quantization error
Highest code boundary at 1 1/2LSB below full-scale input voltage
First code boundary at 1/2LSB
1/8
2/8
3/8
4/8
5/8
6/8
7/8
Analog input voltage (normalized to full scale)

Integral nonlinearity



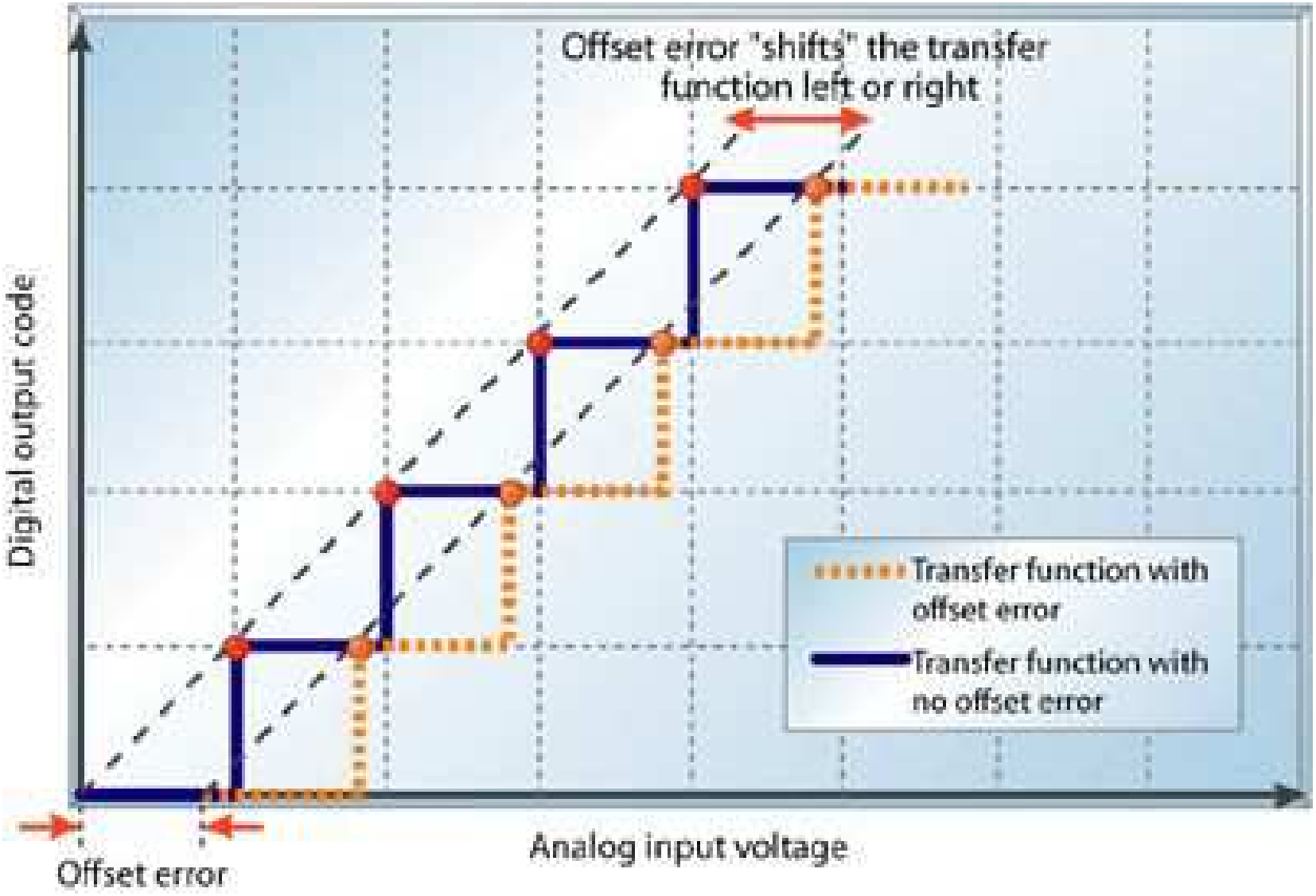
The *integral nonlinearity (INL) is the deviation of an ADC's transfer function from a* ***straight line***. This line is often a best-fit line among the points in the plot but can also be a line that connects the highest and lowest data points, or endpoints. INL is determined by measuring the voltage at which all code transitions occur and comparing them to the ideal. The difference between the ideal voltage levels at which code transitions occur and the actual voltage is the INL error, expressed in LSBs. INL error at any given point in an ADC's transfer function is the accumulation of all DNL errors of all previous (or lower) ADC codes, hence it's called *integral* nonlinearity.

Differential nonlinearity

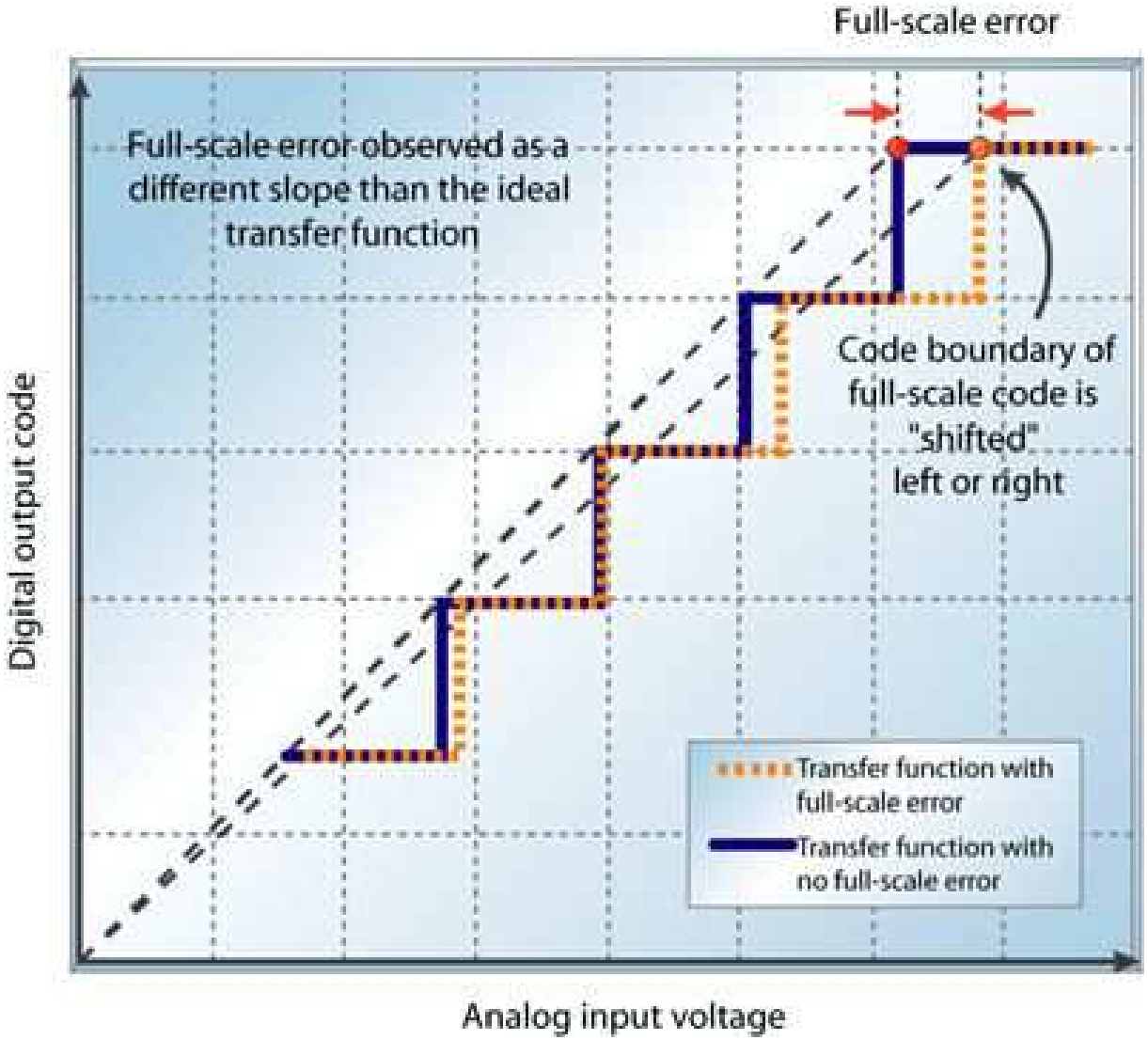


DNL is the worst cases variation of actual step size vs. ideal step size.

It’s a promise it won’t be worse than X.


Offset error "shifts" the transfer function left or right
Digital output code
Transfer function with offset error
Transfer function with no offset error
Analog input voltage
Offset error

Full-scale error is also sometimes called “gain error”



Full-scale error is the difference between the ideal code transition to the highest output code and the actual transition to the output code when the offset error is zero.

Errors

- Errors in a specification are bad.
  - So if you have an INL of ±0.25 LSB, you “know” that the device will never have more than 0.25 LSB error from its ideal value.
  - That of course assumes you are operating within the specification.
    - Temperature, input voltage, input current available, etc.
- Integral nonlinearity and differential nonlinearity are important.
  - Should know what full-scale error is.
  - Can compensate in software.
    - Where is best place to compensate?

Outline

- ~~Context and review~~
- ~~Project and midterm schedule~~
- ~~ADC and DAC operation and errors~~
- Prototyping
- Memory
- PCB design

Prototyping: why?

Get this wrong and you won't finish your design or find bugs in it.

Prototyping: what mpt,s;;y happens

- Somewhat O.K. design.
- Prototyping errors dramatically increase space for bugs to hide in.
- Days to weeks of debugging.
  - Finding mixture of prototyping flaws and design errors.

Prototyping: what should happen

- Somewhat O.K. design.
- Methodical, flawless prototyping dramatically reduces hiding spaces for bugs while increasing prototyping time by only minutes.
- Hours of debugging.
  - All of it on design errors and (very rarely) faulty components.

