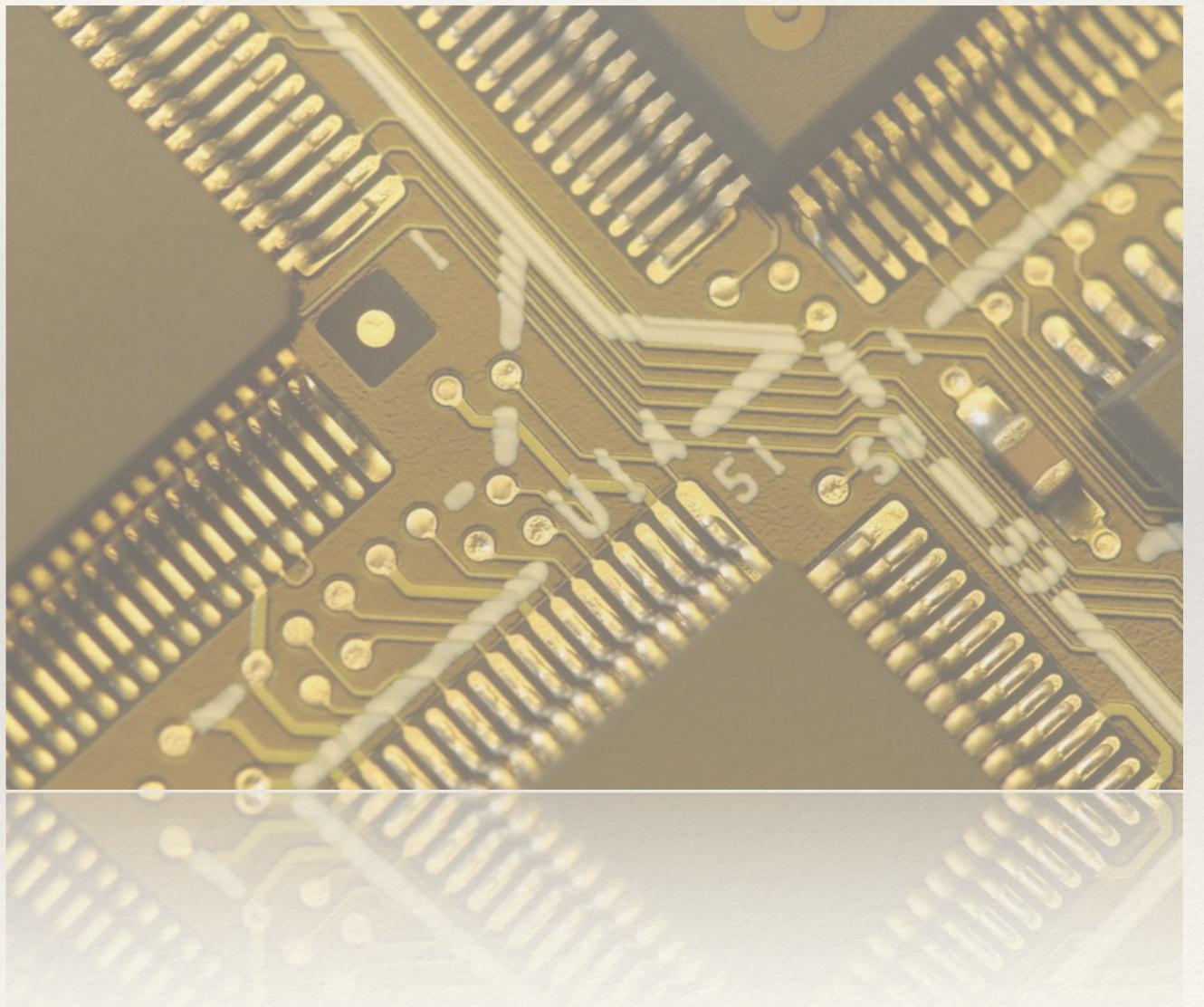

Intro To Eagle

*Design your own
Arduino shield*

EDA (Electronic Design Automation) Software

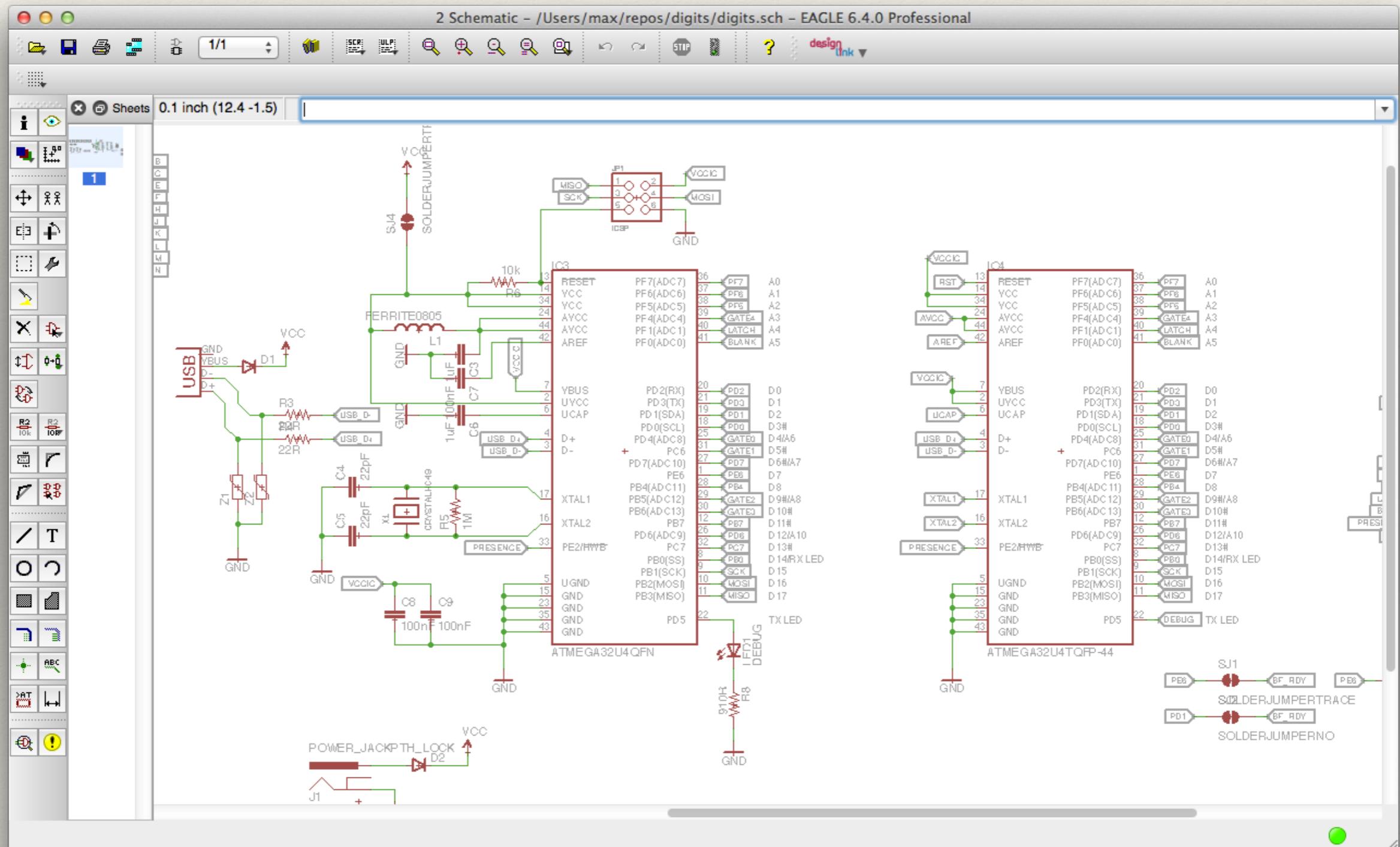
- ❖ Cadence Orcad (orcad.com)
 - ❖ Pro / \$\$\$\$
- ❖ Altium (altium.com)
 - ❖ Pro / \$\$\$
- ❖ KiCad (kicad-pcb.org)
 - ❖ OSS / Free
- ❖ Cadsoft Eagle (cadsoftusa.com)
 - ❖ Hobbyist / Free - \$\$



