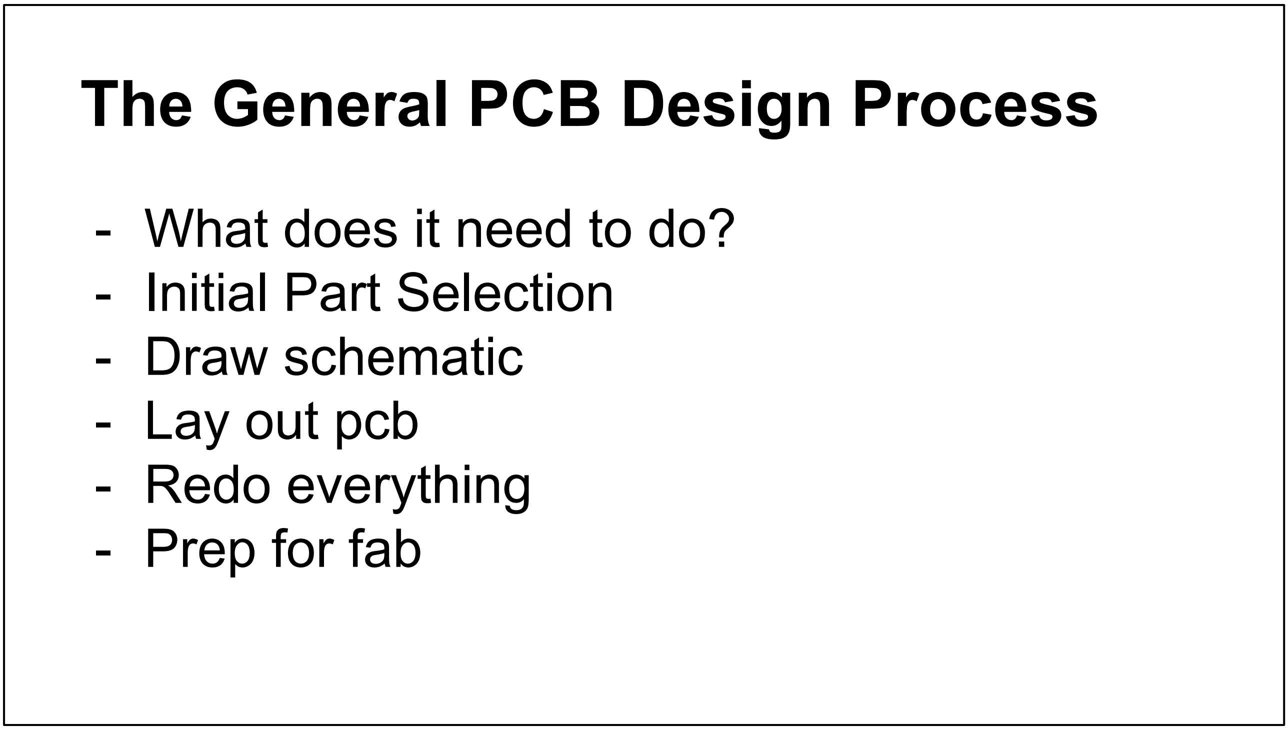
Making Better PCBs

A start-to-finish guide for getting started designing quality boards

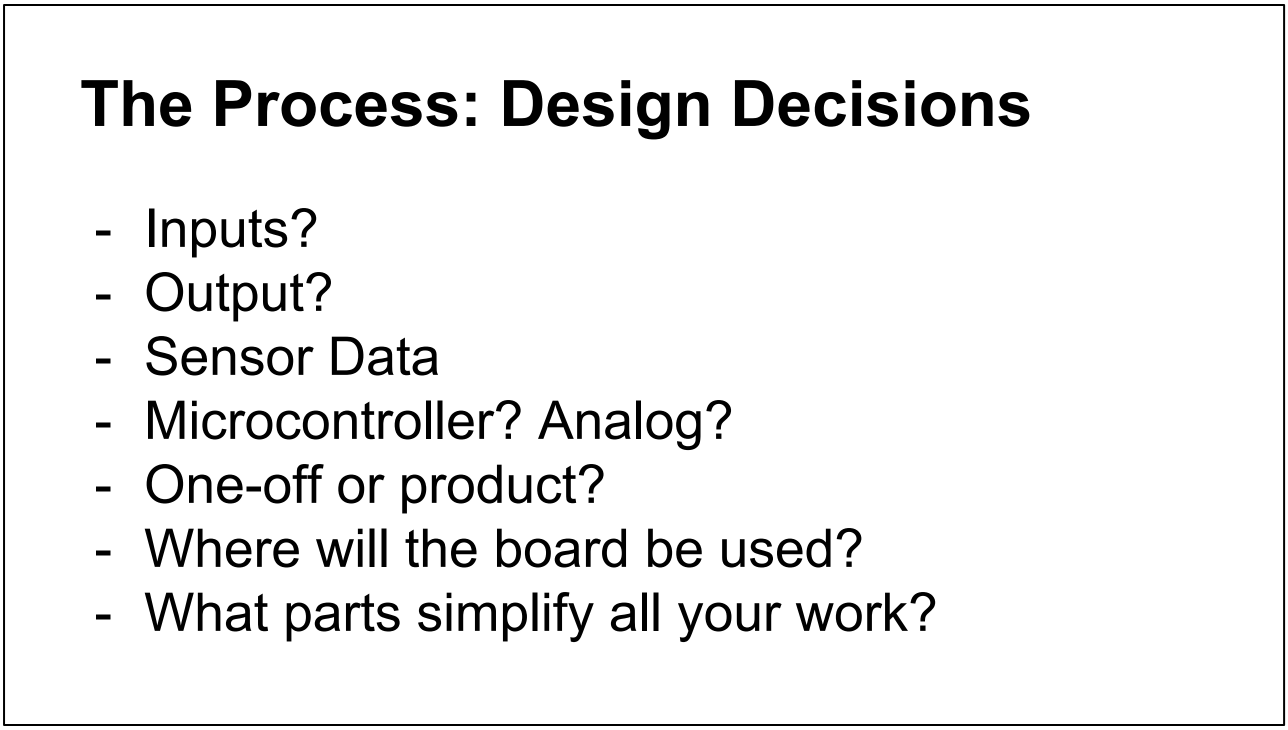
Meta: About this talk

- Kinda wordy! (Sorry)
- Breadth over depth
- Laid out for later reference.



The General PCB Design Process

- What does it need to do?
- Initial Part Selection
- Draw schematic
- Lay out pcb
- Redo everything
- Prep for fab



The Process: Design Decisions

- Inputs?
- Output?
- Sensor Data
- Microcontroller? Analog?
- One-off or product?
- Where will the board be used?
- What parts simplify all your work?

If your device is measuring sensor data, how do you get the data from it?

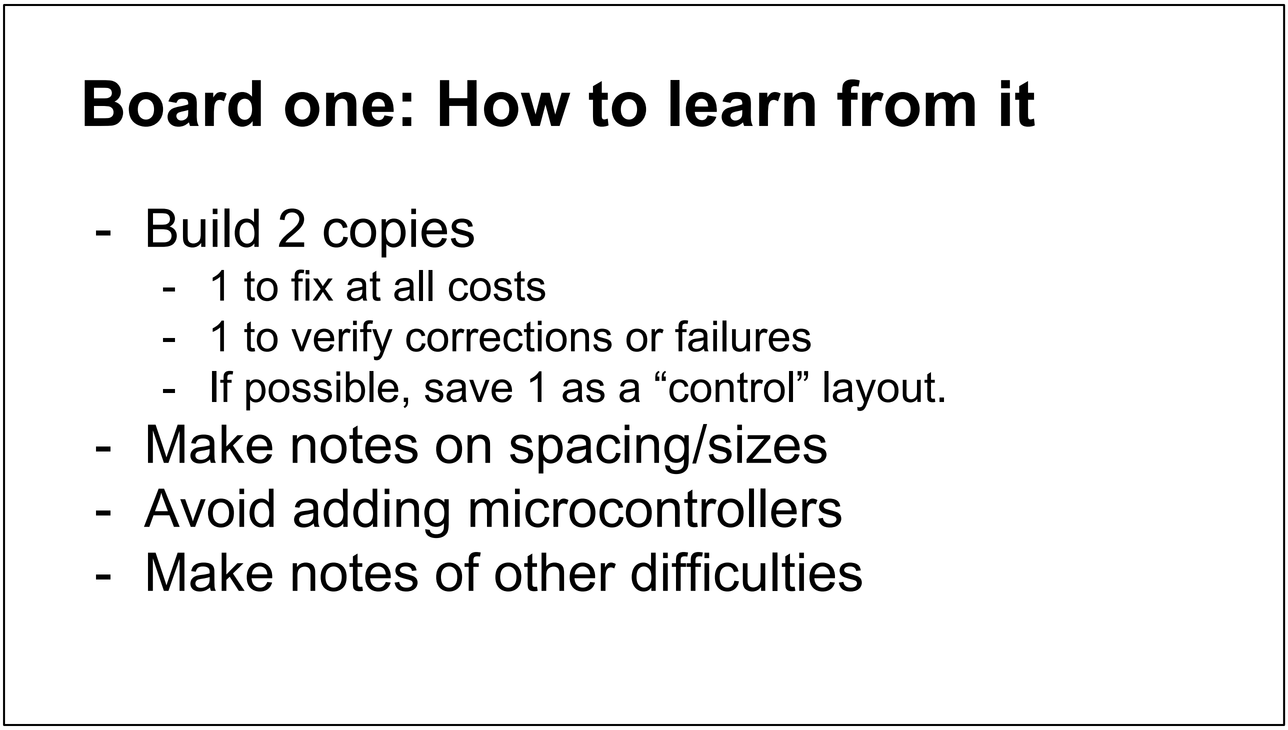
If you’re developing a one-off, you can usually get by with poor designs, terrible yields, and bad practices. You only need it to work once.

If you’re developing a product, best practices for manufacturing should be considered from the get-go.

Your first board:

- Will probably be wrong.
  - Make it cheap
  - Make it useful
- What you’ll mess up
  - Part spacing + orientation
  - Footprints
  - Via/header size
  - Bad part choices
  - Assembly

THAT’S OK!