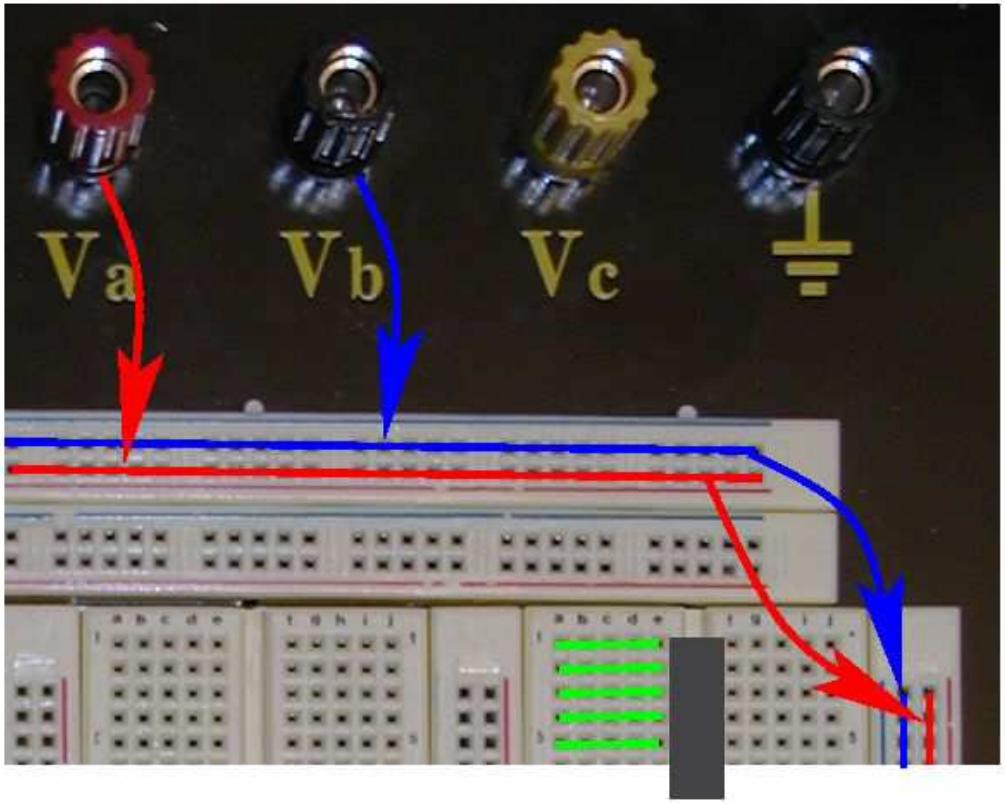
Be obsessive about knowing your tools!

Prototyping topics

- Options.
  - Breadboards.
  - Soldering.
  - Wire wrap.
- Chilling.
- Noise sources.
- The guild handshake of computer engineers.
- ESD.

Breadboards

- Quick.
- Easy.
- Horrible.
  - Unreliable.
  - Low-frequency.
    - <=1 MHz usually safe.
    - 10 MHz works, on good days.
  - Nasty parasitics.



Soldering

- Slow.
- Requires skill.
- Reliable.

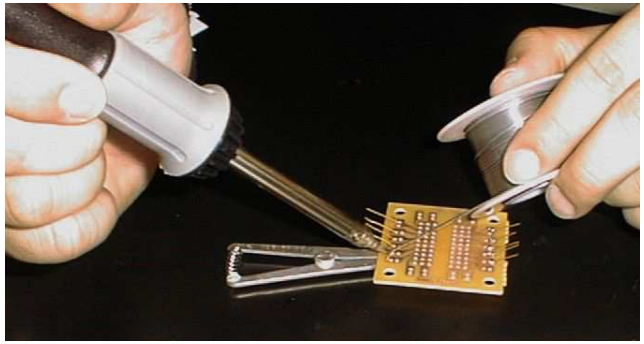
Images from Adafruit.

Soldering tools

- Soldering iron.
  - Sheath: Check before heating.
  - Temperature: 370 degrees C is a good guideline for through-hole.
  - Allow to fully heat before starting.
  - Don't leave on unnecessarily.
    - Top oxidizes.
  - Keep clean and tinned.
- Sponge or scrubber.
  - Used in tinning process.

Soldering tools

- Solder sucker used for desoldering.
- Wick used for desoldering.
- Tinning block.
- Heatsink: used to prevent component damage.



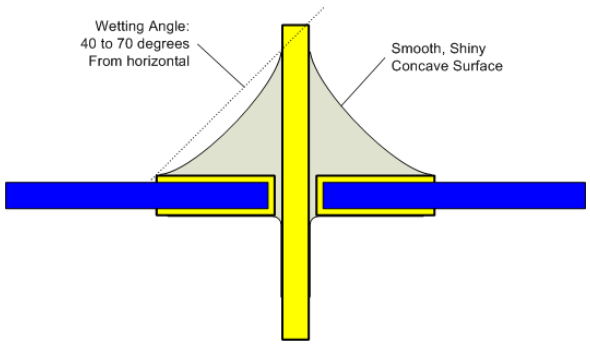
Tinning

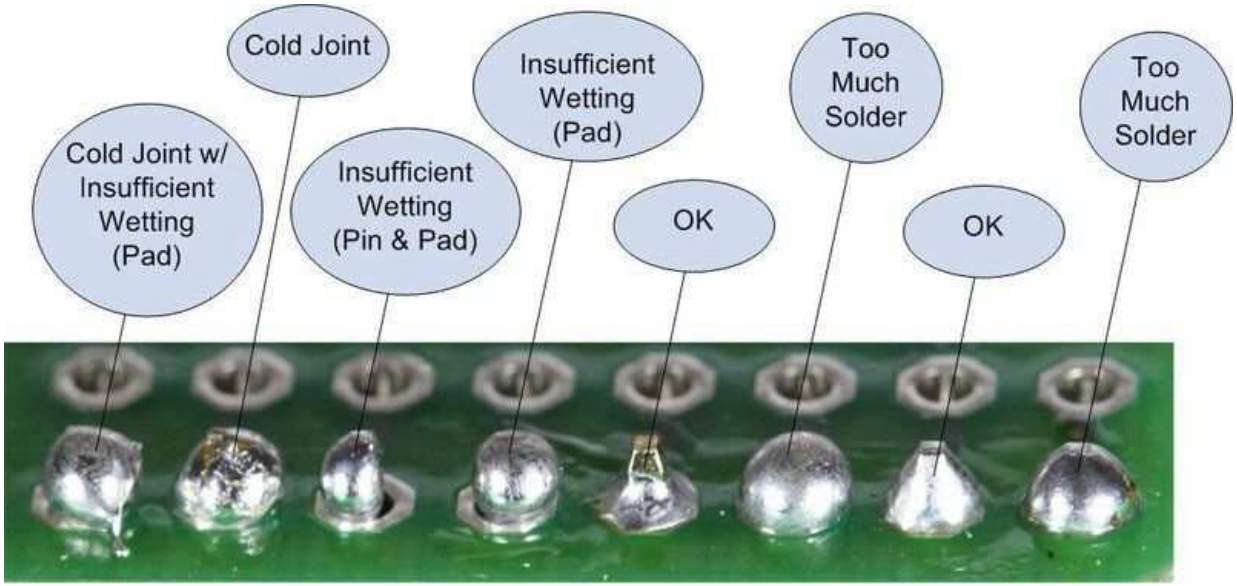
- Oxides on iron prevent adhesion of solder.
- Clean using sponge, scrubber, or tinning block.
- Apply tinning agent or solder.
  - Use solder sparingly when making direct contact with iron. Rosin can etch iron.
  - Should leave surface coated and shiny.
- Monitor: reclean and tin whenever oxides appear.
- Don't do this more frequently than necessary.
- Always do it before starting.

Soldering

- Oxides on traces and leads can prevent capillary action.
- What soldering **is not!**
  - Melting solder and letting it drop onto traces and leads.
- What soldering **is!**
  - Heating traces and leads, allowing solder to wick into gaps due to surface tension.
- Iron should come into contact with trace and lead first, to preheat them.
- Solder should touch trace/lead junction and flow.
- Cam use heatsink on delicate components.

Soldering


Wetting Angle:
40 to 70 degrees
From horizontal
Smooth, Shiny
Concave Surface


Cold Joint w/
Insufficient
Wetting
(Pad)
Cold Joint
Insufficient
Wetting
(Pin & Pad)
Insufficient
Wetting
(Pad)
OK
Too
Much
Solder
OK
Too
Much
Solder

Prototypers creed

This is my soldering station.

There are many like it, but this one is mine.

My soldering station is my best friend.

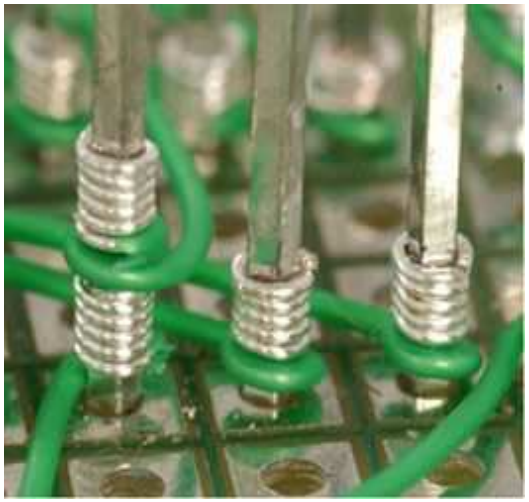
I will keep my soldering iron tinned and my traces and leads clean of oxides so the solder can flow and wick like water into every gap.

Every joint will be a uniform spire of success, not a ball of failure.

Desoldering

- Heat solder in joint and use solder sucker.
- Can clean with wick.
  - Heat the wick when in contact with joint.
- This is a pain and soldering is somewhat slow and involved even if you are good at it.

Wire wrap



- Amazingly fast, once practiced.
- Requires skill.
  - Lost art.
  - Many people don't know how.
- Reliable.

Wire wrap

- Measure strip length using too.
- Strip using slot.
- Insert wire to insulation in off-center hole.
- Put center hole over pin.
- Spin clockwise between fingers with **very gentle** pressure on back of tool.
- To unwrap, use opposite end counter-clockwise.
- Snip, leaving enough for stripping other end.
- Very slow at first.
- After 100, shockingly fast.
- Reliable. Thick.