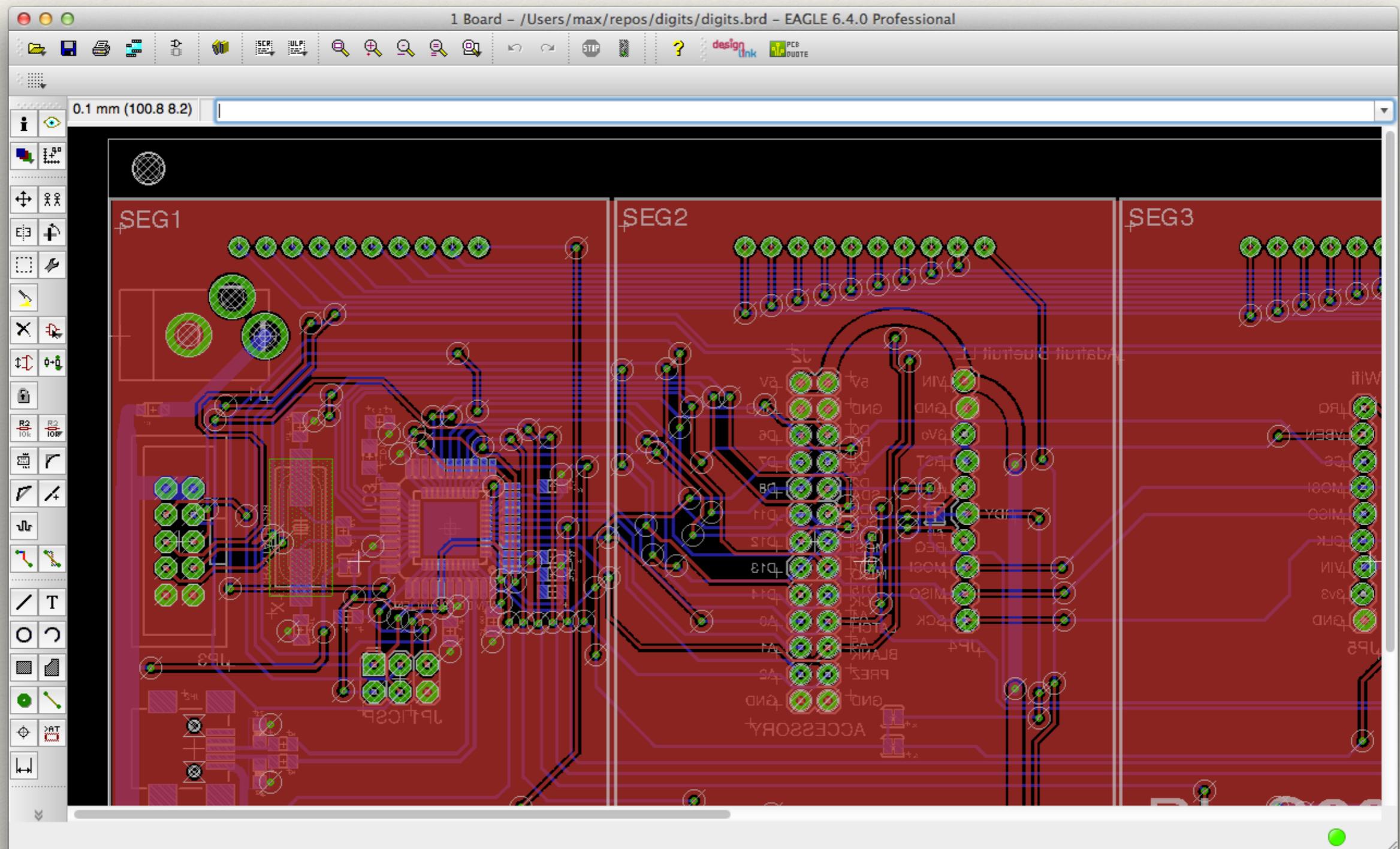
Cadsoft Eagle

- ❖ Three modules:
 - ❖ Schematic Capture
 - ❖ Board Layout
 - ❖ Library Editor
- ❖ Versions:
 - ❖ Light (1 schematic sheet, 2 signal layers, 100x80mm routing area)
 - ❖ Standard (99 schematic sheets, 6 signal layers, 160x100mm routing area)
 - ❖ Professional (999 schematic sheets, 16 signal layers, 4x4m routing area)
 - ❖ Hobbyist (99 schematic sheets, 6 signal layers, 160x100mm routing area)
 - ❖ For individual, non-commercial use only (not available for China)