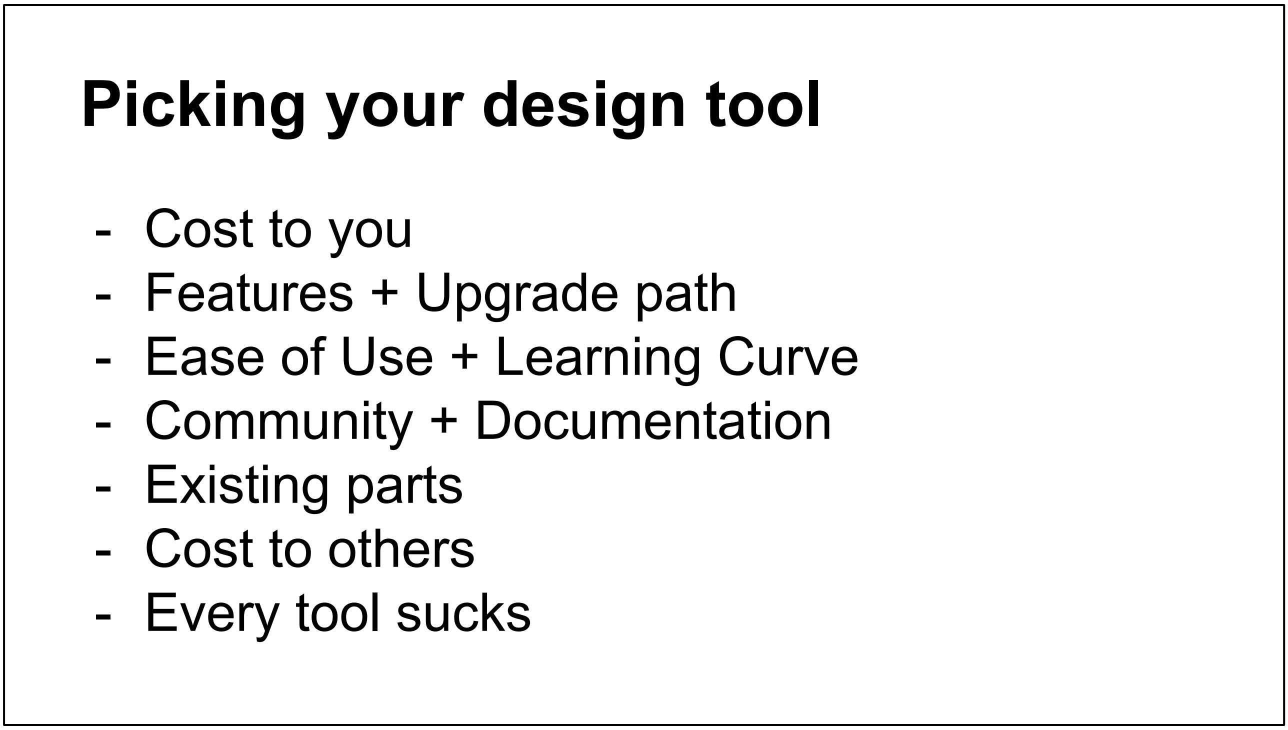


Board one: How to learn from it

- Build 2 copies
  - 1 to fix at all costs
  - 1 to verify corrections or failures
  - If possible, save 1 as a “control” layout.
- Make notes on spacing/sizes
- Avoid adding microcontrollers
- Make notes of other difficulties

Why no micros? Adds lots of complexities to determine failure.

- Bootloader
- Programmers
- Toolchains
- Code
- Often high-speed signals for clock and progamming headers



Picking your design tool

- Cost to you
- Features + Upgrade path
- Ease of Use + Learning Curve
- Community + Documentation
- Existing parts
- Cost to others
- Every tool sucks

No matter what tool you use, make sure that the tool works for you.
Features:

- Make sure that the tools you need later are available at a cost you can afford
- Worth looking forward to avoid a large time investment on multiple design tools.

Existing parts helps when you’re getting started, but shouldn’t be considered a “key” factor. With experience, you’re likely to rely mostly on your own libraries, not community. (More later)

Cost to others:

- Open projects are less-open if no one else can open the files
- However, community benefits from having the project exist at all, even with non-usable source files
- Someone can port it from one tool to another
- In the end, do what you need to make projects happen.

Picking Your Design Tools

Clearly biased suggestions:

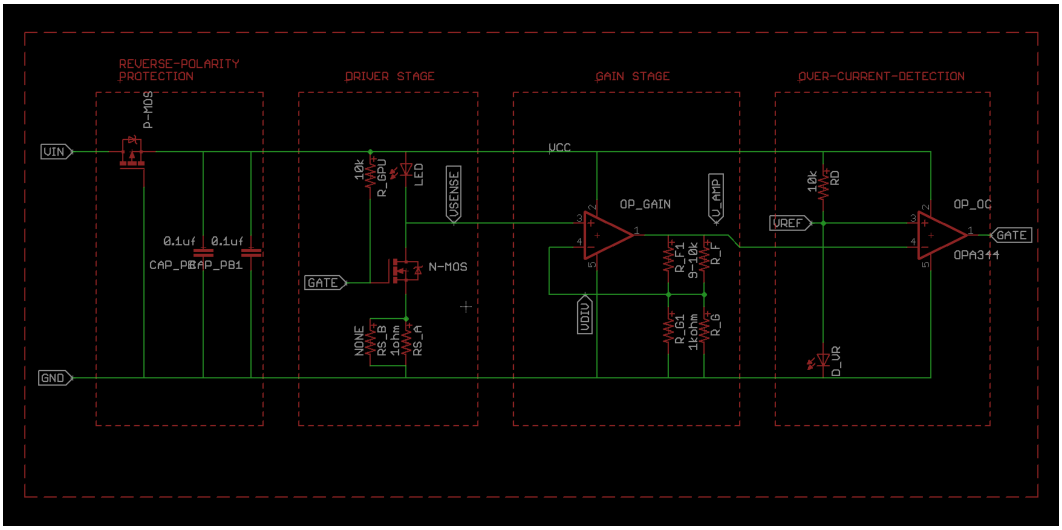
- KiCAD
    - Free, open source
    - Great community
    - No design restrictions
    - !! Quirks, bugs, version-specific issues
    - Rapidly improving

Picking Your Design Tool

- Eagle
  - Easy to get started with
  - Great community
  - Well developed community libraries
  - de-facto standard in the Arduino community
  - !! VERY stagnant development.
  - !! Lots of restrictions on free versions

If in doubt, try KiCAD first.

Schematic layout!



This schematic is from my Bigger Better Breadboard project : https://github.com/tekdemo/bigger-better-breadboard/tree/master/LEDs

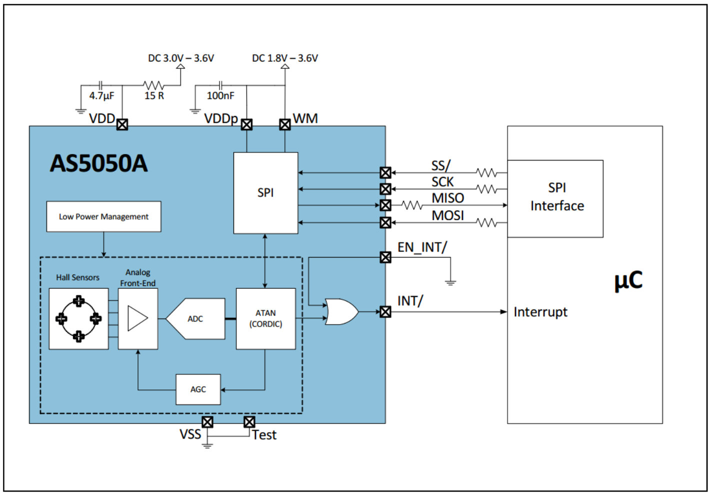
Glossary: Schematic

- Component
  - A part, such as a resistor or integrated circuit (IC)
- Net
  - A name for a particular signal
  - Can connect many wires, ICs, or board layers.
- Schematic
  - The symbolic connections between parts.
  - Connects Net, not "traces".

Schematic

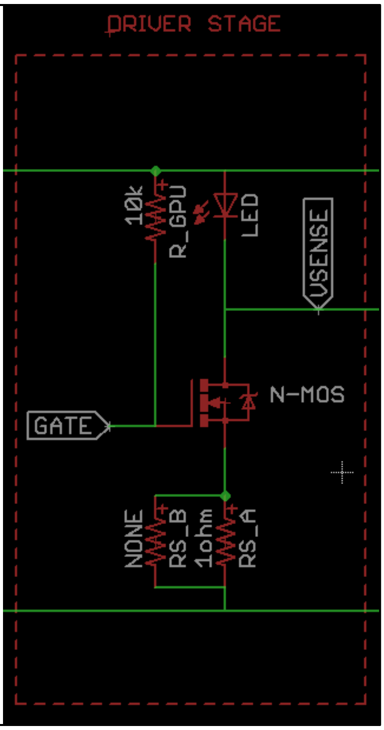
- Datasheets: A dummies guide
- Provide suggested components
- Details connection interfaces and options

Figure 10:
Typical Application Using SPI 4-Wire Mode and INT/ Output



Schematic: Name nets

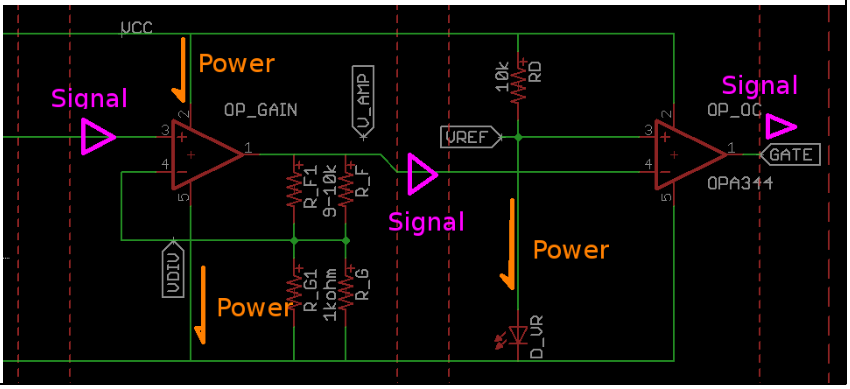
- Defaults are stupid: n$7
- Human-readable means human-checkable
- Can be checked during PCB Routing too
- Allows meaningful labels



Some design tools (eg, KiCAD) can also show net names ON the trace during routing. This can make it trivially easy to keep good track of your routing.

Schematic: Conventions

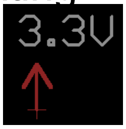
- Power flow goes top to bottom
- Signal flow goes left to right
- Most parts follow this (not all)

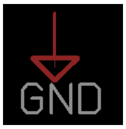


- Following conventions will reduce mental time, improve checking
- Helps prevent footprint errors later
- !! Always check community footprints. This schematic actually had op amps with Power on bottom, and GND on top. Wasn’t caught, and was a pain to correct.

Schematic: Conventions

- Active low nets: Prefix name with “n”
    - eg: nEnabled, nDisabled, nActive,nFloating
- Avoid long lines to power/ground
    - Use labels or Power/Ground symbols





- Use descriptive names to avoid confusion
    - PWM_RED_LED
    - I2C1_SDA
    - UART1_TX_OUTPUT

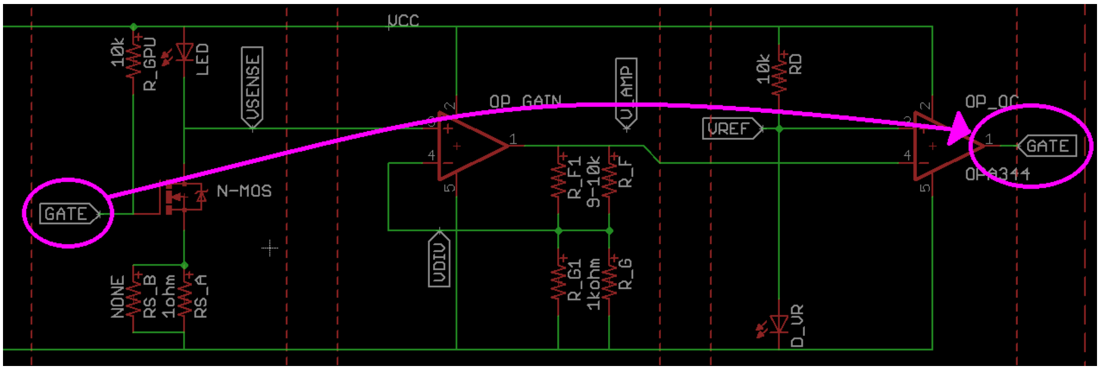
Active low nets usually come from pins with internal pull up resistors

- Very handy, since you can often leave them disconnected if you don’t need to use that function.