Wiring trick

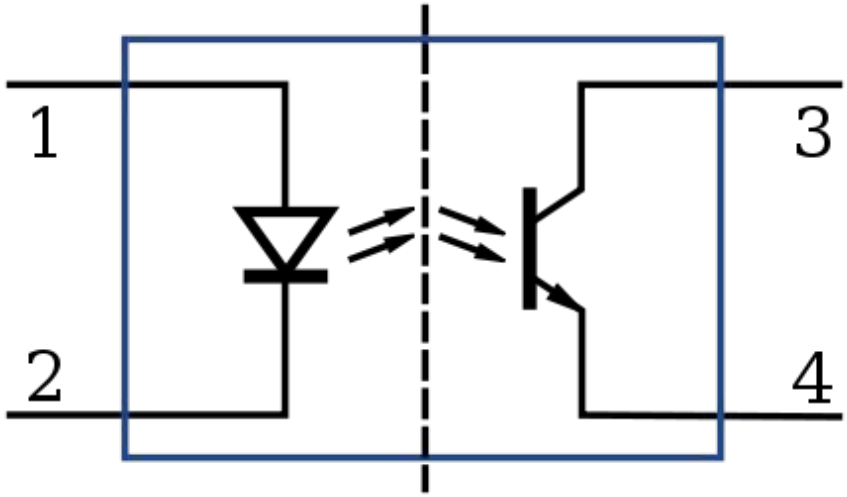
- Black: ground.
- Red: Vdd.
- If you only have two other colors to spare, distinguish odd/even, not data/address.
- Now you can actually trace the wires from pin to pin instead of getting lost in a spaghetti sea.

Noise sources

- Capacitive coupling.
  - Avoid long, || wires or planes unless you want capacitance.
  - Pay attention when low-voltage and high-voltage signals run close to each other.
- RF
  - Inductive coupling / antenna effects.
  - Less local than capacitive. Can be harder to debug.
  - May require shielding.
- High-f noise sometimes easy to filter with ferrite beads.

Noise sources

- Motors are bad.
- Solenoids and mechanical relays are often worse.
- Sparking is a bad sign.
- Common to need to isolate noise source and computer.
- Independent power supplies.
- Transistors for signaling, or even better opto-isolators.



Noise rules of thumb

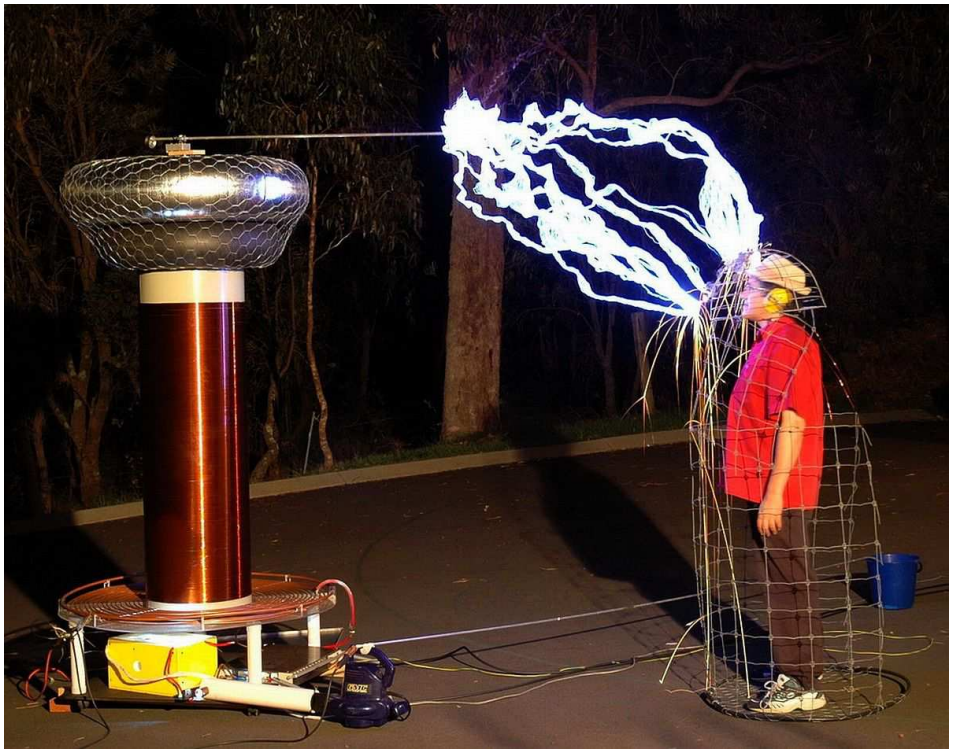
- Small motors.
  - Independent power supplies.
- Big motors, solenoids, and sparky relays.
  - Add independent opto-isolators.
- May need conductive shielding, too.
- Sometimes you can live with noise via clever design.
  - Reboot from safe memory.
  - ECC.

Faraday cages

- A conductive sphere cancels the effects of external electrical fields.
- Mesh works for most fields we would care about.
- Needs to be highly conductive. Iron doesn't work well. Cu, Al do, but Al hard to connect electrically due to all the sapphire.

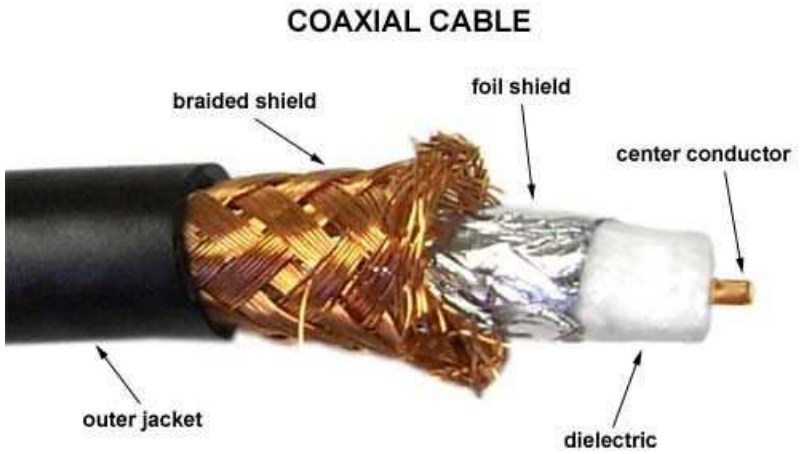
Positive: Safe from electrocution and hearing damage.

Negative: X-rays mutating his DNA. Ozone damaging his lungs.



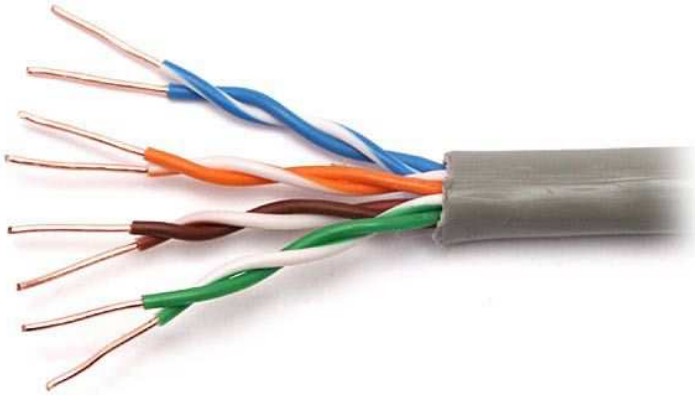
Coaxial cables

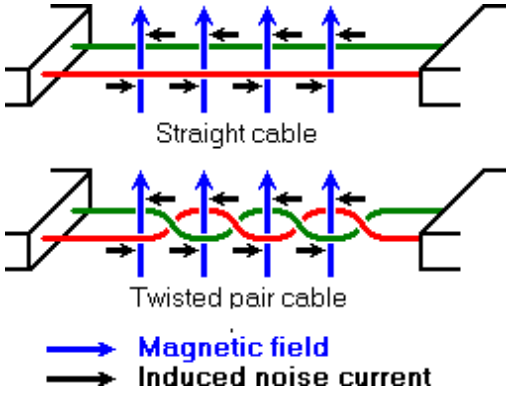
- Very high noise resistance.
- Expensive.



Twisted pair

- High noise resistance.
- Inexpensive.
- Can roll your own.





The guild handshake of computer engineers

- Some of you have high potential (7,000 V).
- Others have low potential (0V).
- Your potential changes a lot over time.
- If ESD sensitive component is on path between high-potential and low-potential student, it may be damaged or destroyed.

- Handshaking protocol
  - A holds component away from B in one hand.
  - A reaches out other hand and touches B's hand.
  - A hands over component while other hands still in contact with each other.

- Explain first to non computer engineers.
  - Avoid blushing, looking down, nervous laughter.
  - Especially today.

Electro static discharge

- High potential difference results in high momentary current through ESD-sensitive structure.
- Examples.
  - Destroy gate oxide.
  - Erase non-volatile memory locations.
- Might cause consistent faults, but might cause rare intermittent faults.
- CMOS generally more susceptible than BJTs.
- Highest risk before PCB mounting.

Electro static discharge prevention

- Least effective → most effective.
- Handshake, conductive foam/mylar, and nothing else.
- … and touching grounded equipment cases before starting work and periodically.
  - Don't touch while touching live circuit. Dangerous current path.
- … and using grounded mat.
- … and/or wearing grounded wrist strap.
- … and wearing shoe/ankle grounding straps and using grounded floor.

- Danger: Don't short to ground. Use >1 MOhm resistor.