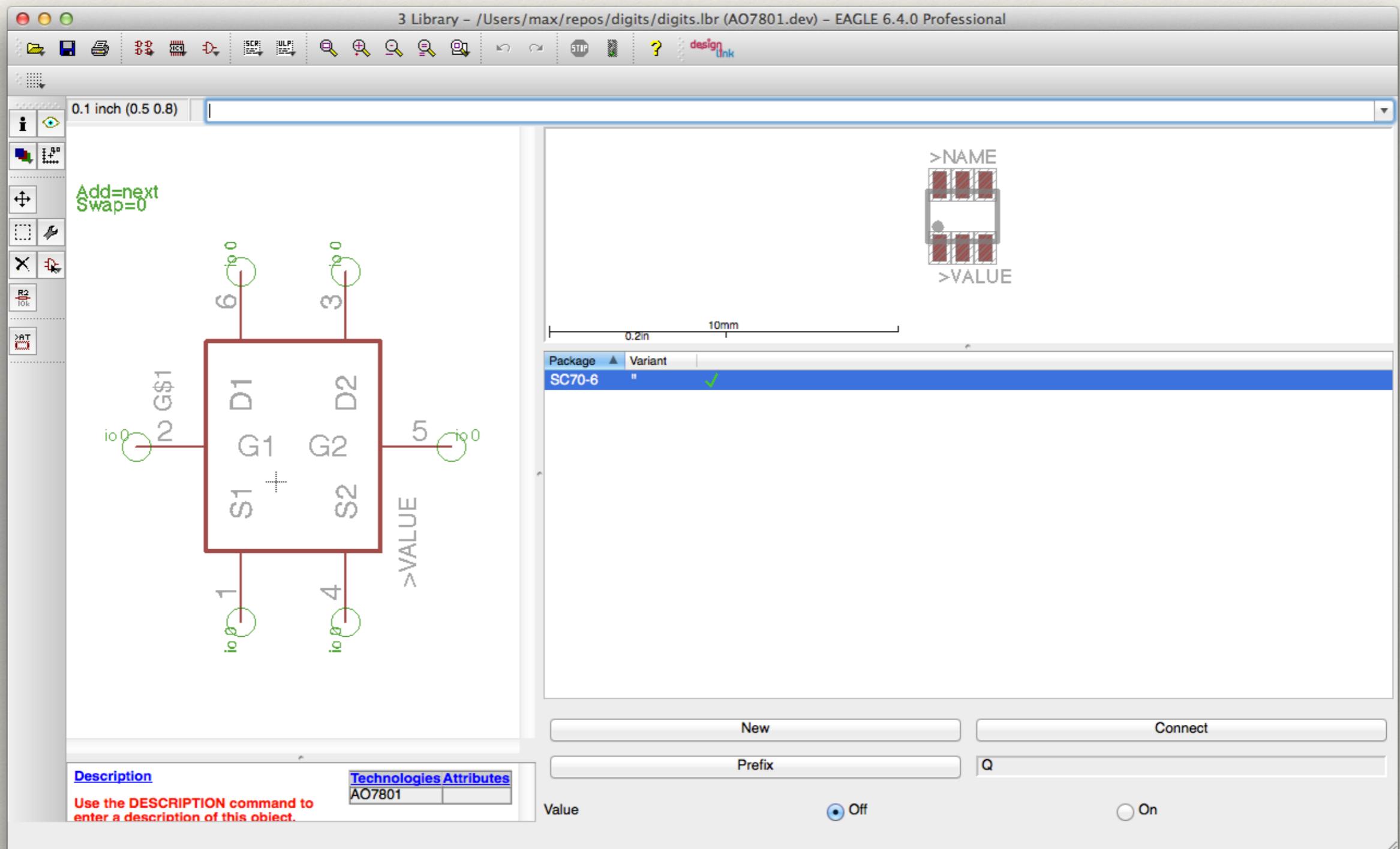
Schematic Capture



Board Layout

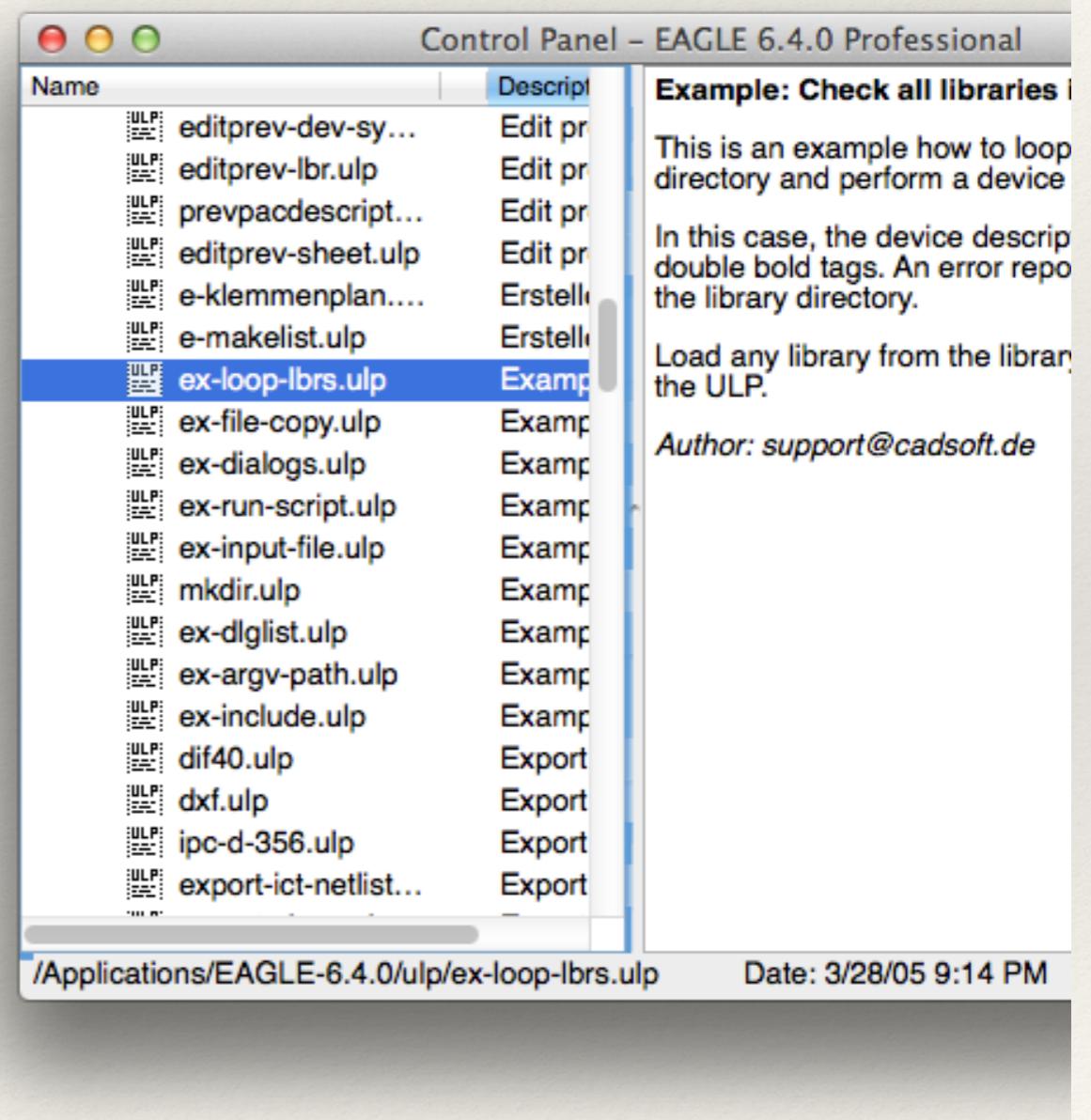


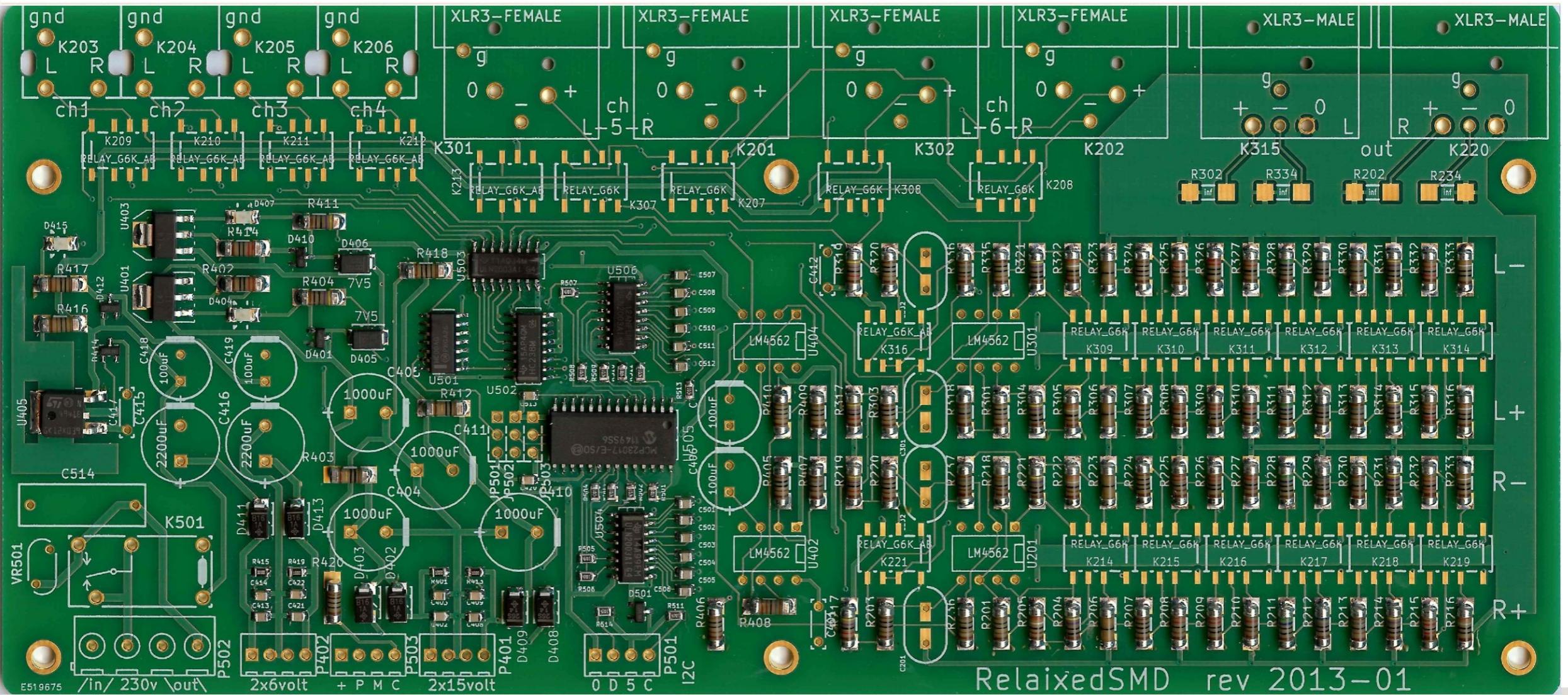
Library Editor



Other Cool Features

- ❖ Scriptable
- ❖ ULPs (User Language Programs)
 - ❖ Special language, like C but worse
 - ❖ Ties into core functionality
- ❖ Command Line Interface
 - ❖ Some features available via OS command line