Net names are often helpful to match for programming reference. For mono-directional ones, mark them with output to make it clear

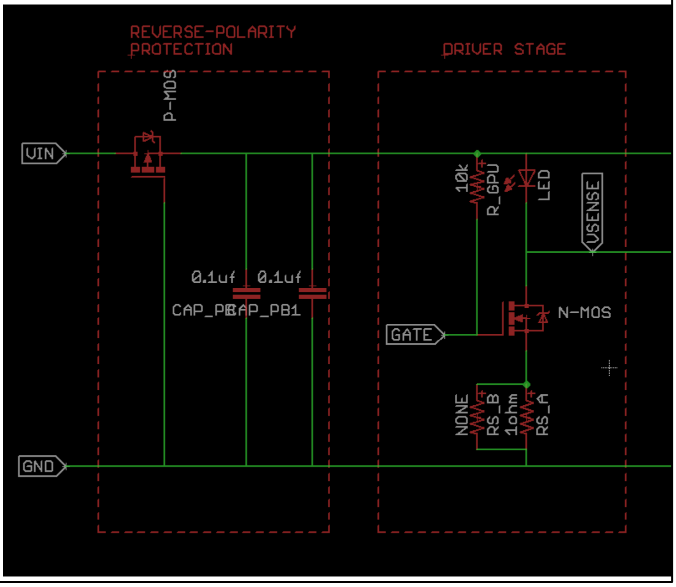
Schematic: Don’t worry about lines

- Use net names and labels
- Reduced clutter = better checking



Schematic:Create Functional Blocks

- Group related parts
- Label nets between blocks



Good boards start from a good schematic

Helps you generate logical places for test points later

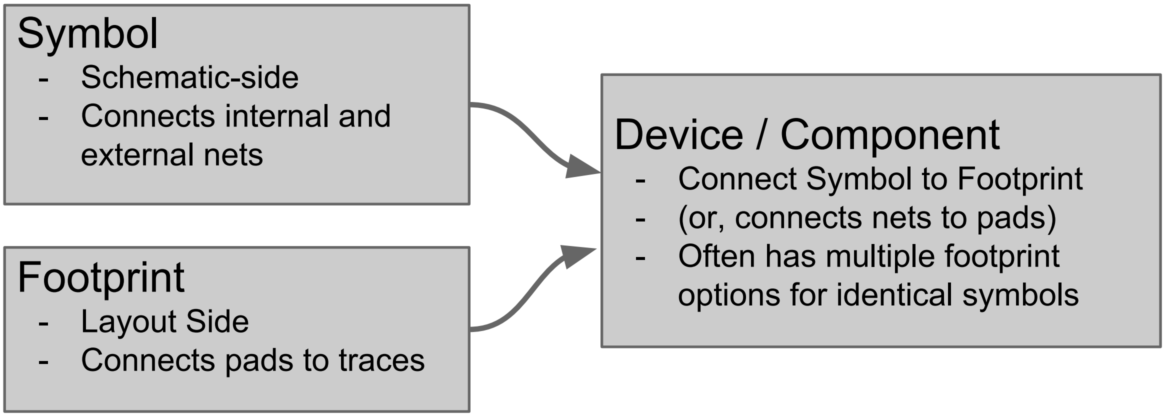
Helps indicate important signals

Schematic: Tips for Beginners

- Add Extra status LEDs
  - Add one per power net
  - One for microcontroller to blink/toggle
- Add test points
  - Give you a place to test voltages
  - Makes cutting/rerouting traces easier
  - Add in schematic so you don’t forget
- Make UART signals easy to fix
  - Add jumpers or test points to swap pins

Libraries

- Broken into 3 parts



Symbol

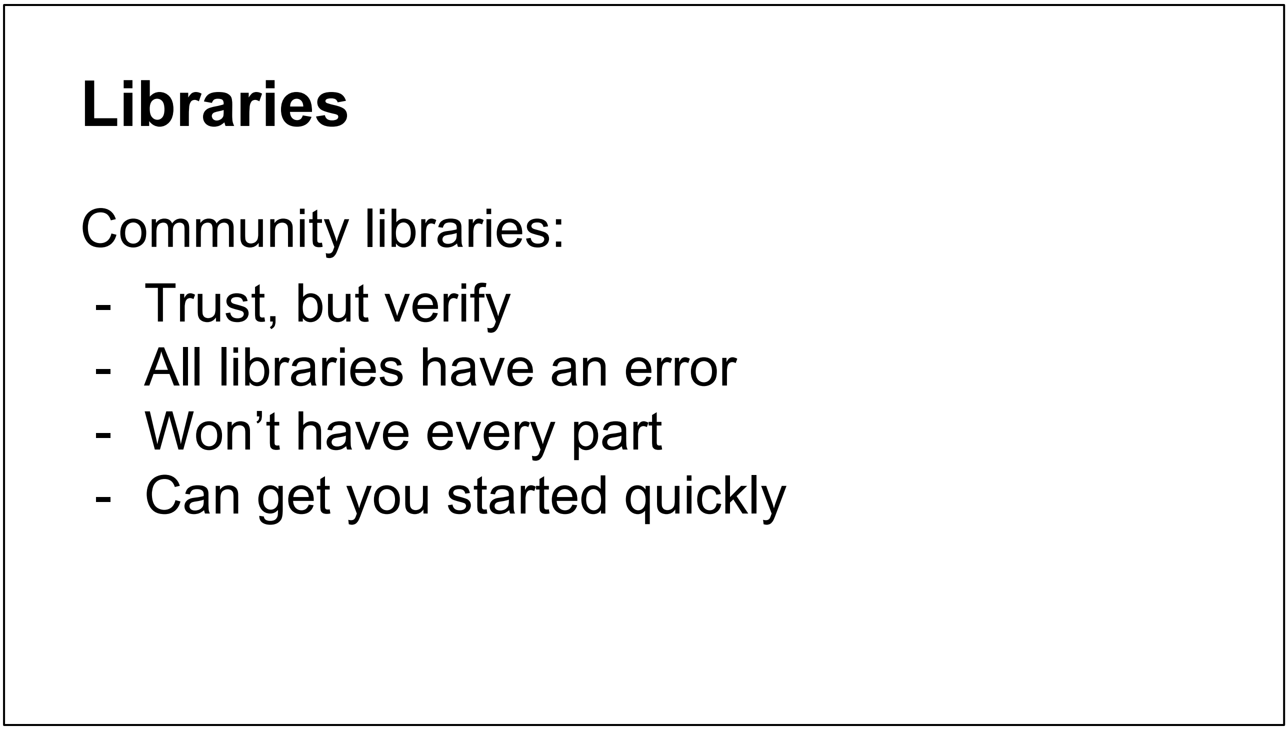
- Schematic-side
- Connects internal and external nets

Footprint

- Layout Side
- Connects pads to traces

Device / Component

- Connect Symbol to Footprint
- (or, connects nets to pads)
- Often has multiple footprint options for identical symbols



Libraries

Community libraries:

- Trust, but verify
- All libraries have an error
- Won’t have every part
- Can get you started quickly

Remember: You WILL find the error in a library. Eventually.

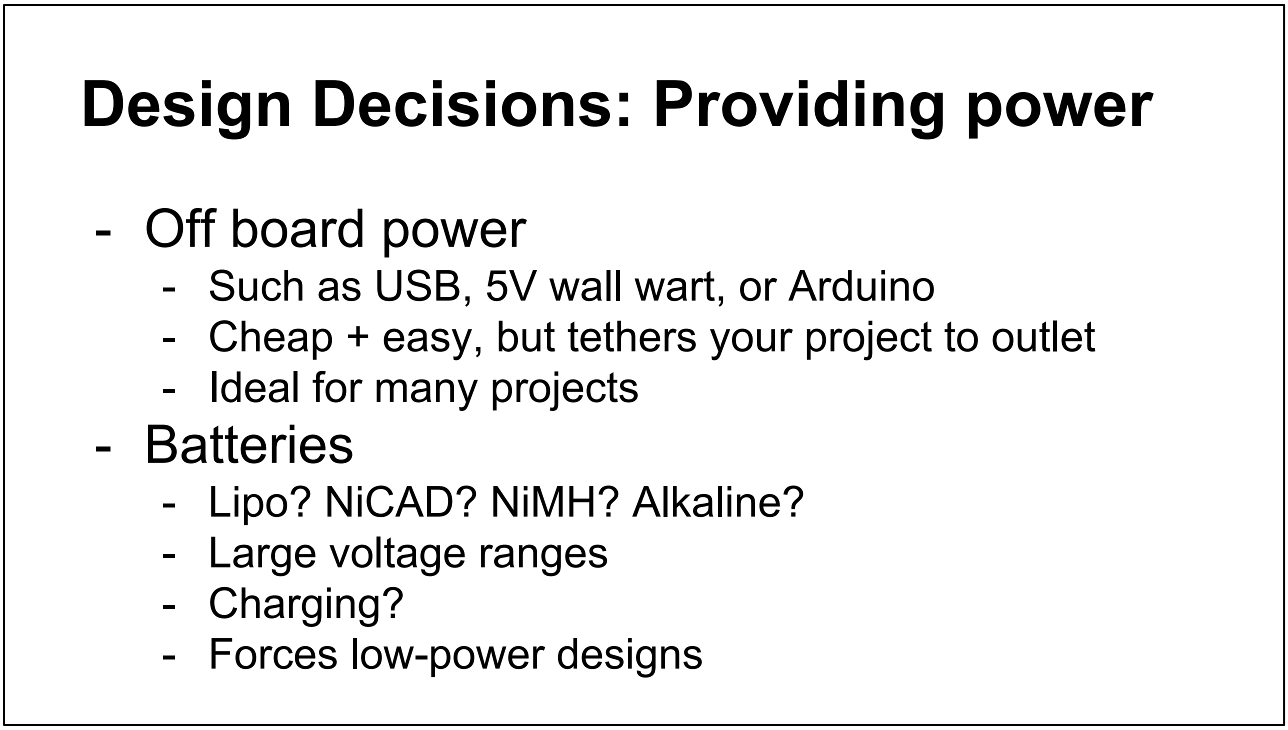
Libraries

Make your own!

- You’ll need to
- You’ll get the hang of it
- Can save you work
- Can (usually) be trusted

Suggestion:

1 library for new and untested footprints tested yet. Put any new footprints in this. Make 1 library for known, trusted footprints. Only copy stuff over once you’ve been happy with the results on a pcb



Design Decisions: Providing power

- Off board power
  - Such as USB, 5V wall wart, or Arduino
  - Cheap + easy, but tethers your project to outlet
  - Ideal for many projects
- Batteries
  - Lipo? NiCAD? NiMH? Alkaline?
  - Large voltage ranges
  - Charging?
  - Forces low-power designs

Be mindful of battery voltages, since it can often drive the component selection for other parts of your system, especially voltage regulation

Design Decisions: Regulating power

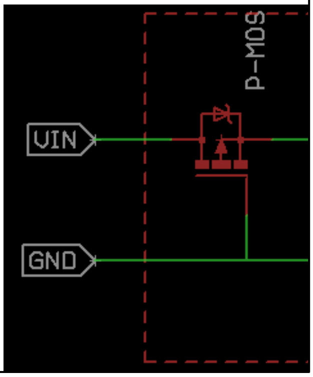
- Linear Regulator
  - Pro: Cheap and simple
  - Con: Watch the voltage drops
  - Con: Can be hot and inefficient
- Switching Regulator
  - Con: More parts, and much more complex
  - Con: Often needs more PCB area
  - Pro: Can provide large voltage drops efficiently
  - Pro: Can increase voltage source provides

Design Decisions: Regulating power

- Unregulated battery power
  - HERE BE DRAGONS. May cause unexpected issues. Sometimes serious.
  - Voltage range will drive component selection
  - Can be good for simple blinky or analog circuits