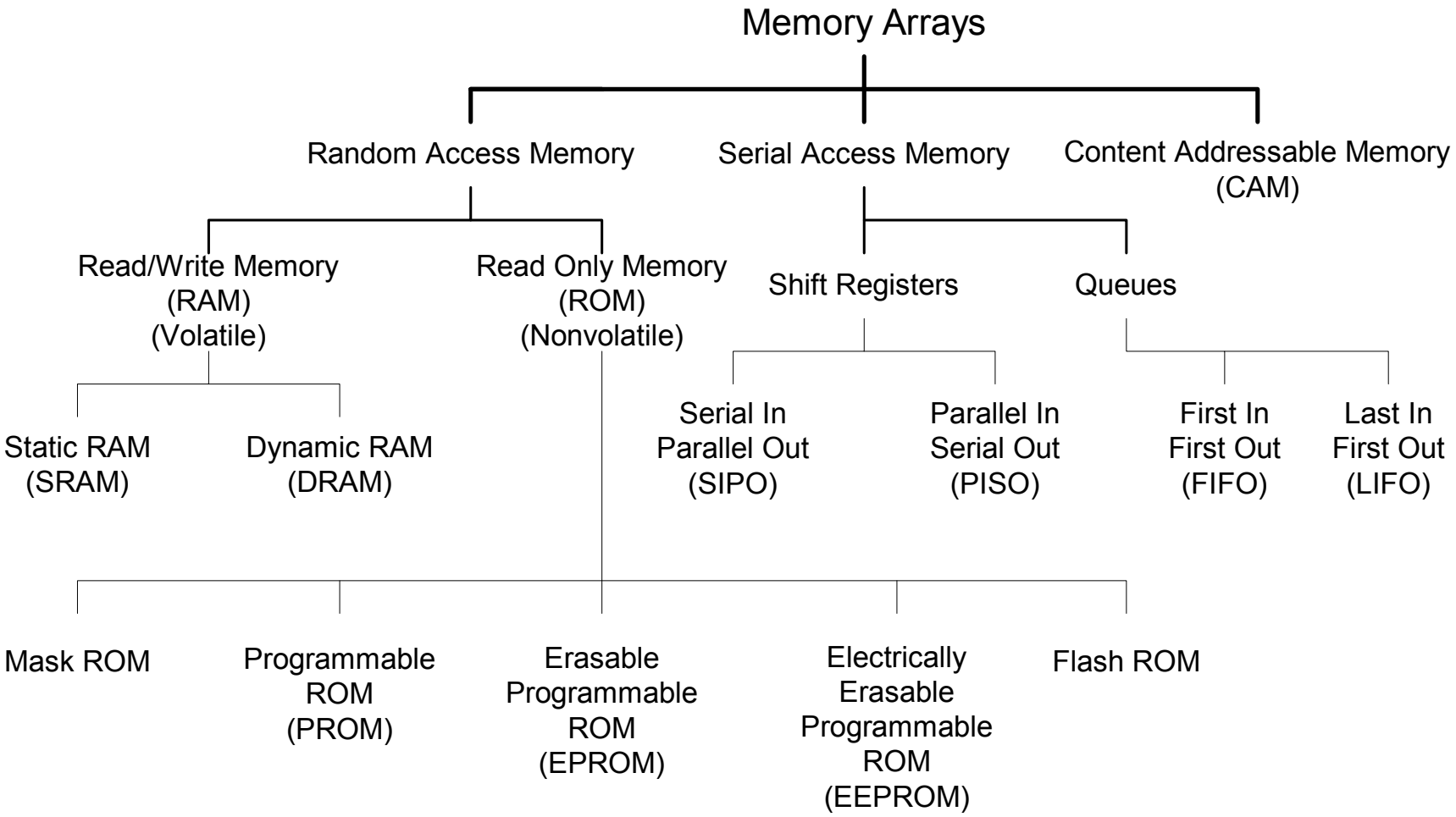
Outline

- ~~Context and review~~
- ~~Project and midterm schedule~~
- ~~ADC and DAC operation and errors~~
- ~~Prototyping~~
- Memory
- PCB design

Memory: why?

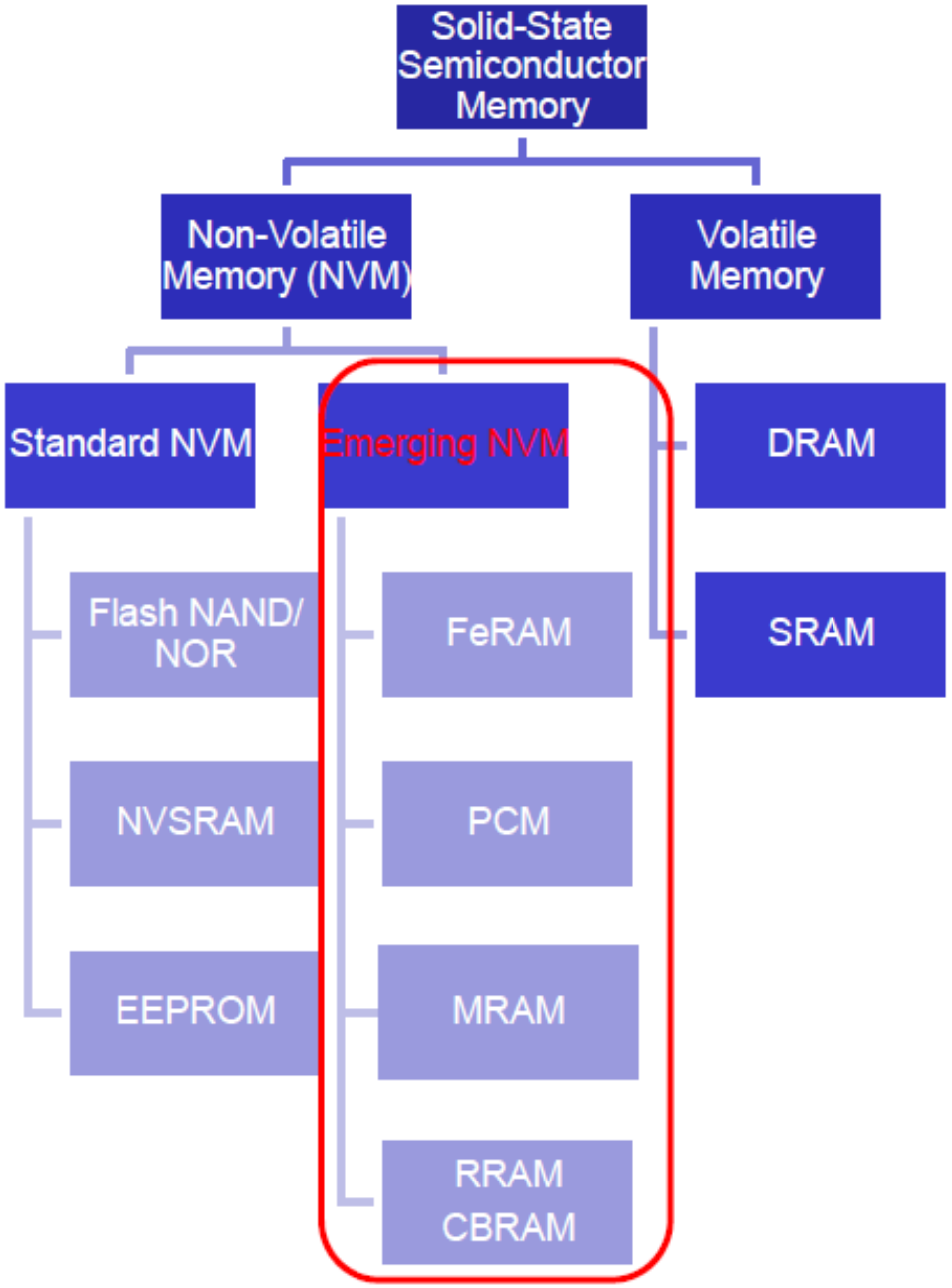
- You'll be dealing with this a lot in your projects.
- A little review now can save you trouble later.

Memory array types



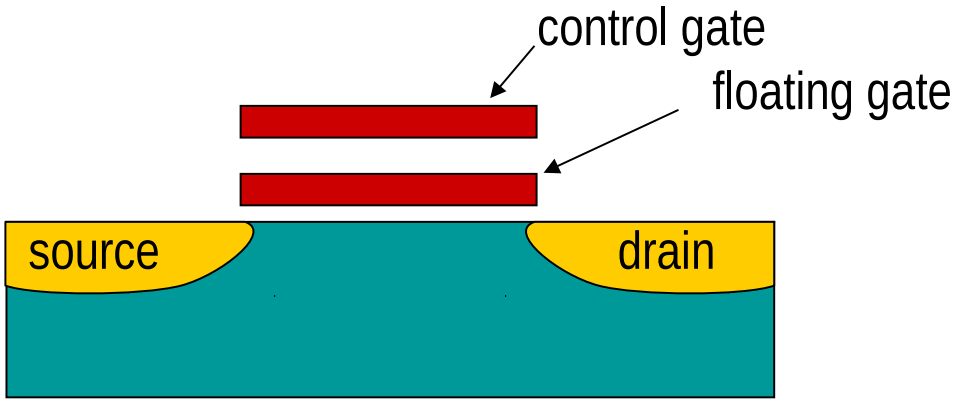
Nonvolatile memory types

- Ferroelectric RAM: polarization changed by electric field.
- Phase change memory: amorphous vs. crystalline.
- Magnetoreseistive
- RAM: state recorded in magnetic field.
- Programmable metallization cell: redox filament.
- Flash: ~ms.
- EEPROM: ~ms.
- SRAM/DRAM: ~ns.



Floating gates

- Write: hot-electron injection or Fowler-Nordheim tunneling.
  - High voltage on control gate >> operating voltage
  - Electrons are trapped in the floating gate.
  - Will not discharge for many years.
- Erase? Fowler-Nordheim tunneling.



- Read by seeing whether it acts like a transistor or a wall.
- Tend to self-destruct after 100,000 writes/erasures.

Outline

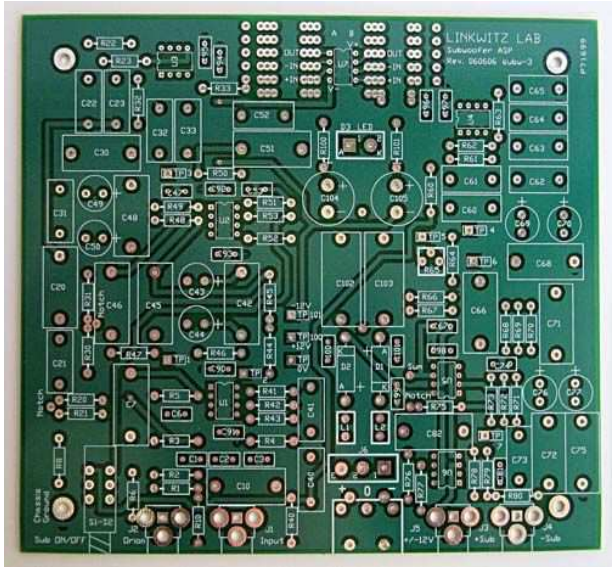
- ~~Context and review~~
- ~~Project and midterm schedule~~
- ~~ADC and DAC operation and errors~~
- ~~Prototyping~~
- ~~Memory~~
- PCB design

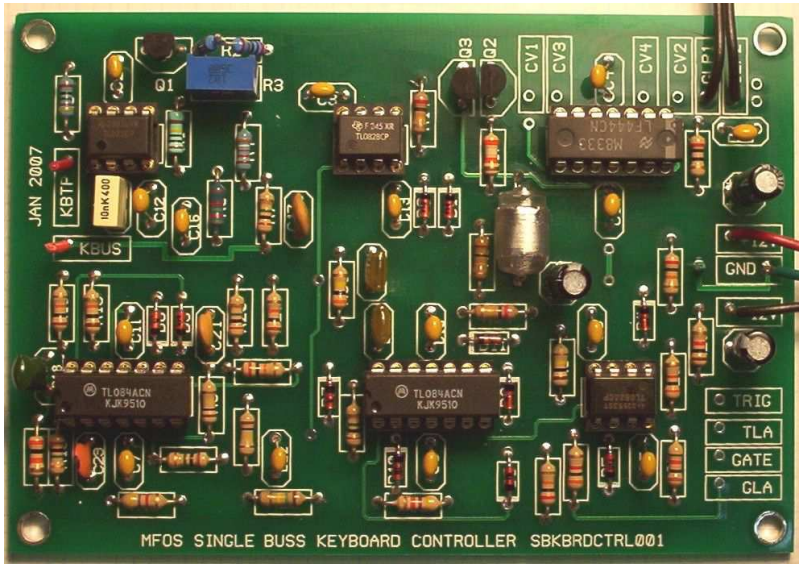
PCBs: why?