Designing a PCB

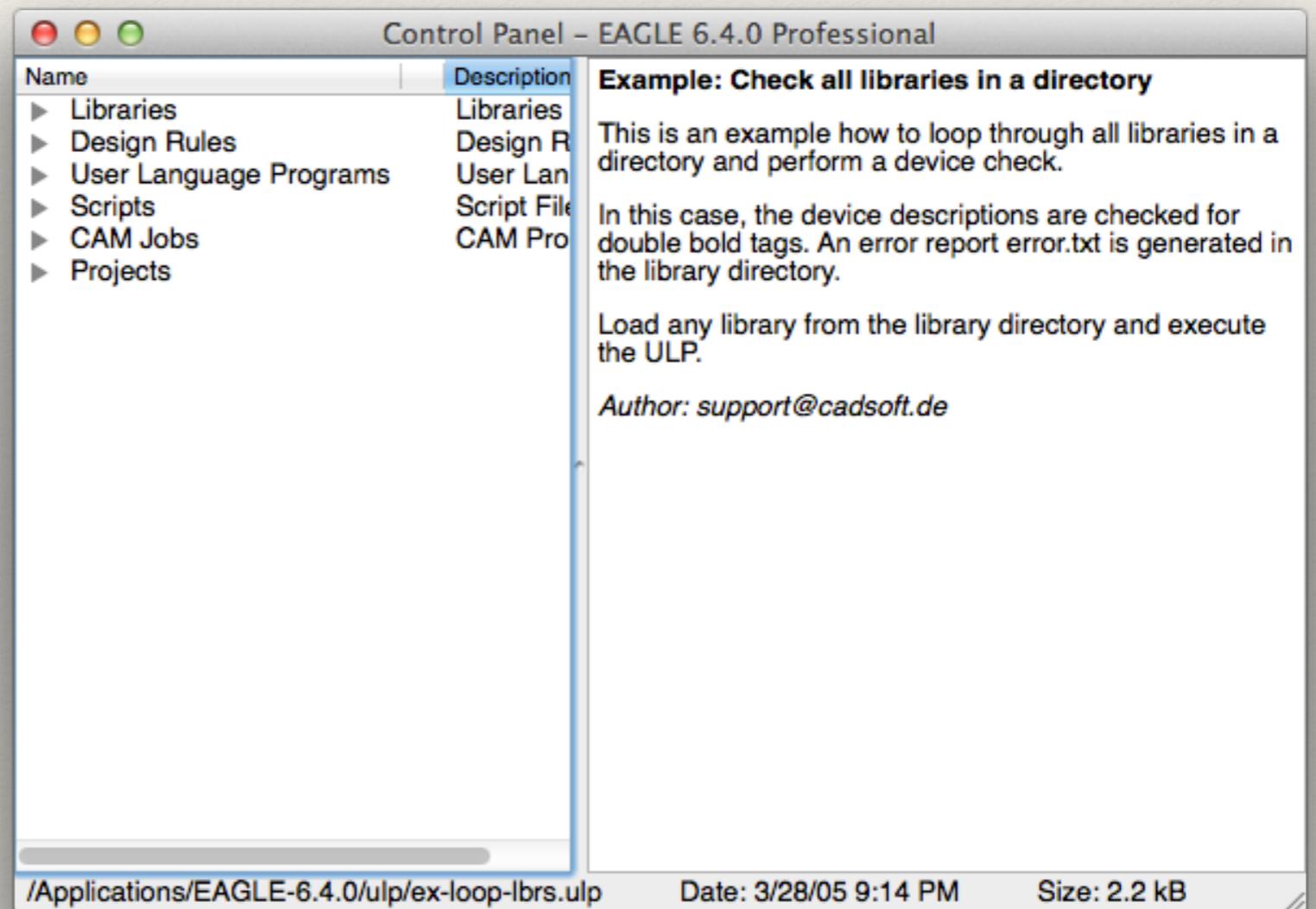
High-level Design Process

High Level Process

- ❖ Do your research!
 - ❖ Read datasheets
 - ❖ Study other designs - iterate
- ❖ Draw sketches, block diagrams
- ❖ Think about mechanical issues ahead of time
- ❖ Hardware is hard. Expect to iterate and prototype.

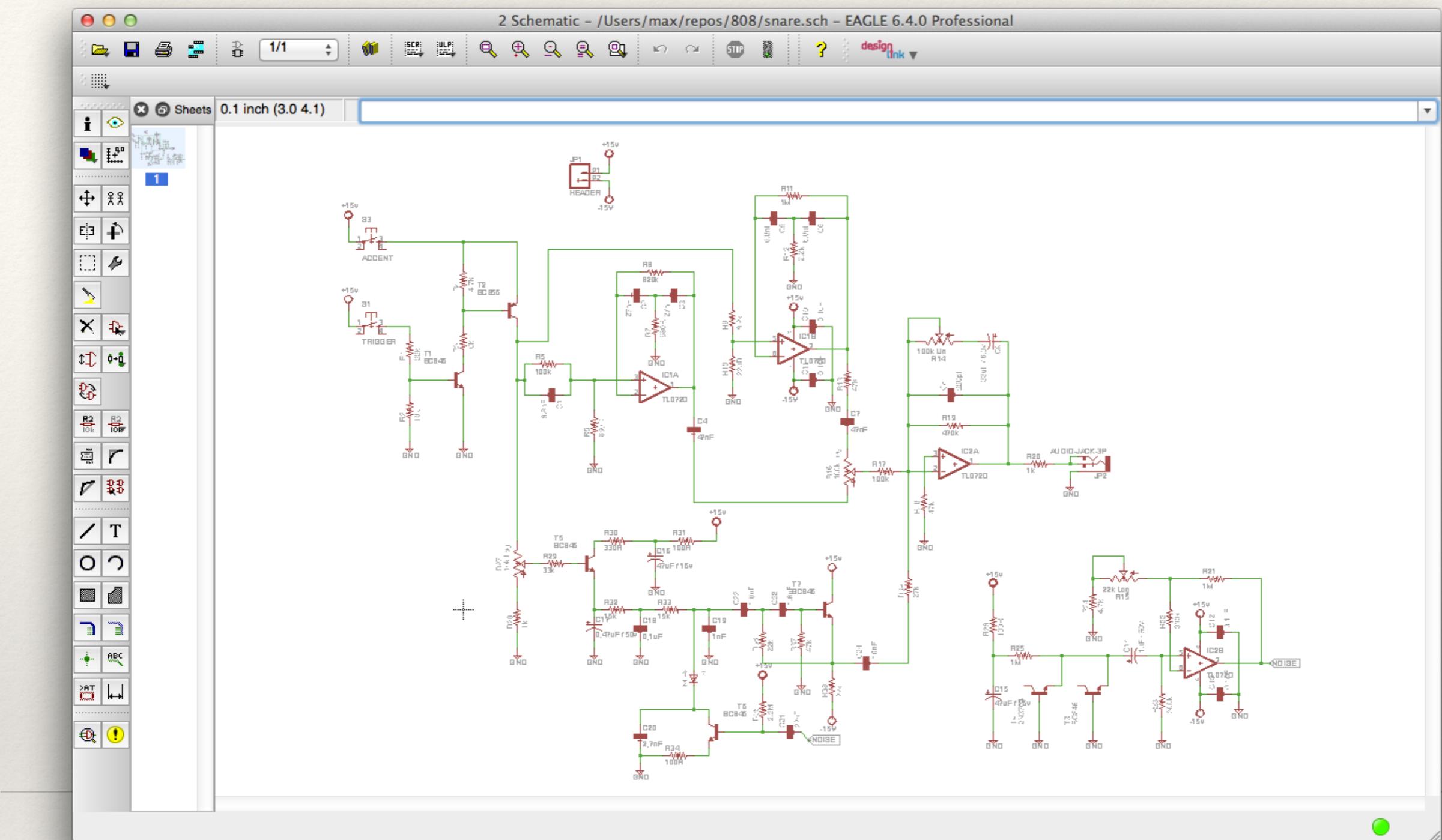
Starting Eagle

- ❖ Don't bother with Eagle Projects
- ❖ Use this screen to:
 - ❖ Create new Schematics
 - ❖ Open Schematics, Boards, and Libraries