Design Decisions: Regulating power

- Input Power Protection
  - Good idea when moving past prototype stage
  - Can be very simple:
    - Reverse Polarity can be 1 part
    - Overcurrent or overvoltage can be 1 IC, or a few parts
    - Some ICs do everything for you

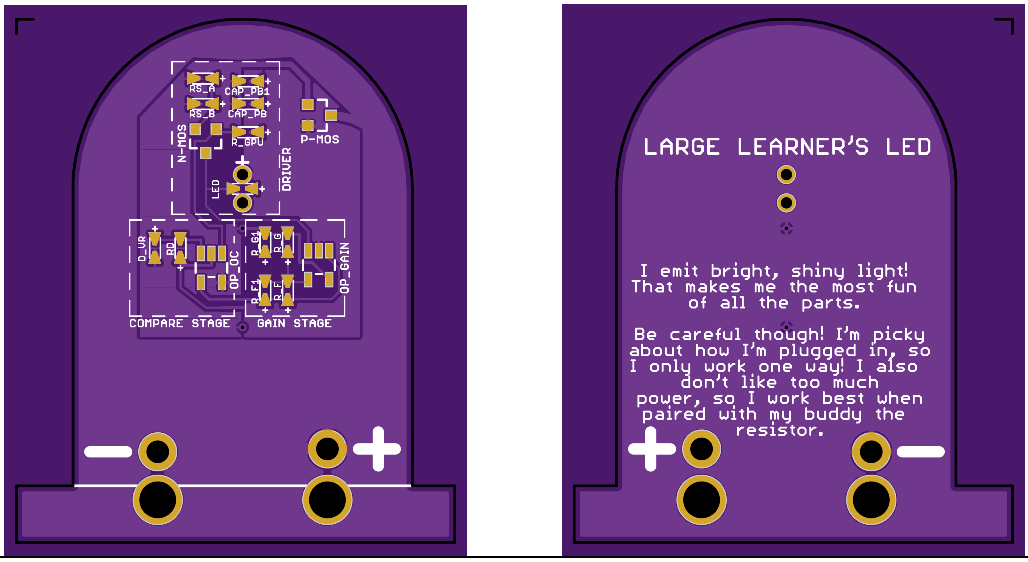


Read up on good protection things. Be mindful of

- Current going through protection circuit
- Voltage across protection circuit
- What your circuit realistically needs
- Failure modes of over-voltage over-current, reverse polarity. Can range from magic smoke release, to not working until it's plugged in properly.
-

If giving a board to others users, a keyed power connector can also help reduce errors, and alleviate need to do some power protection

PCB Routing: Physical Connections



I emit bright, shiny light! That makes me the most fun of all the parts.

Be careful though! I'm picky about how I'm plugged in, so I only work one way! I also don't like too much power, so I work best when paired with my buddy the resistor.

Again, LED Board for consistency
https://github.com/tekdemo/bigger-better-breadboard/tree/master/LEDs

PCB Routing: Glossary

AKA Board Layout

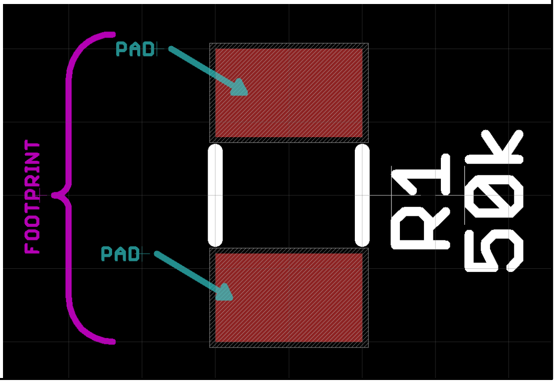
- DRC: Design Rule Check
  - Uses specifications to verify manufacturing requirements
- DFM: Design For Manufacturing
  - Guidelines and suggestions for preventing errors
  - Usually not enforced by design tool
  - Covers wide selection of design and layout areas
  - Critical as volumes and complexity increases

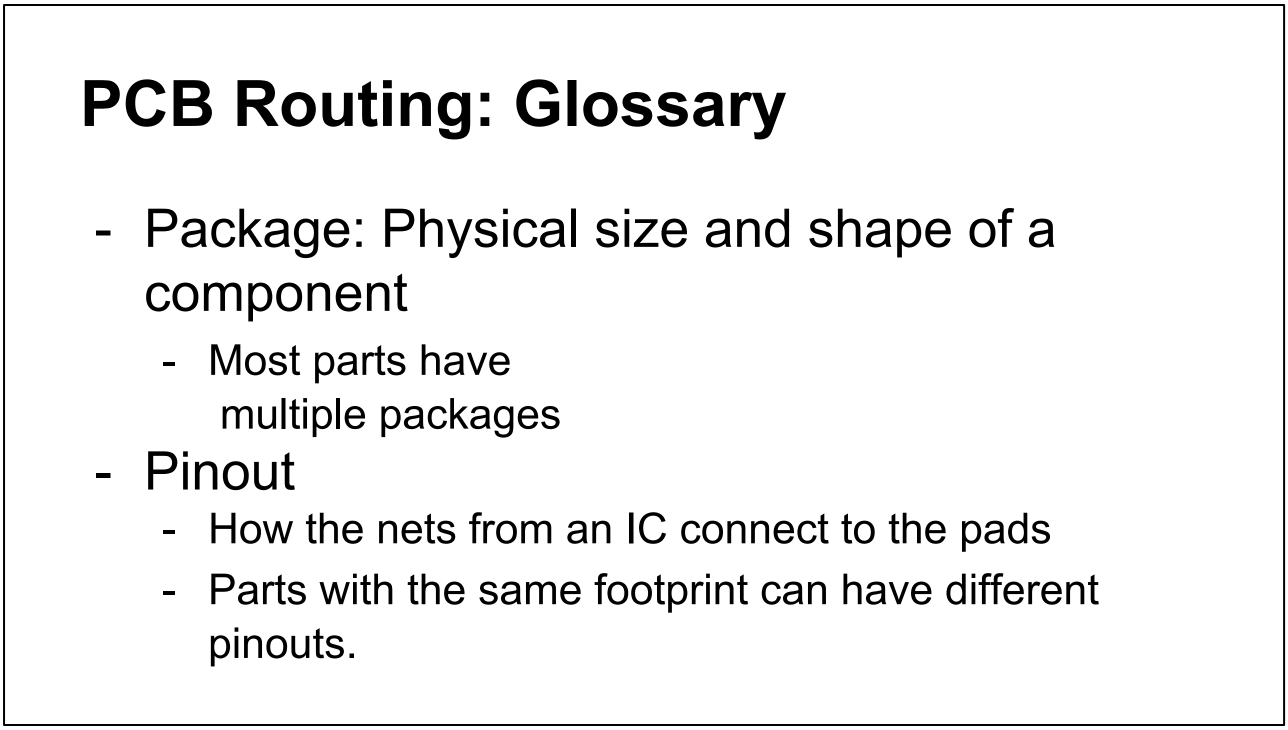
More on DFM Later.

- Can often be ignored for one-offs or short runs

PCB Routing: Glossary

- Stop Mask: Indicates a section of the board that should be exposed
- Pad: A small exposed section of copper.
    - Typically refers to where you'll be attaching a part
- Footprint: A set of pads that match a component's pin arrangement





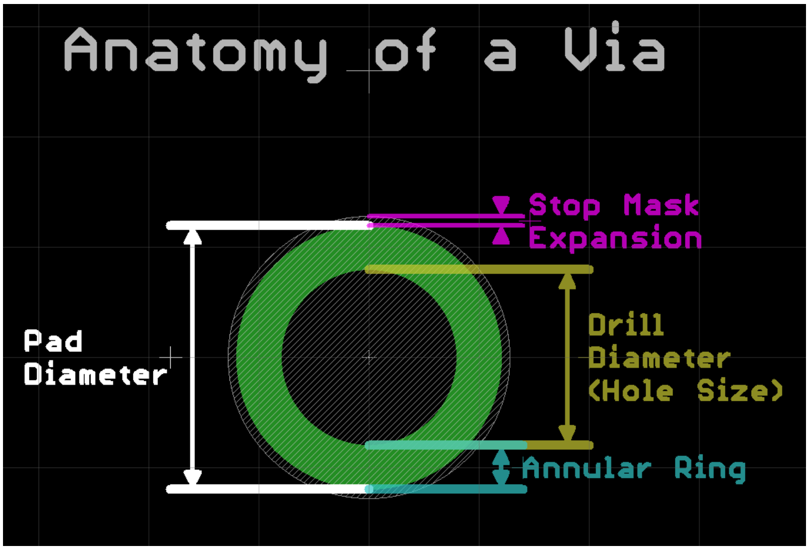
PCB Routing: Glossary

- Package: Physical size and shape of a component
  - Most parts have multiple packages
- Pinout
  - How the nets from an IC connect to the pads
  - Parts with the same footprint can have different pinouts.

Pinouts:

- Very common to have pinout variances on voltage regulators, transistors, and other points.
- Can be very confusing if you’re not careful