- Even if you aren't making one for your project, need to understand how they work for debugging / reverse engineering.

Printed circuit board design

- Physical support.
- Electrical connections.
  - Traces have restricted dimensionality.
  - Very thin, high resistance.
  - Holes/vias and pads.
  - Rework is hard.

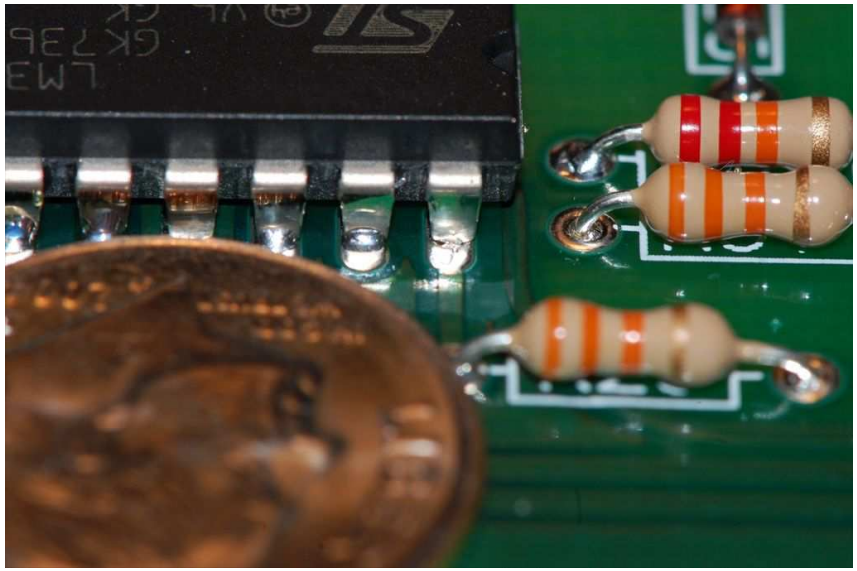


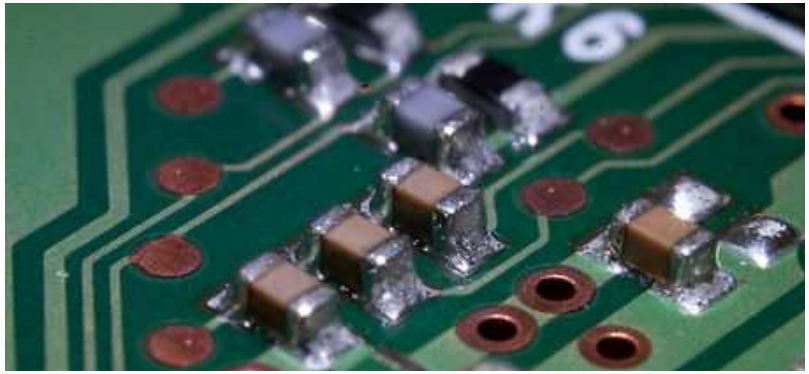


http://www.linkwitzlab.com/Pluto/supplies-subw.htm,
http://www.musicfromouterspace.com/analogsynth/SINGLEBUSSKEYBOARD2007/SINGLEBUSSKEYBOARD2007.php

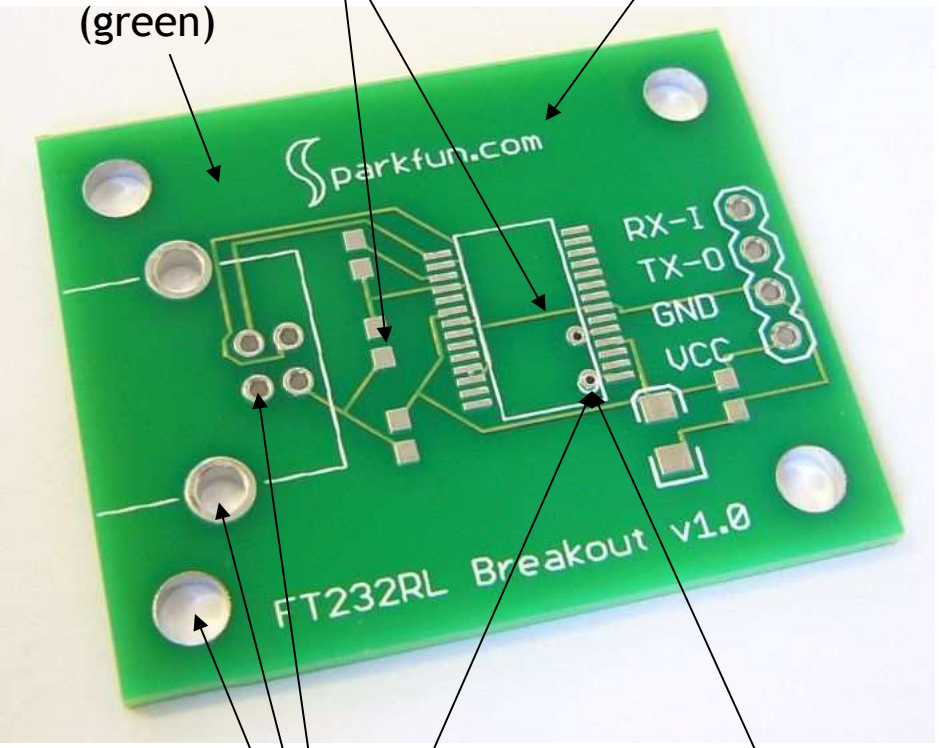
Basic terminology

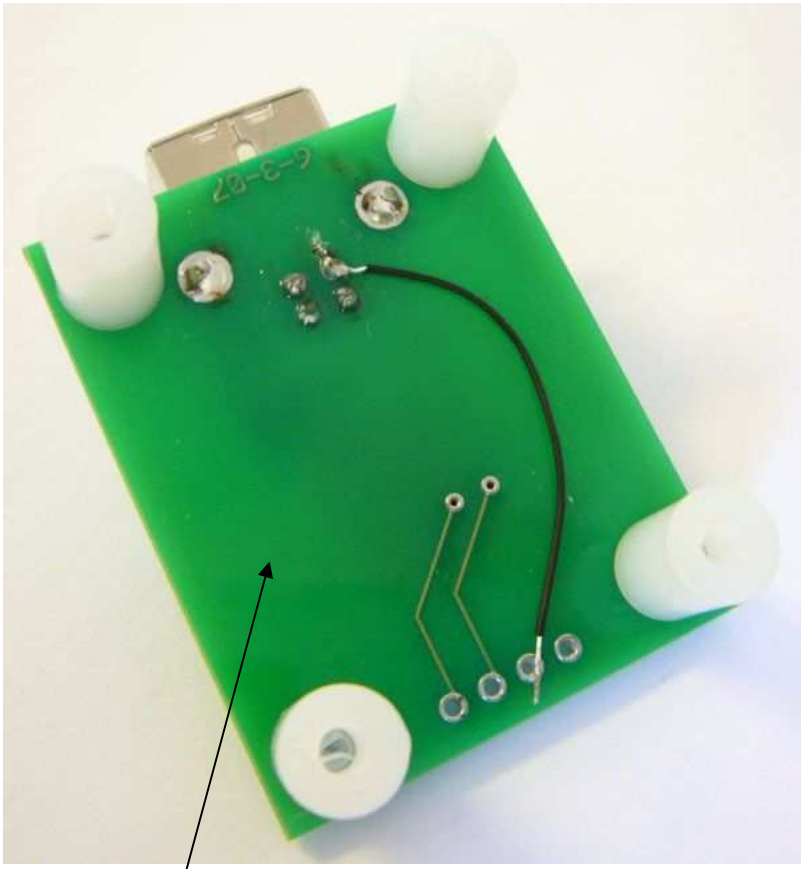
- Interconnects: traces.
- Inside of a given “layer” traces which cross are electrically connected.
- Can have multiple conductive layers by stacking/bonding boards.
- Through-hole: Having holes in the PCB designed to have pins put through the holes.
  - Contrast with surface mount where device goes on top.





Parts of a PCB





Vias

- Sometimes you need to connect traces on layers.
  - Use a via: a plated through hole
    - Generally smaller than a through hole for a pin.

Clearances

- There will be space between the traces, other traces, and plated holes.
  - You need to meet the requirement of the manufacturer.

The layered construction of a PCB: a six layer board

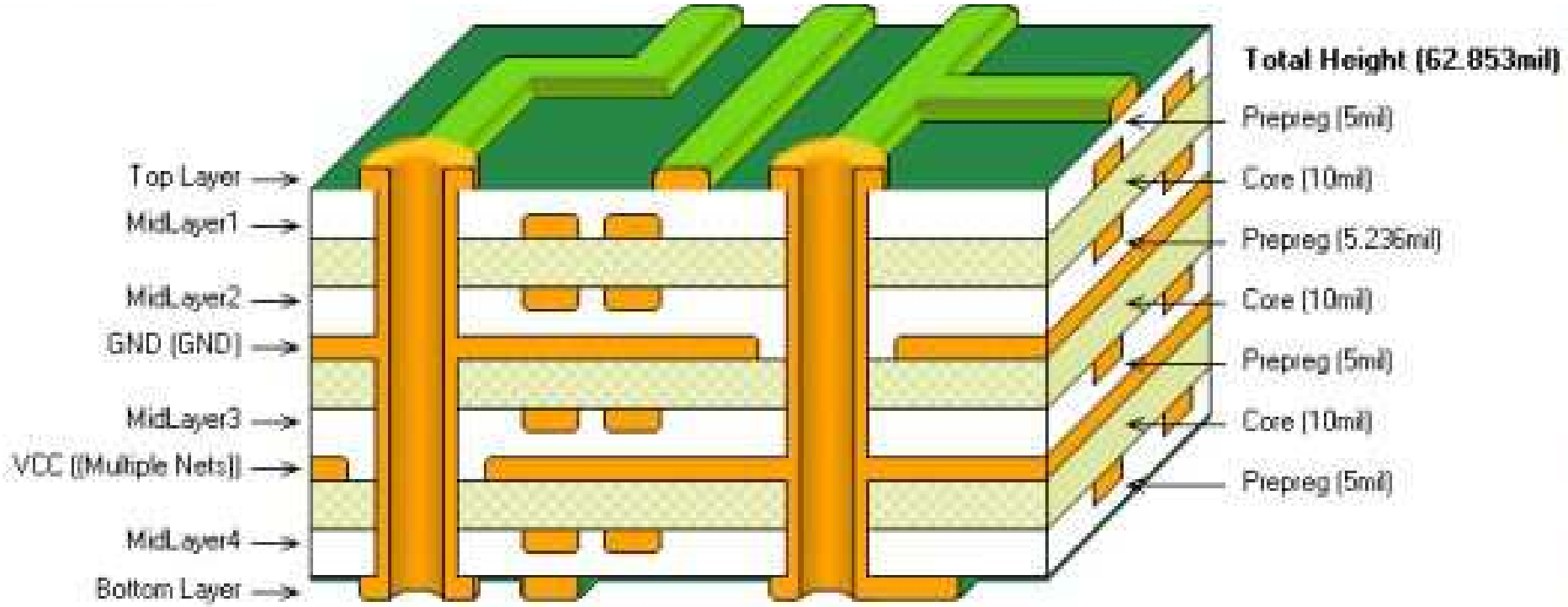


Figure from altium.com

What do do with layers?