Eagle Workflow

- ❖ Find or create Parts (also called "Devices")
- ❖ Add parts to a Schematic
- ❖ Define the electrical connections between Parts (wire it up)
- ❖ Lay out the physical Parts on your Board (PCB)
- ❖ Route the connections (draw the actual wires - "traces" - on the PCB)
- ❖ Export your Board for fabrication



Demo: Schematic Capture

Demo: Commands: ADD

- ❖ Type the first letter of the library name to jump around in the selection window
- ❖ Use arrow keys to open and close selection tree branches, Enter makes a selection
- ❖ You can search with asterisks: for example, '*sot*89*' to search for everything in a SOT-89 package
- ❖ Sticky ADDs: use 'Esc' to stop adding a part

Demo: Commands: NET

- ❖ Use NET to draw wires, not Wire!
- ❖ Use right mouse button to switch between bending styles
- ❖ Sometimes nets do not connect! This could be due to your grid settings or alignment of the planets:
 - ❖ Use MOVE to shake the part, should rubberband
 - ❖ Use JUNCTION to connect nets
 - ❖ Double check that you're not missing connections

Demo: Commands: MOVE, GROUP, ROTATE, MIRROR

- ❖ Right click while MOVEing rotates
- ❖ Use ROTATE to spin without changing origin (or enter arbitrary angle)
- ❖ Use MIRROR to reflect the part across its center
- ❖ Use GROUP to select multiple devices and wires
- ❖ Define a group with a box or polygon
- ❖ You can ROTATE, COPY, DELETE, etc. groups... so they're very useful
- ❖ Only one group at a time

Demo: Commands: CUT, COPY, PASTE

- ❖ Use COPY to copy and paste devices
- ❖ To copy and paste groups:
 - ❖ Click COPY
 - ❖ Use GROUP to select the group
 - ❖ Select Copy Group to Copy and Paste

Demo: Commands: NAME, VALUE, SHOW, INFO, SMASH, LABEL

- ❖ Use NAME to set the name of devices and nets (e.g. IC1 or Vcc)
- ❖ Use VALUE to set the value of the device (e.g. 47k)
- ❖ Use SMASH to separate the name and value from a device symbol so you can move them around (almost never use this on schematic)
- ❖ Use INFO to pop up a dialog box about a part
- ❖ Use SHOW to show the name in the status bar and highlight it (VERY useful for nets)
- ❖ Cool Feature: SHOW also highlights the same nets on the board
- ❖ Use LABEL if you want the name of your net or bus to be displayed
 - ❖ I like the arrow ones

Demo: Commands: DELETE

- ❖ Deletes closest thing
- ❖ Multi-level undo!

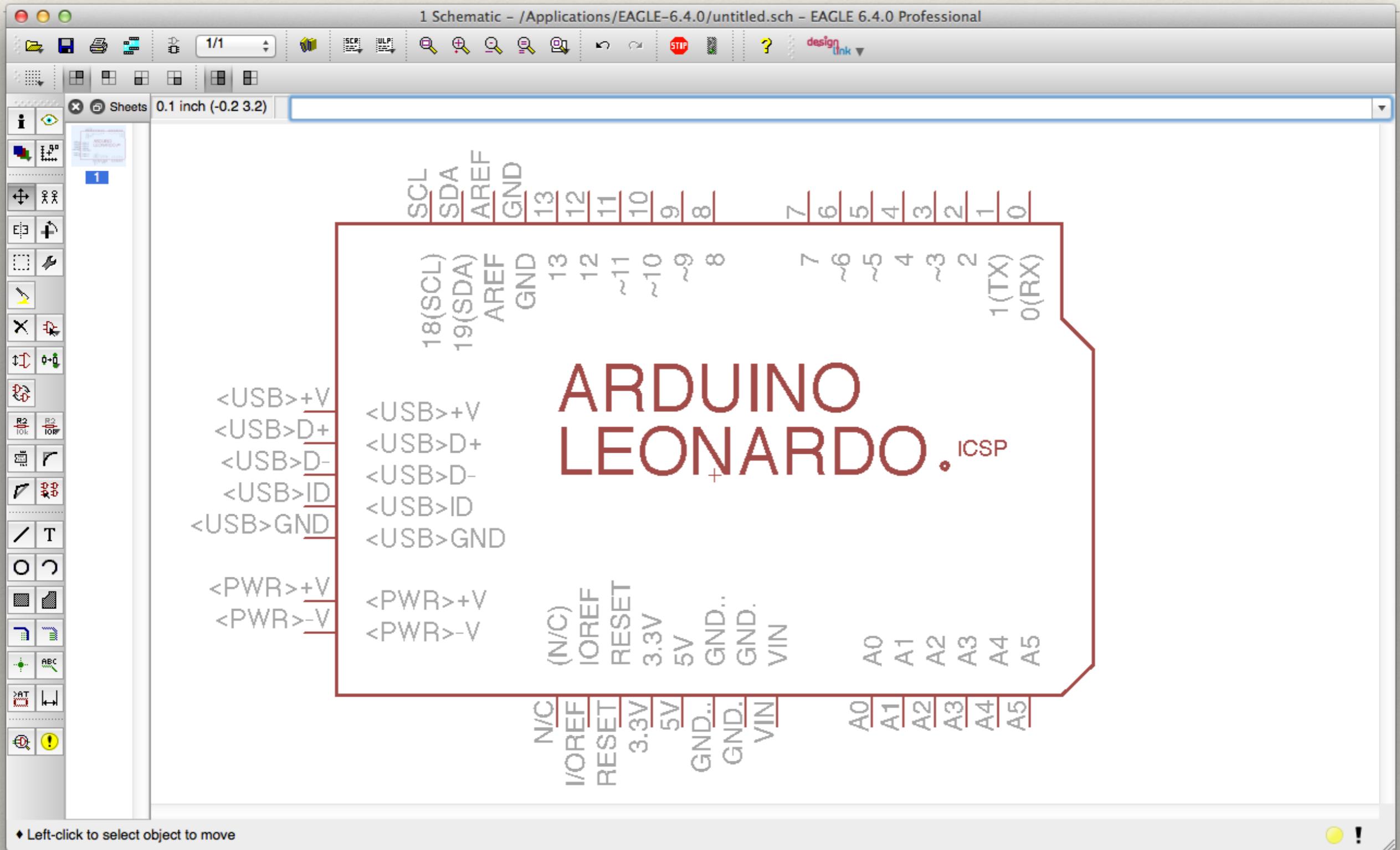
Demo: Commands: ERC

- ❖ Electrical Rule Check
- ❖ Checks to make sure you didn't make any bonehead errors in your Schematic
- ❖ Errors - pay attention, these are things that could cause your circuit to blow up
- ❖ Warnings - Not as urgent, but give them a look to make sure they're not important
- ❖ Can ignore:
 - ❖ If you rename a power supply net, "Supply pin x overwritten..."
- ❖ Do not ignore:
 - ❖ Warnings about things being close together (but not electrically connected)
 - ❖ "Only one pin on NET x..."