PCB Routing: Glossary

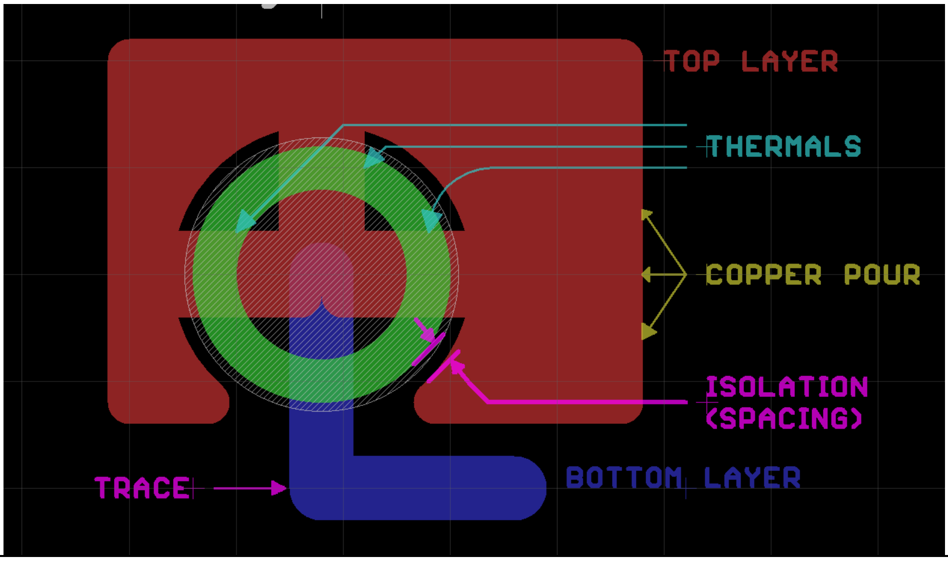


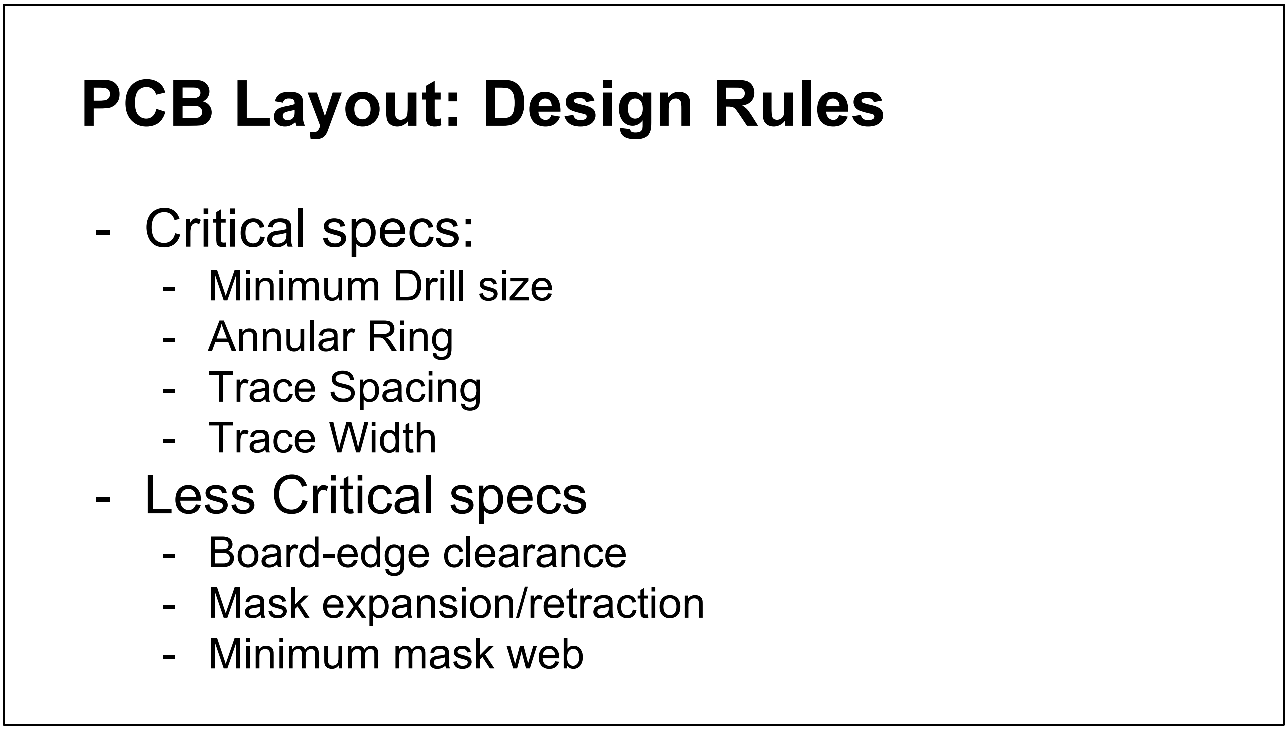
Typically, design tools expect annular ring and Mask Expansion as a DRC spec. Then, with a given hole size, they calculate the pad diameter.

Conversely, for specific holes you can often specify pad diameter and drill size, and the design tool will only complain if the annular ring is too small.

Typically, the drill diameter you specify is the “finished” hole size, after all fabrication processes. After that, the only variation will be the fabrication tolerance (Often around 2-5 mil).

PCB Routing: Glossary





PCB Layout: Design Rules

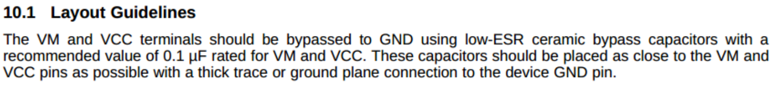
- Critical specs:
  - Minimum Drill size
  - Annular Ring
  - Trace Spacing
  - Trace Width
- Less Critical specs
  - Board-edge clearance
  - Mask expansion/retraction
  - Minimum mask web

The annular ring serves to simplify fabrication tolerances. It covers most errors that can go wrong, including layer misalignment, drill tolerances, and a few others.

PCB Layout:

Datasheets: A dummies guide

- If routing matters, ICs often have examples and notes.
  - High Current
  - RF
  - Low-power
  - Analog



10.1 Layout Guidelines

The VM and VCC terminals should be bypassed to GND using low-ESR ceramic bypass capacitors with a recommended value of 0.1 µF rated for VM and VCC. These capacitors should be placed as close to the VM and VCC pins as possible with a thick trace or ground plane connection to the device GND pin.

10.2 Layout Example

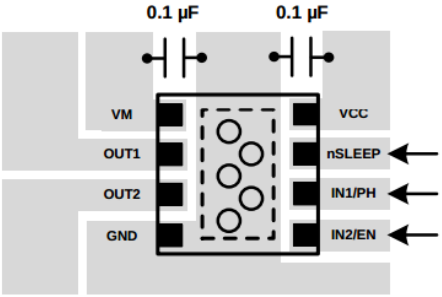
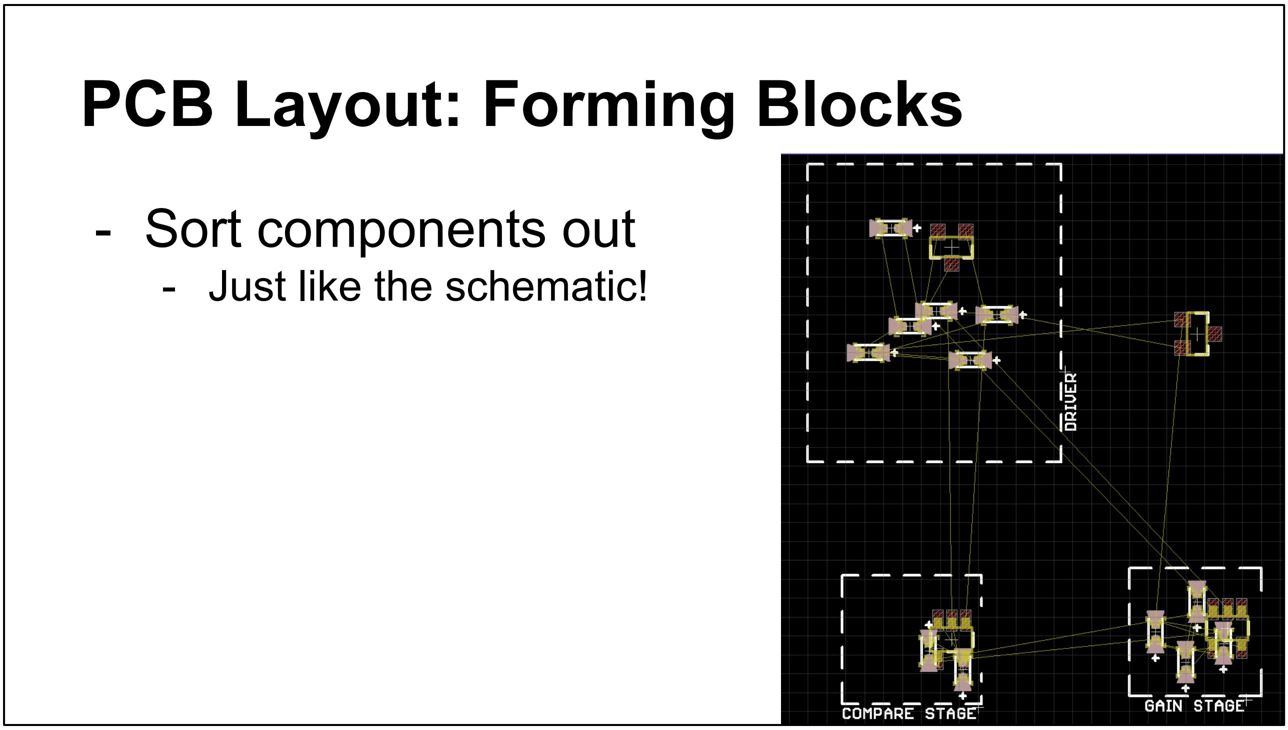


Figure 11. Simplified Layout Example

More on special signal types later.

PCB Layout: The general process

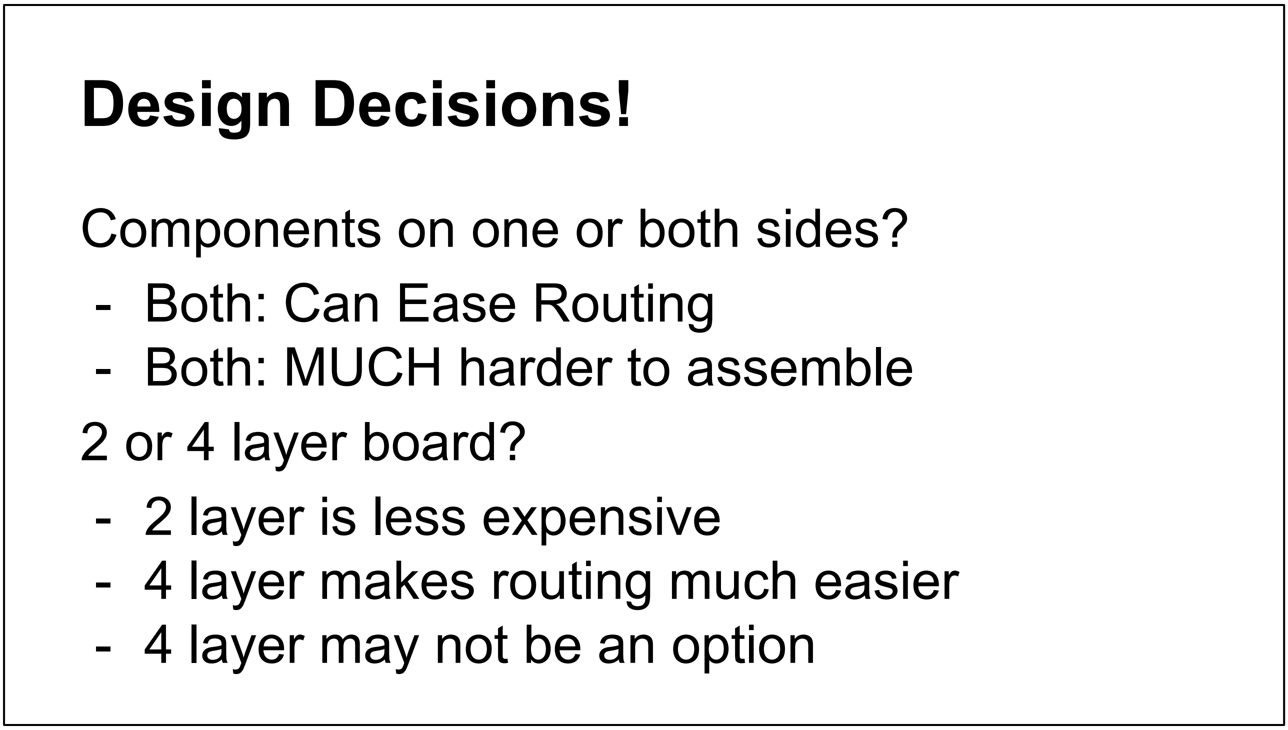
- Separate parts into blocks you made earlier
  - Organize parts
  - Route power and ground
  - Route Signals
- Combine blocks on desired board shape
  - Route power and ground
  - Route Signals
- Make corrections



PCB Layout: Forming Blocks

- Sort components out
  - Just like the schematic!

- Usually Reduces Layout time
- Improves error checking
- Doesn’t always work
  - Mechanical constraints
  - Some designs are not easily grouped



Design Decisions!

Components on one or both sides?

- Both: Can Ease Routing
- Both: MUCH harder to assemble

2 or 4 layer board?

- 2 layer is less expensive
- 4 layer makes routing much easier
- 4 layer may not be an option

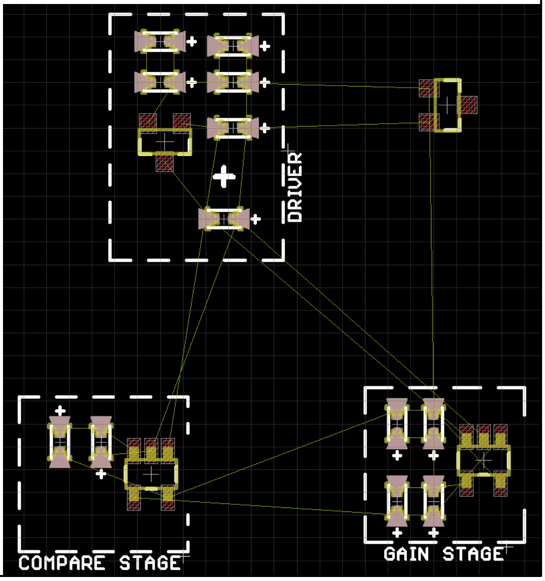
In most cases, trying to get all SMD components on one side is a good idea.