- Orthogonal routing layers.
- Ground planes to increase local power supply capacitance and minimize resistance.
- Power plane for similar reason.
- More layers → higher cost.

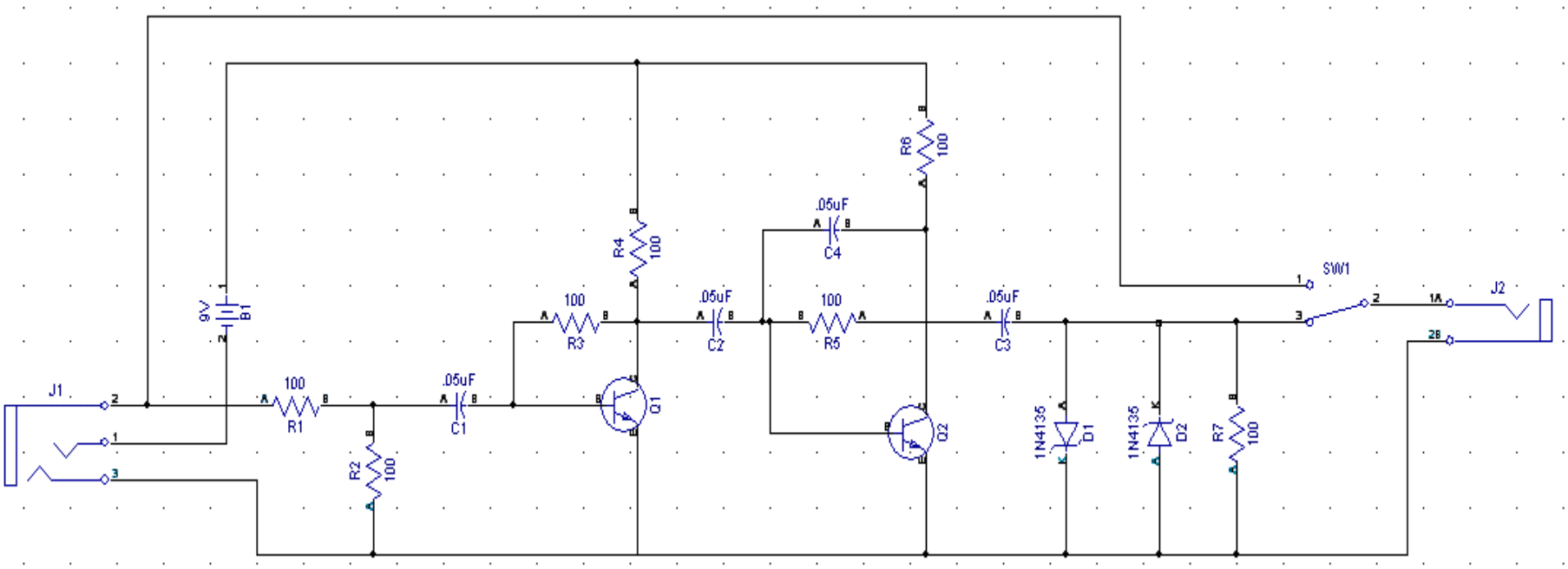
How to design PCB

1. Create schematic
2. Place parts
3. Route interconnect
4. Generate files

Step 1: Create schematic

- The first thing you want is something that looks like a textbook circuit diagram. It just shows the devices and how they are connected.
  - Sometimes you will worry about pinouts here (say when working with a microprocessor maybe).
  - But usually you don't.
- **No notion of layout** belongs here!

Example schematic



Why a schematic?

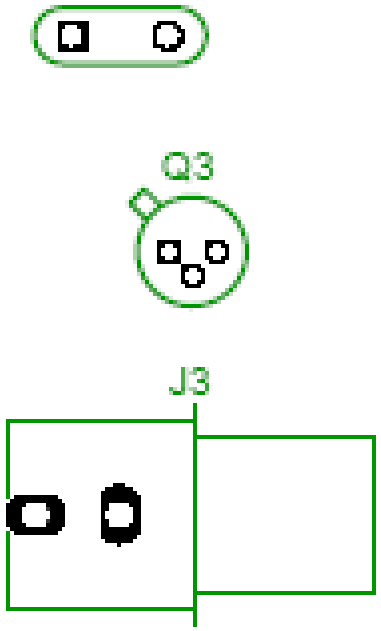
- In general it is drawn to be *readable*.
  - This is probably what your sketch on paper would look like.
  - You can find and fix bugs more easily here than the PCB layout.

Step 2: Place parts

- You need to place the patterns on the board.
  - You need to not overlap them to that the components can actually fit on the board.
  - You want to leave room for the traces to connect everything.
- This is *very* much an art form.
  - In fact you will find people who rant about “sloppy” or “unprofessional” placements.
- Some tools will do this for you.
- Sometimes they screw up.

Patterns

- Once you know what it is you want to build, you need to figure out how to lay it out on the board.
  - You need to know how big each piece is, and where the holes need to be placed.
- Each device has a pattern which shows exactly that.
  - You will occasionally need to create a pattern.



Step 3: Route interconnect

- A route is a connection between devices.
  - It may consist of multiple traces
- There are design rules which include:
  - Minimum trace width
  - Minimum spacing between traces and holes
  - Minimum spacing between holes and holes.
- These rules will vary by manufacturer.
  - Even better, *units* will vary by manufacturer!
  - Time for a brief aside…

Issues of measure

- PCB designers use odd terminology.
  - A “thou” is a thousandth of an inch.
  - A “mm” is a millimeter
  - A “mil” is a thousandth of an inch.
    - Thou is generally preferred over mil to avoid confusion, but most tools/vendors use mil.

Trace width

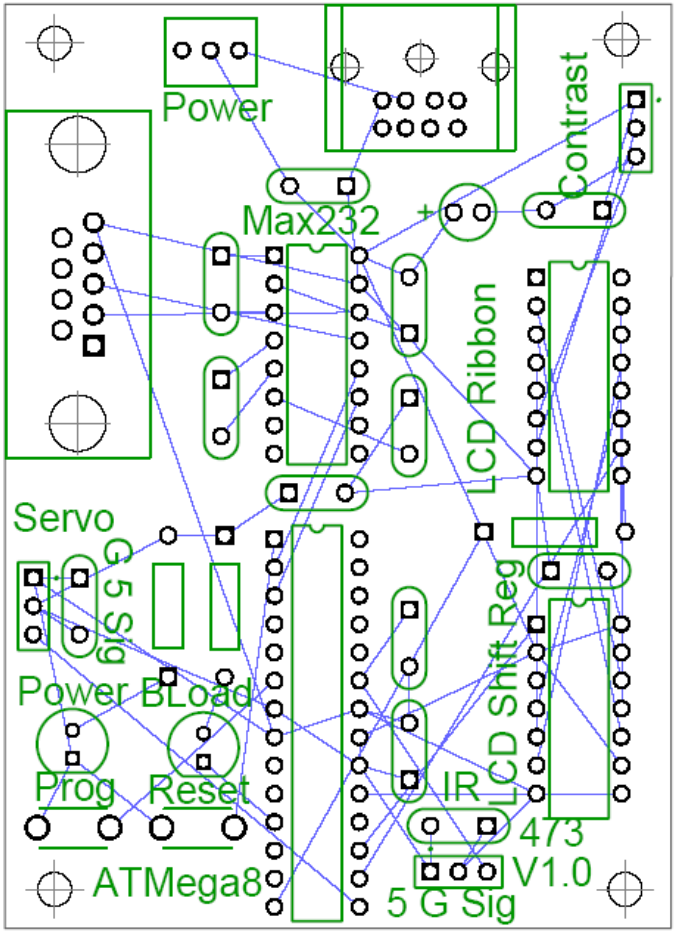
- In general most PCB manufactures seem to have trace-width minimums of 6-10 thous.
  - Most are willing to go smaller for a price.
- A rule of thumb is to use a 50 thou minimum for power/ground and 25 for everything else.
  - This is to drop the resistance of the traces.
  - In general you are worried about heat dissipation
- There are lots of guidelines for width/power but in general you are looking at:
  - A 10cm trace needs to be 10 thou wide if it will carry 1 amp.
  - 5 amps at 10cm would require 110 thou.

Trace width continued

- The problem with wide traces is that they are hard to route.
  - In particular you might wish to go between pins of a device.
- One solution is to be wide normally and “neck down” when you have to.
  - This is more reasonable than you think.
    - Think resistors in series.