REVIEW!

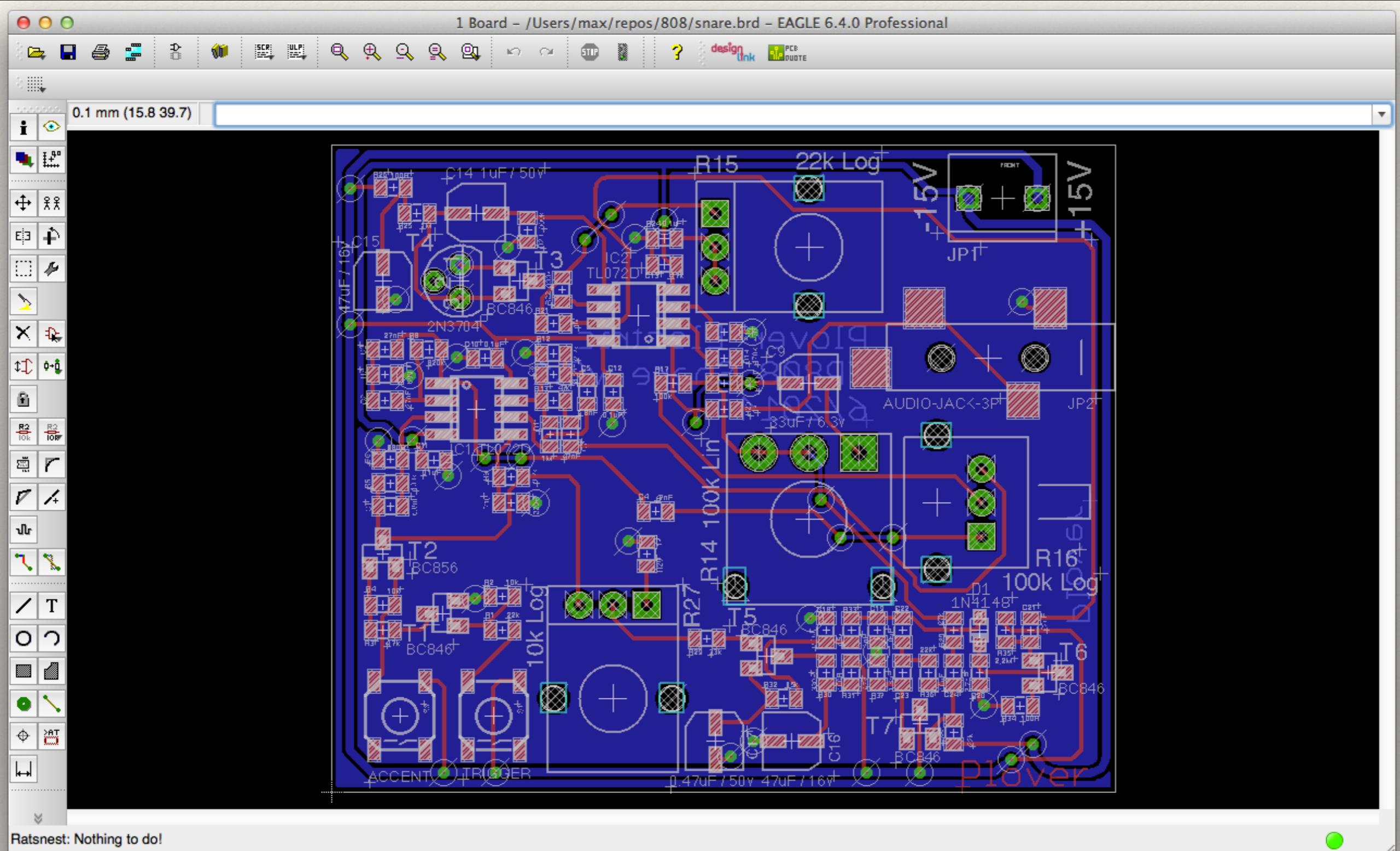
- ❖ Don't even think about starting your board yet
- ❖ Review your design
- ❖ Remember: hardware is hard, and also expensive.
Review it again.
- ❖ Get someone else to review your design
- ❖ Now is a good time to prototype weird or questionable circuits

Design your Arduino shield



REVIEW!

- ❖ Don't even think about starting your board yet
- ❖ Review your design
- ❖ Remember: hardware is hard, and also expensive.
Review it again.
- ❖ Get someone else to review your design
- ❖ Now is a good time to prototype weird or questionable circuits



Demo: Board Layout

Demo: Board Layout

- ❖ Mostly the same commands as schematic editor
- ❖ Added features for routing traces
- ❖ 'Airwires' are the funny thin rubber-banding wires that connect pins
- ❖ A 'ratsnest' is all the airwires together

Demo: Placing Components

- ❖ Place components to minimize crossed airwires
- ❖ Use MIRROR to flip components to the other side of the board
- ❖ You can't place component origins or pads outside of the size restrictions

Demo: Routing Airwires

- ❖ Use the ROUTE command to route airwires
- ❖ Choose which side you want to route on (top, bottom)
- ❖ Switching a route in progress from top to bottom will automatically add a via
- ❖ Your job: get rid of all airwires without violating the design rules
- ❖ Use RATSNEST a lot. Cleans up the airwires.
- ❖ Right mouse button cycles through 'bending modes'

Demo: Routing Airwires: Rules of Thumb

- ❖ It's best to route one direction on one layer, and orthogonally on the other. This makes routing easier, especially with a large number of wires.
- ❖ Prefer round or octagonal vias. Square ones are dumb except for marking something.
- ❖ USE WIDER TRACES. Size matters. Thicker = better if you have room. Lower impedance.

Demo: Layers

- ❖ Use the DISPLAY command to display the various layers of your board
- ❖ Understanding your layers is important!