- Difficult or impossible to reflow parts when they’re on both sides
- Often limits you to hand-assembling one side of the board
- In mass production, 2 sided assembly services is much more expensive.

PCB Layout: Routing Blocks

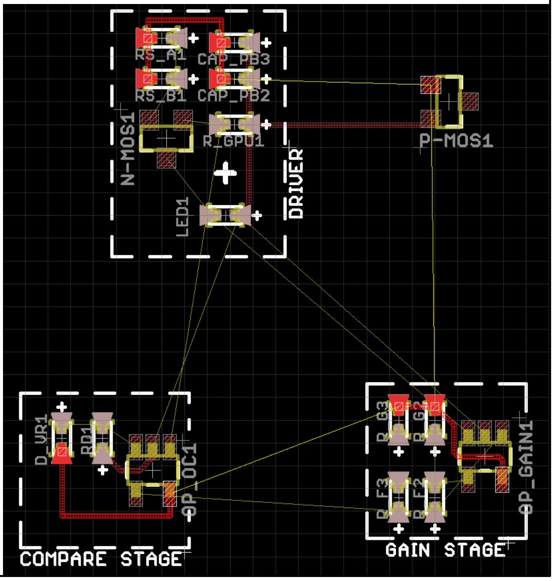
Untangle Airwires

- Minimize crossover
- Minimize routing length



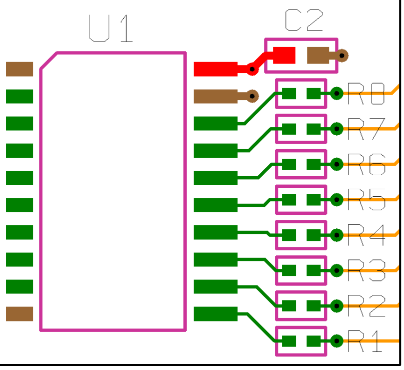
PCB Layout: Routing Blocks

- Power and ground first
- Bypass caps second
- Important Signals
  - Analog
  - communication
  - high-power
- Everything else



PCB Layout: Bypass caps

- Magic sprinkles of electronics
- Every IC should have one
- Requires intentional routing
  - Minimize “loop area” between IC Power, IC Ground, and cap pads
  - Avoid vias (generally)
  - Should be physically close to IC

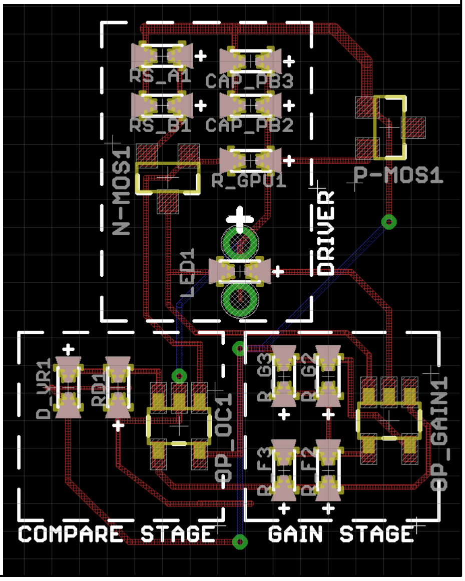


http://blog.optimumdesign.com/how-to-place-a-pcb-bypass-capacitor-6-tips

Worth doing some research. Especially if running specialized ICs, high frequency, or mixed-frequency circuit boards.

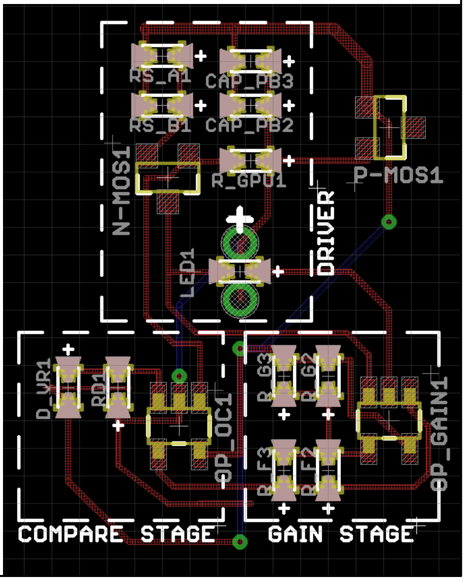
PCB Layout: Connecting Blocks

- Arrange blocks
  - Minimize airwire distance
- Plan for mechanical
  - Odd PCB shape?
  - Offboard connections?
  - Mounting?
- Connect!
  - Power, then signals



PCB Layout: Routing Strategies

- Short + Long
  - Put Parts + short traces on top
  - Long connections on bottom
  - Ground pour on bottom
- Up/Down + Left/Right
  - Put up/down traces on top
  - left/right traces on bottom
- Varies by design!



These two help a lot when getting started.
Many ICs favor particular strategies, or force their own

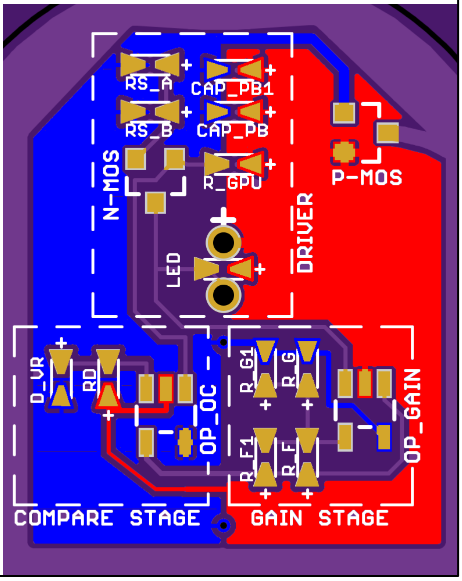
Some types of signals force specific strategies and priorities (discuss later)

Odds are, at this point you're going to re-do routing. That's fine!

- Usually you'll discover a much more effective way to route the boards.
- Will often be dramatically simpler than the first time.

PCB Layout: Design Decisions

- Ground/Power Planes?
  - Route first! (for now)
- 4 layer Routing Strategy:
  - Top: Parts + short connections
  - Ground plane
  - Positive power plane
  - Bottom: Longer connections



Route first! Prevents common mistakes with ground planes

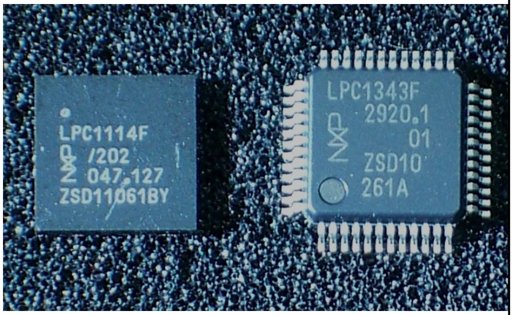
- Poor ground connections
- disconnected sections
- long, zig-zag traces

4 layer

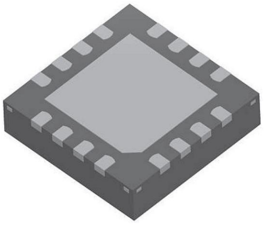
- Super easy to route
- Difficult (but not impossible), to make terrible ground planes

PCB Layout: Design Decisions

Surface Mount Parts



  - Lots of advantages
  - Need to use them eventually
- Beginner Friendly
  - Passives: 1206, 0805
  - IC: Leadless packages, MSOP-*, SOT-*
- Less friendly
  - 0402: Really tiny and hard to place
  - BGA, pitches 0.5mm and lower

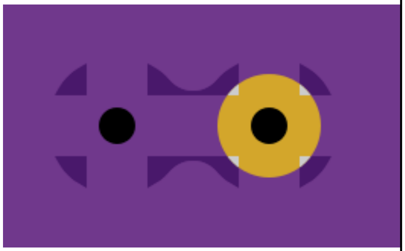


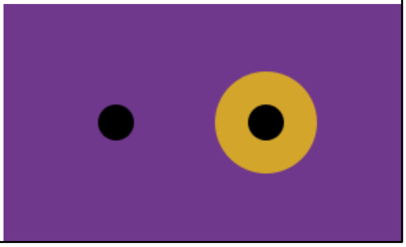
Fine-point pins are really tricky. They hold solder, and are hard to de-bridge.

Leadless packages are surprisingly easy to rework if you screw up assembly. Just remove excess blobs.

PCB Layout: Design Decisions

- Via Tenting?
- Thermals?
- Assembly consideration
  - Space for rework
  - Leave room for silk
- Heatsinks?
  - Voltage Regulators
  - Motor drivers



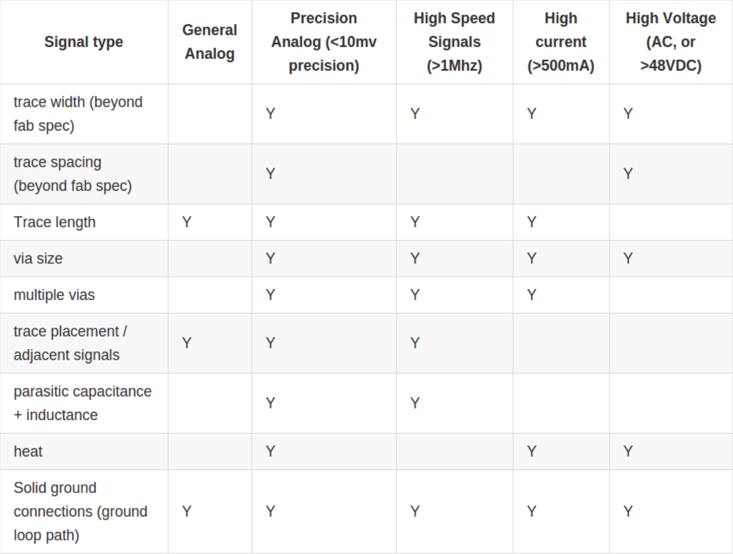


- Tenting:
  - pro: prevents assembly mistakes
  - pro: looks nicer
- Not tenting:
  - con: Prevents access for debugging
  - Improves access and debugging
- Thermals:
  - improve reflow on pads
  - Not critical on vias
  - cons: Can reduce current flow when handling high-current traces

PCB Layout: Routing Busses

- Bus: A group of signals performing the same function
  - Common for chip-to-chip communication
- Should be routed at the same time
  - Try to keep together
  - Length differences may matter
- Consult routing guides for the protocol
  - More/less picky, voltages, pull ups, oh my.