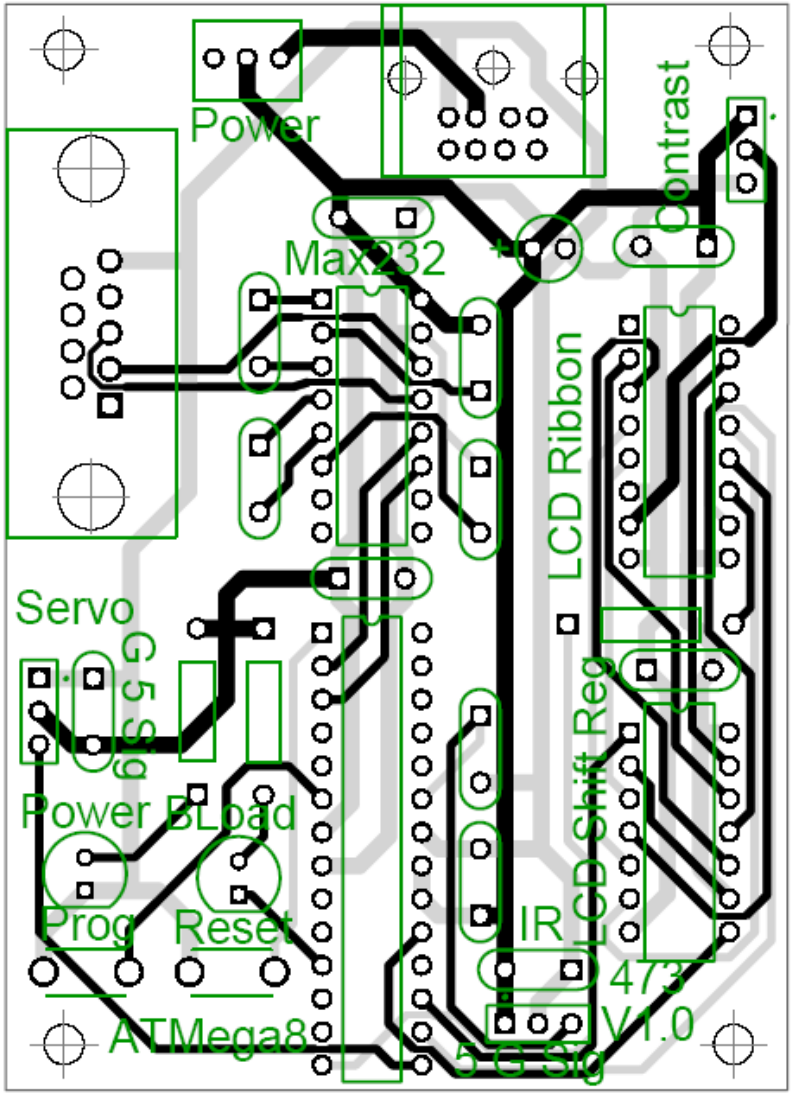
Rat's nest.

- A rat's nest shows the placement of the devices **and** the connections but not the routing
  - Automatically generated for you.
    - Sometimes before placement, sometimes after
      - Varies by tool.

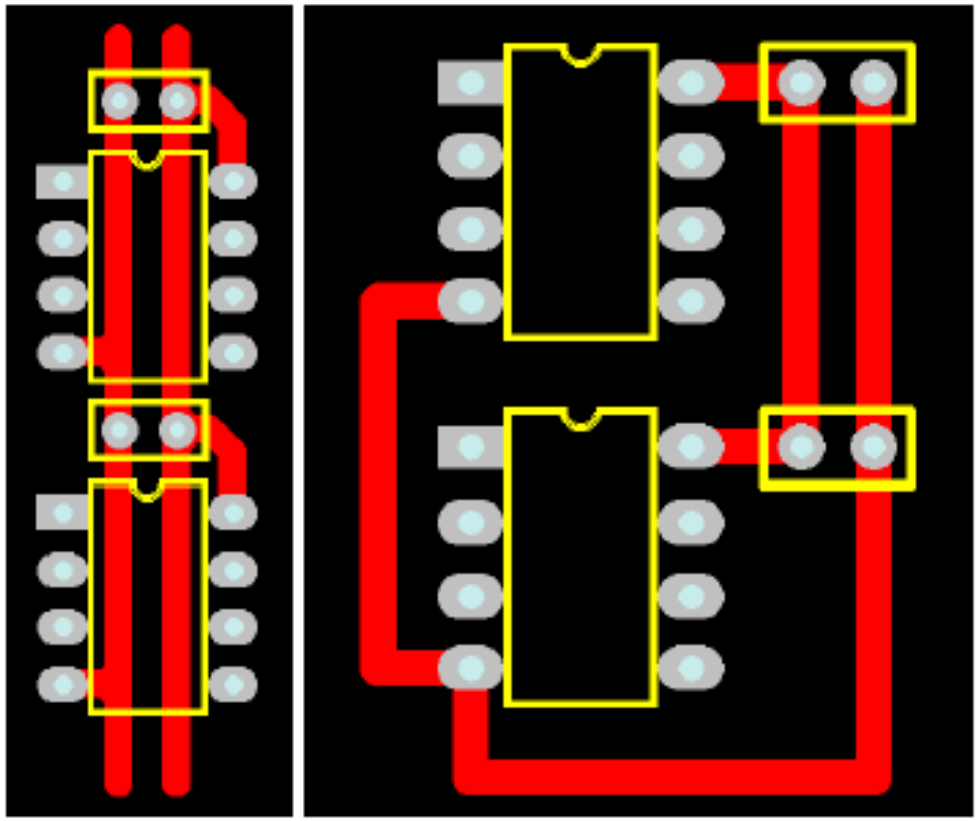


Routing for real

- You can use an autorouter to route your traces
  - Some people hate these as the design will be "ugly".
  - Saves a lot of time.
  - Oddly, not always as good as a person can do.
    - But much faster.
- Still generally need to do some (or all) of the routing by hand
  - Very, very tedious.



Routing quality

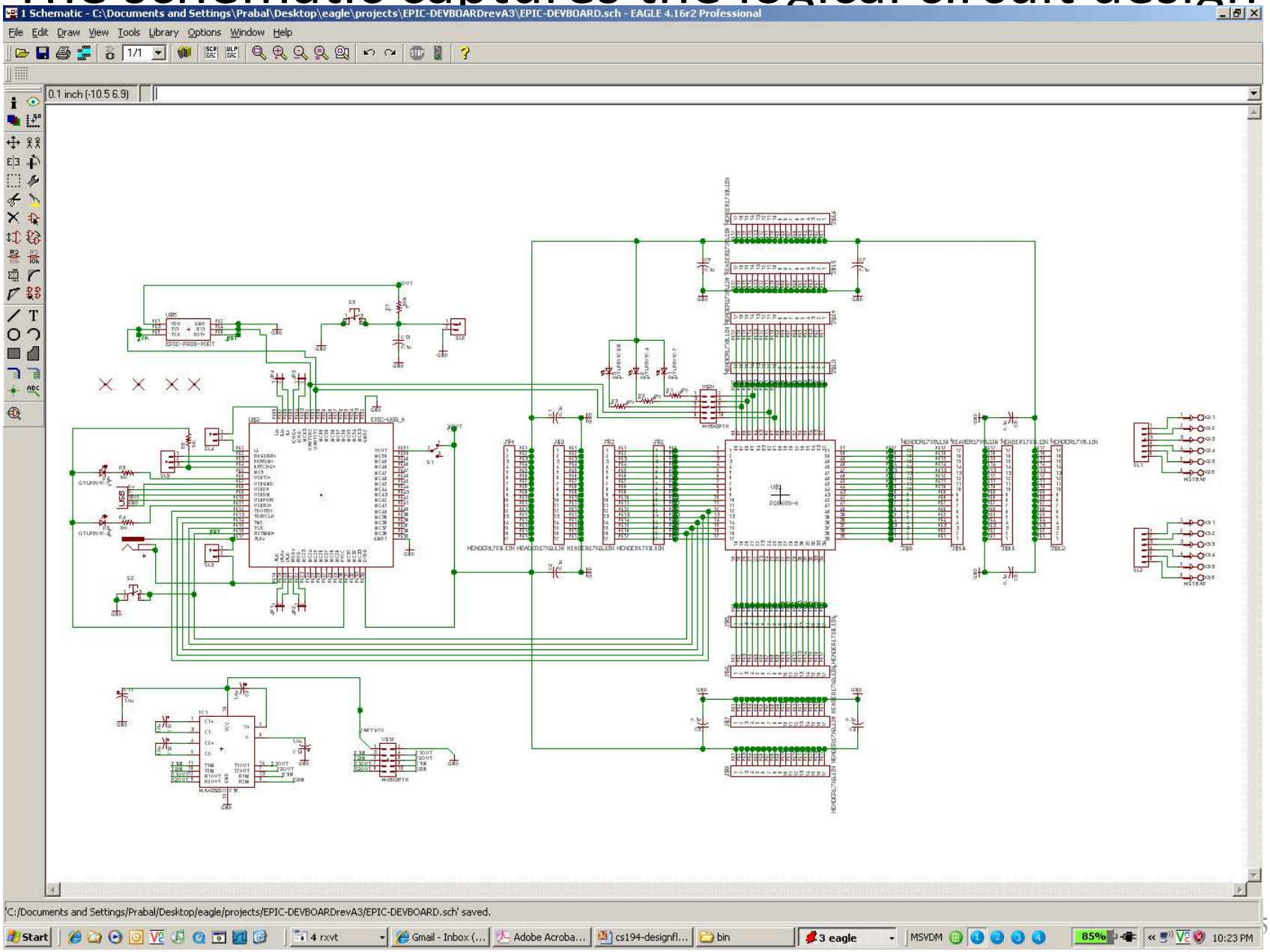


An example of GOOD power routing (Left) and BAD power routing (Right)