Routing: Notable Signal types



Signal type	General Analog	Precision Analog (<10mv precision)	High Speed Signals (>1Mhz)	High current (>500mA)	High Voltage (AC, or >48VDC)
trace width (beyond fab spec)		Y	Y	Y	Y
trace spacing (beyond fab spec)		Y			Y
Trace length	Y	Y	Y	Y	
via size		Y	Y	Y	Y
multiple vias		Y	Y	Y	
trace placement / adjacent signals	Y	Y	Y		
parasitic capacitance + inductance		Y	Y		
heat		Y		Y	Y
Solid ground connections (ground loop path)	Y	Y	Y	Y	Y

General analog:

- Typical hobby-level stuff, with voltage swings of 100+mV causing no issues.
- Greater than/less than voltage comparisions usually fall in here

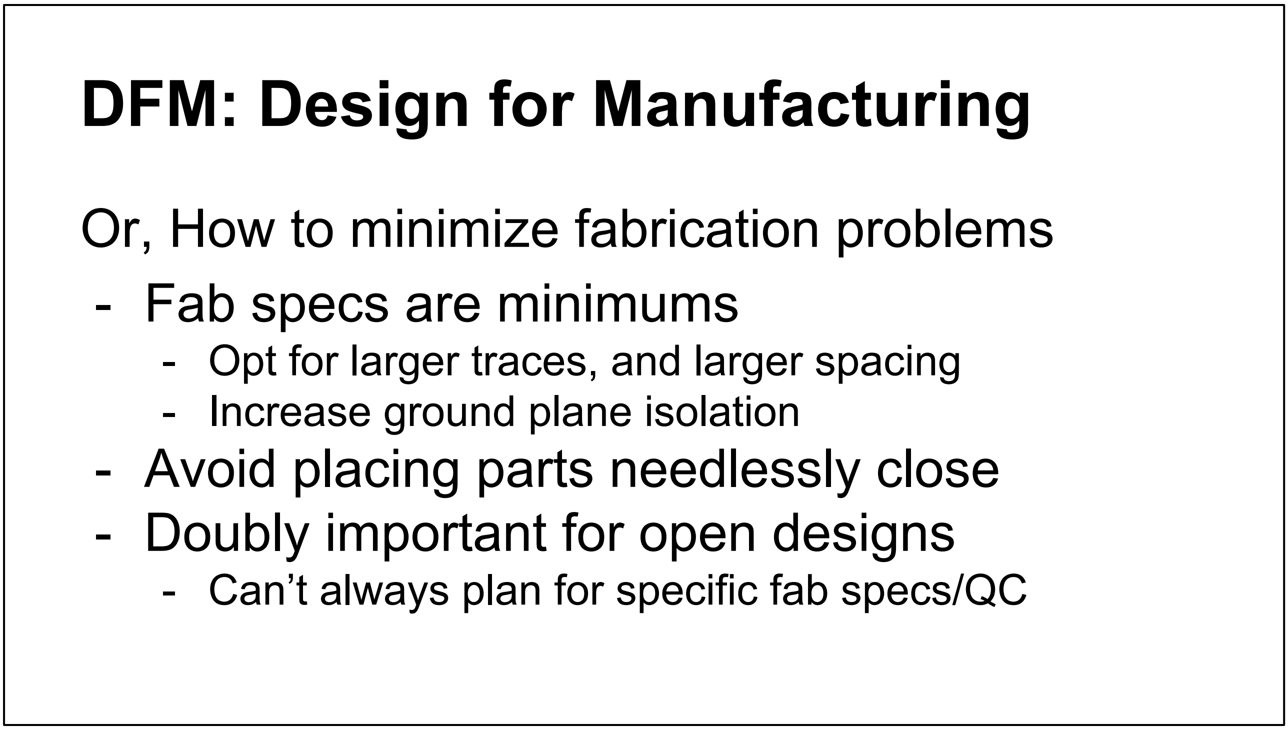
Precision Analog:

- Usually for fine logging or measurements.

High Speed:

- Common for clock signals, Wireless radios, some serial busses, video signals

Datasheets and whitepapers will help when configuring routing for all of these signals



DFM: Design for Manufacturing

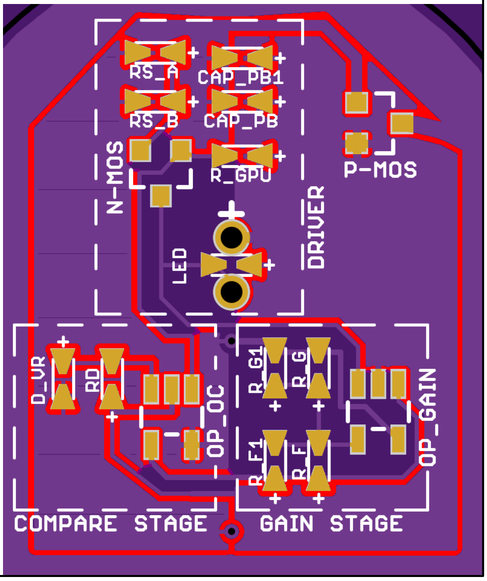
Or, How to minimize fabrication problems

- Fab specs are minimums
  - Opt for larger traces, and larger spacing
  - Increase ground plane isolation
- Avoid placing parts needlessly close
- Doubly important for open designs
  - Can’t always plan for specific fab specs/QC

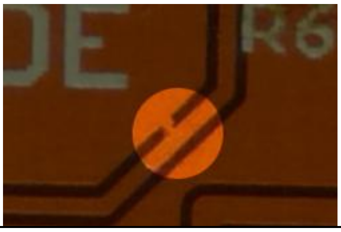
For open designs, any fabrication errors might be seen as design flaws on your end.
Don’t frusturate the community!

DFM: Ground Plane Isolation

- Set above fab min spec
- Touches lots of traces
- Hard to find and troubleshoot



- Example Short:



When routing, you'll naturally avoid placing traces really close together, and making good use of space. However, design tools by default, put ground planes at the DRC spec, which is the *minimum* fab spec. Don't let that ruin your meticulous DFM efforts! Use the minimums when needed, but avoid for your ground pours.

The pic above (red) shows the ground plane contact with other signals. That's a lot of places for things to go wrong. As designs increase in complexity, this becomes increasingly more difficult to troubleshoot.

To adjust this, check the "Isolation" settings in the DRC or Polygon/copper pour setting. A good start is fab spacing spec +25% or so.

Picture from http://www.sebastians-site.de/ and http://dangerousprototypes.com/docs/Get_your_PCBs_made

Fabrication: Getting prepped

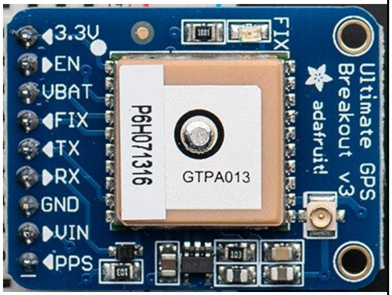
- Run DRC
- Double check pinouts
- Verify availability of components
- Put date, version, and name on board
- Mechanical Concerns
    - Mounting holes?
    - Enclosures?
    - Part sizes?

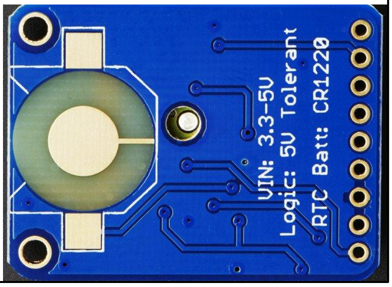
Often overlooked on mechanical concerns is interference with tall parts. Very important if your board has an enclosure.

Some design tools (including KiCAD!) can generate 3D models of your parts, and you can import them into 3D cad models

Fabrication: Getting prepped

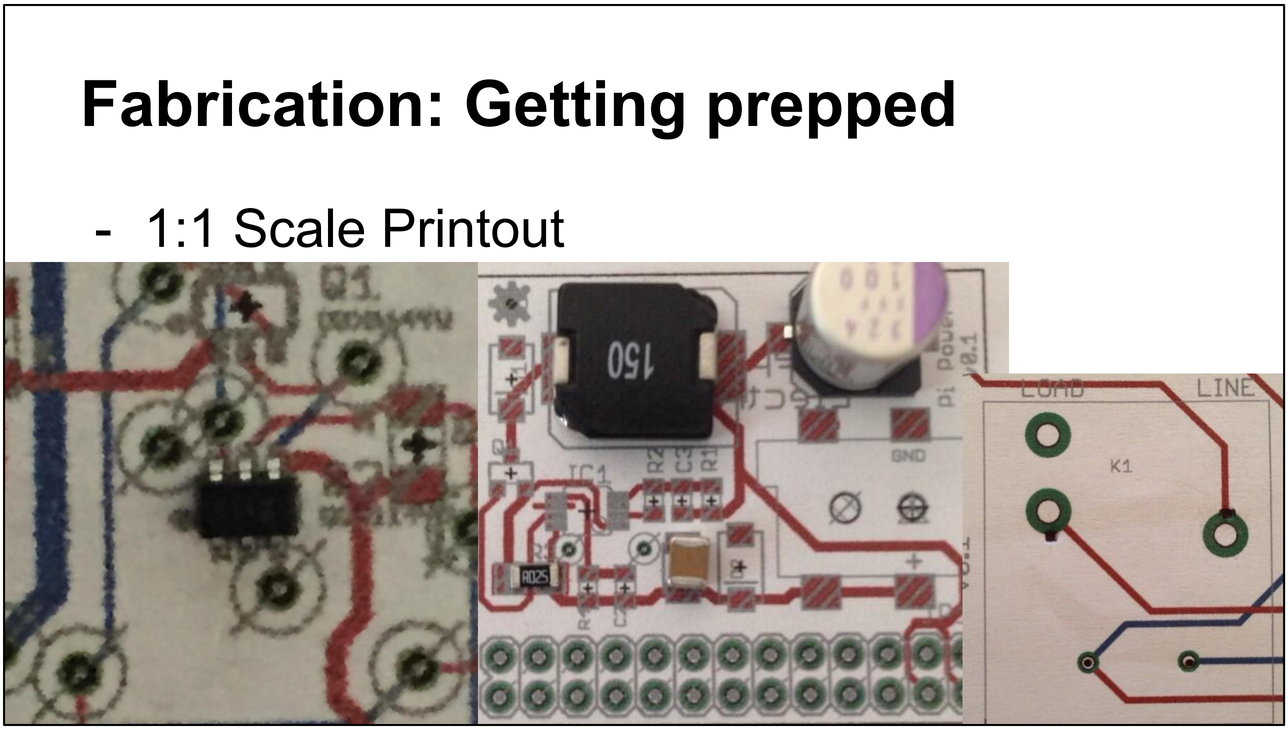
- Clean up silkscreen
  - Component names
  - Label off-board connections
  - List power input voltage ranges
  - Mark polarity of components
  - Make silk readable size (>0.035”)





https://www.adafruit.com/products/746

Extra credit for Adafruit adding IO directions for their inputs and outputs.