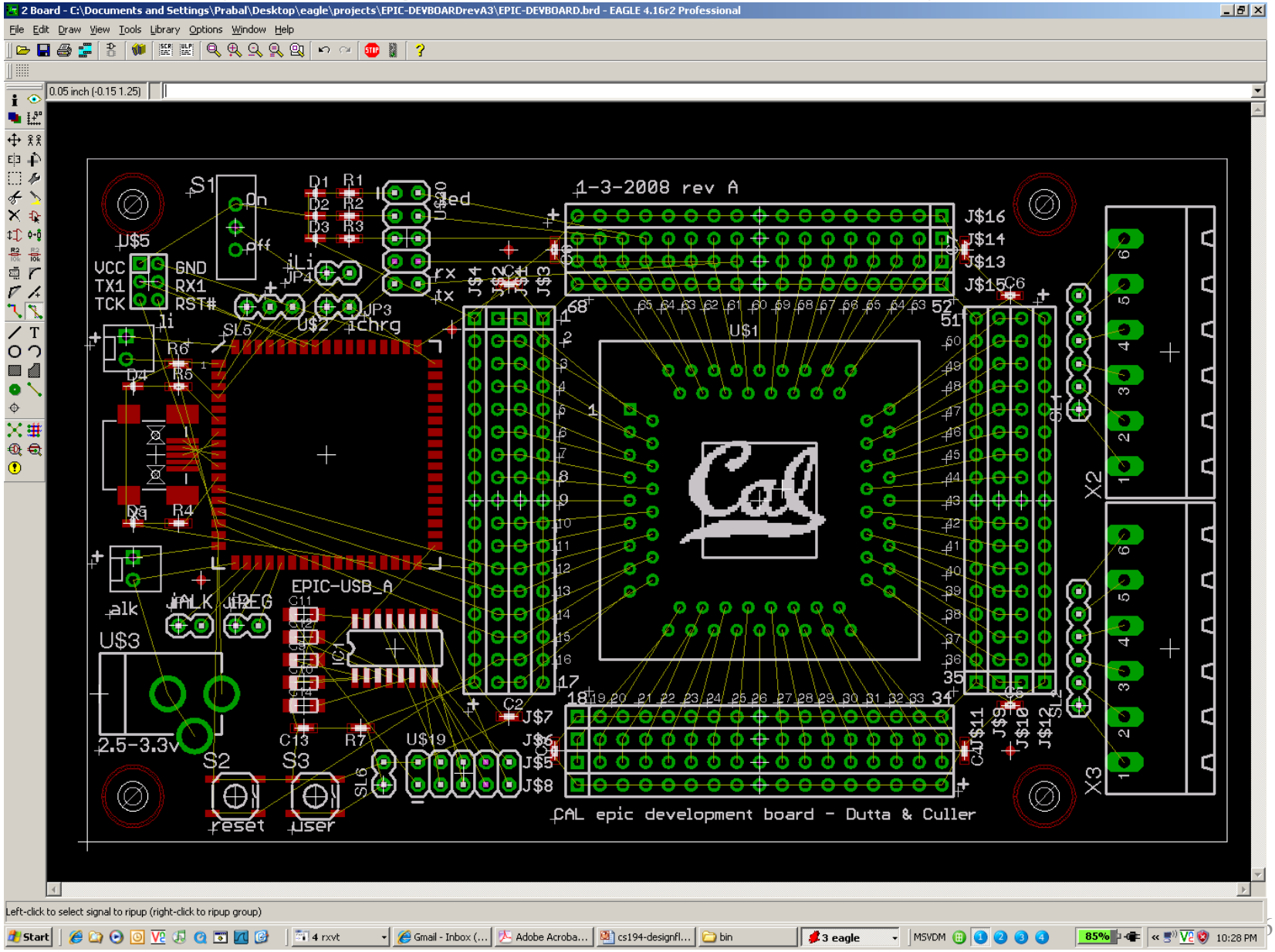
Step 4: Generate files

- Once the design is done, a set of files are generated.
  - Each file describes something different
    - Copper on a given layer
    - Silkscreen
    - Solder mask
  - Most files are in “Gerber” format
    - Human-readable (barely) ASCII format
    - Has commands like *draw* and *fill*.
  - Drill files are a different format called Excellon
    - Also human-readable (barely) ASCII with locations and diameters for the holes.
- Generally you zip all these files up and ship them as a single file to the PCB manufacturer.
  - Often a good idea to include the design file(s) too.

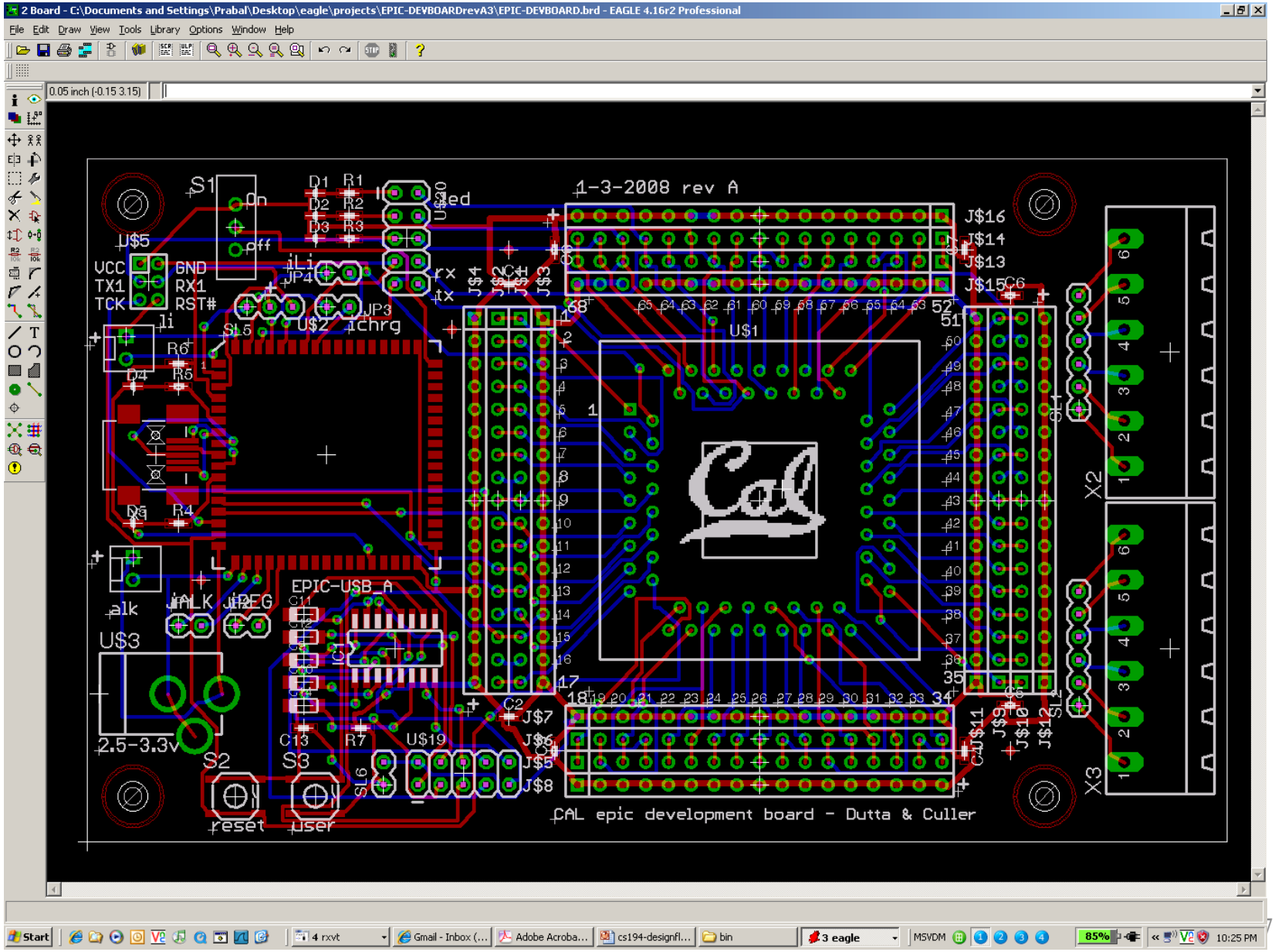
The schematic captures the logical circuit design



Floorplanning captures the desired part locations



The auto-router places tracks on the board, saving time



The layered construction of a PCB: A six layer board

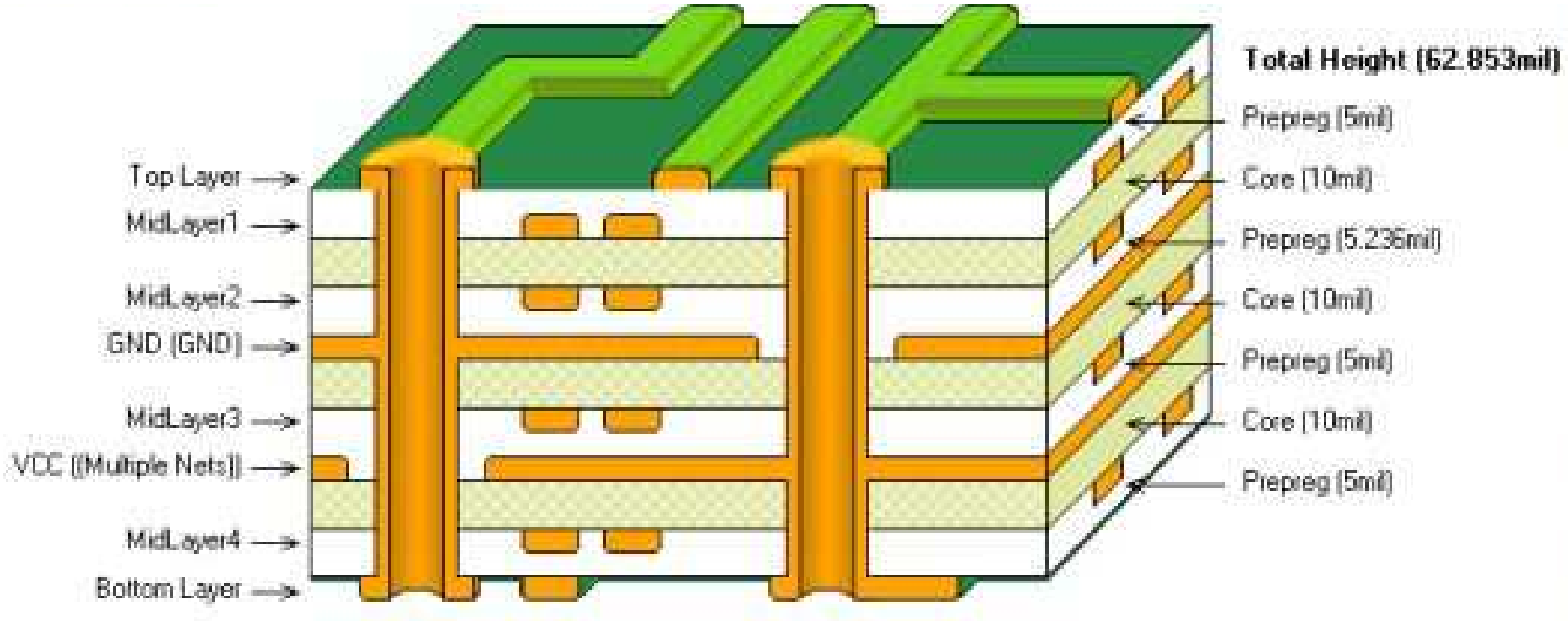


Figure from altium.com

Doesn't need to be expensive / complex

- Can CAD/CAM mill away solid Cu layer.
- Can use lithography.
  - Photoresist.
  - Mask (can print with laser printer).
  - Projector.
  - Etchant (many are dangerous to breathe and touch).
  - Safe way to dispose of Cu-containing solution.