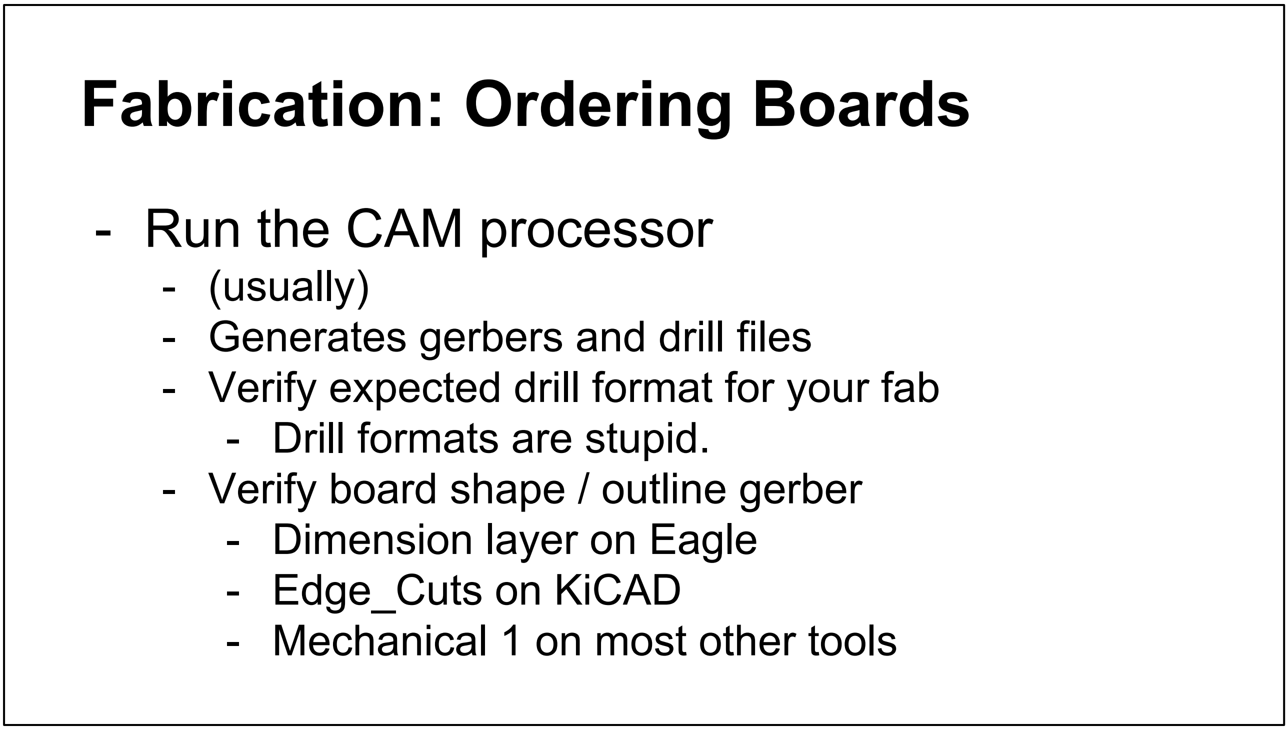
Fabrication: Getting prepped

- 1:1 Scale Printout

Easy way to double-check a lot of potential board issues.

On the right, is a relay where the footprint didn’t match the actual part. Money saved with a piece of paper.

Pictures from https://twitter.com/nwsayer



Fabrication: Ordering Boards

- Run the CAM processor
  - (usually)
  - Generates gerbers and drill files
  - Verify expected drill format for your fab
    - Drill formats are stupid.
  - Verify board shape / outline gerber
    - Dimension layer on Eagle
    - Edge_Cuts on KiCAD
    - Mechanical 1 on most other tools

Some fabs can accept certain design files directly

- Oshpark accepts Eagle,
- Some fabs accept ORCAD or Altium files

If possible, preview drill files with the tool your fab uses.

- Not always possible, since most fabs have custom tools

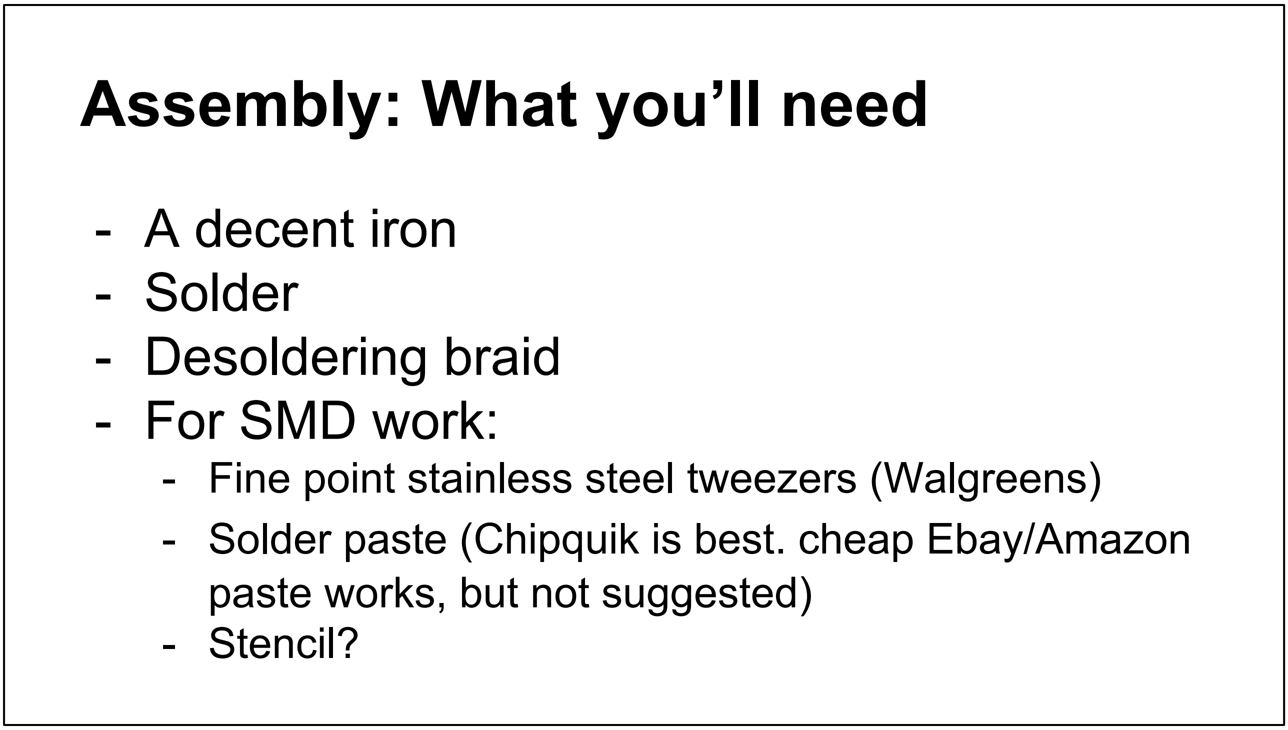
Fabrication: PCB Layers / Gerbers

- Simple files corresponding to a single part of the fabrication process
  - Copper Placement
  - Hole placement
  - Solder resist placement
  - Silkscreen printing
-

Fabrication: PCB Layers / Gerbers

- Complexities come from format ambiguity
  - Plated drills?
  - How is the edge defined?
  - How is the drill format handled?
  - Positive or negative internal layers?
- If in doubt, ask your fab!
  - Folks get paid to help you. Take advantage!

- Some fabs require multiple drill files, others require singular drill files
- Even with a known format, the fab can interpret things in certain ways
  - How the edge is defined on the edge-layer
  - Which layer defines the edge



Assembly: What you’ll need

- A decent iron
- Solder
- Desoldering braid
- For SMD work:
  - Fine point stainless steel tweezers (Walgreens)
  - Solder paste (Chipquik is best. cheap Ebay/Amazon paste works, but not suggested)
  - Stencil?

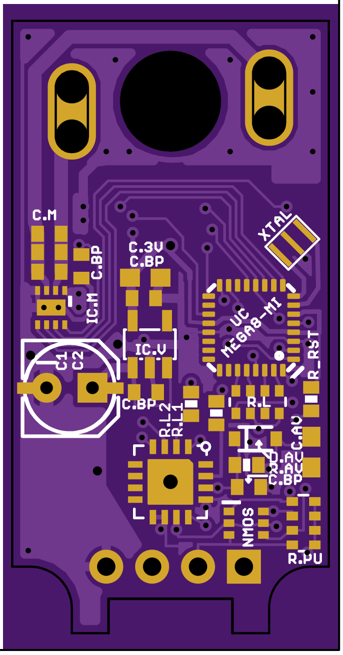
For irons, avoid the Firestarter ones for 10-15 bucks.

- Minimum is a ~$35 Weller iron from ebay. Get a ⅛ chisel tip
- Suggested irons are Hakko or Metcal. If you’re sticking with the hobby, it’s worth paying for

Kapton stencils are cheap, make for easy work, and very reliable. Highly suggested. Also ship faster than PCBs, so it’s unlikely to cause delays.

Assembly: SMD Placement

- Expect to bump them a bit
  - Usually pretty forgiving
- Start with hard-to-reach parts
  - Stuff in center, tight fit, etc
- Go to “easy” parts
- End with large or finicky footprints
  - BGA, TQFP, etc



Assembly: Planning BOM

- Giant printout
  - Include values, names, notes
- Parts list

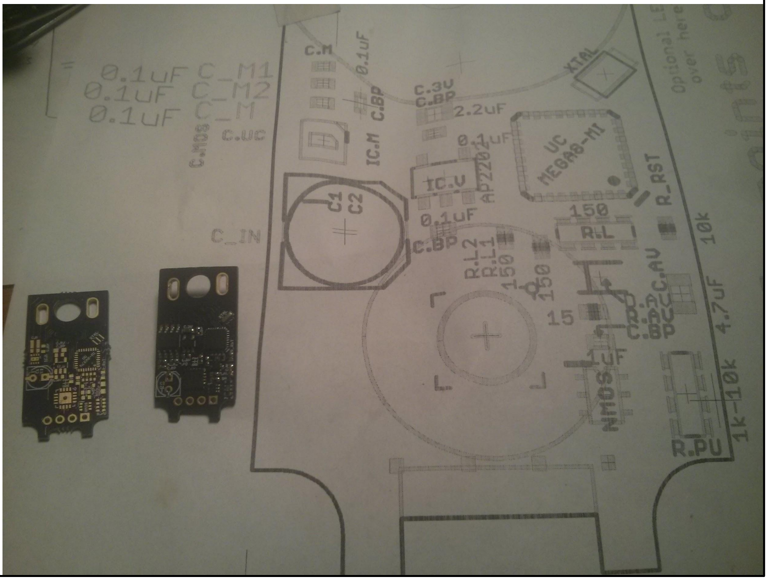


Figure out your assembly order

- Pull parts out of the box, and place on any/all components on the large printout.
- This gives a good check for values and names, quantities and placement on the PCB
- Transfer from paper to any boards

Alternatively, you can get a large printout that has line-by-line printouts

- Tricky to read
- lots of double-checking
- Gives me a headache.

Assembly: Reflow Procedure

- Minimum required items:
  - Old pan
  - IR Thermometer
- My home procedure
  - Medium til around 250F-300F (flux melts into matte pools)
  - High until solder flows (~250F)
  - Nudge parts if needed (careful!)
  - Remove pan from burner

Not perfect, but it *works.* Try one at a time until you get a feel for it

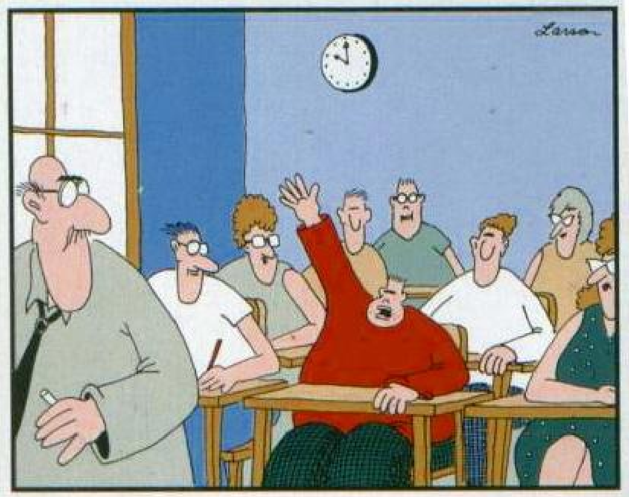
Produces poor temperature profiles, so not ideal for BGA parts, or sensitive components with precise assembly needs.

Assembly: Through-Hole

- Lots of good video tutorials
- Mostly practice
- Lots of flux
- Pro tips:
  - Poster Tack is your friend.
    - Holds board firmly to table
    - Blob around loose parts
  - Sparkfun Locking headers = <3

    https://www.sparkfun.com/tutorials/114

Questions!



"Mr. Osborne, may I be excused?
My brain is full."