Demo: Layers

- ❖ TOP -- Top side copper
- ❖ BOTTOM -- Bottom side copper
- ❖ PADS -- Through hole device pads
- ❖ VIAS -- Plated through holes without associated devices
- ❖ UNROUTED -- Airwires
- ❖ DIMENSION -- Board outline
- ❖ t/b DOCU -- Package element; only shows up in Eagle
- ❖ t/b PLACE -- Silk screen
- ❖ t/b ORIGINS -- Package crosshairs, needed to select parts
- ❖ t/b NAMES -- Package names, appears on silkscreen
- ❖ t/b VALUES -- Package values, usually appears on silkscreen
- ❖ t/b STOP -- Solder mask
- ❖ t/b CREAM -- Solder paste ("stencil")
- ❖ t/b KEEPOUT -- No-Go areas for packages
- ❖ t/b/v RESTRICT -- No-Go areas for traces
- ❖ DRILLS -- Plated through holes
- ❖ HOLES -- Non-plated holes
- ❖ MILLING -- Areas to be removed by milling
- ❖ DOCUMENT -- Additional documentation
- ❖ REFERENCE -- Reference marks

Demo: Ripping up Routes

- ❖ It's often necessary to totally reroute a net
- ❖ RIPUP on a routed trace turns it back to an airwire
- ❖ RIPUP on an airwire rips up until it hits another airwire or a pin
- ❖ RIPUP button + GO rips up everything (remember, you do have an undo command!)
- ❖ Ripping up will ONLY affect the visible layers, so you can easily rip up just the top or bottom parts of a board

Demo: Grids

- ❖ Use this to switch grid size / units
- ❖ Changing units here will affect wires, drills, etc.

Demo: Semi-Advanced: Polygons

- ❖ Polygon fill -- Activate w/ RATSNEST Deactivate w/ RIPUP (Must hit outer edge)
- ❖ Delete on polygons deletes single points; the polygon is deleted when only two points remain
- ❖ Cool Feature: Use polygons to create automatic ground planes!
- ❖ You can also use polygons to make automatic power planes - this is useful for high current applications

Demo: Advanced: Autorouting

- ❖ Autorouting is nice, but it often doesn't do what you want
- ❖ Component placement is critical to getting a good autoroute
- ❖ Hinting the autorouter by manually routing important signals helps
- ❖ AUTO GND -- auto route GND
- ❖ AUTO ! GND -- auto route everything except GND
- ❖ Net CLASS determines the properties of autorouted traces:
e.g. a power class which has thick traces

Demo: DRC

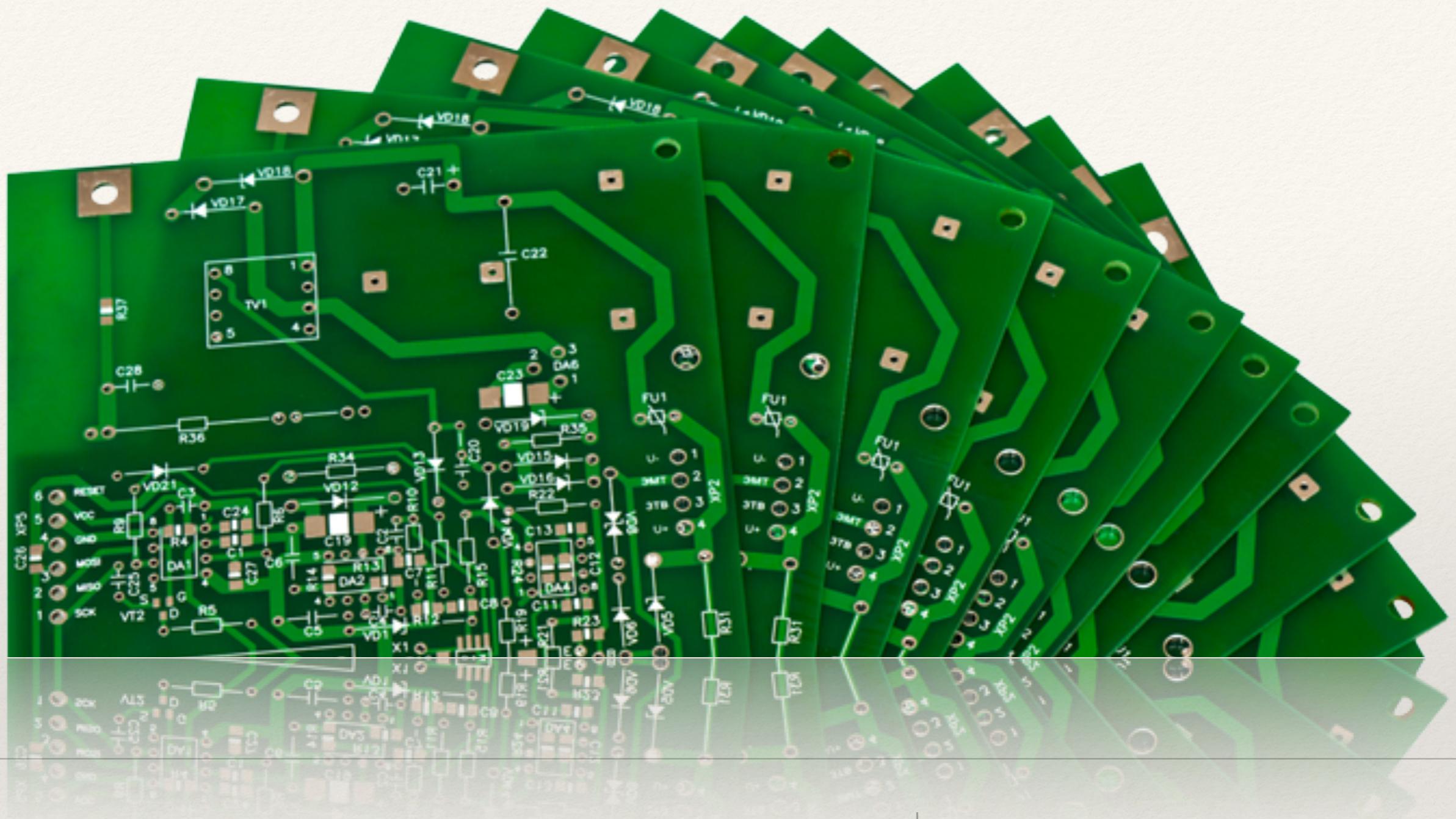
- ❖ Design Rule Check
- ❖ Like ERC, this is a MUST DO
- ❖ Remember: Only visible layers are checked
- ❖ Post-DRC dialog graphically finds errors (turn on centering)
- ❖ Violations leave behind highlights which can be cleared in the Post-DRC dialog
- ❖ Can load a DRU file (previously saved or downloaded from a fab house)
- ❖ Check settings against a capabilities document from a fab house

Demo: DRC

- ❖ When to care:
 - ❖ Do NOT violate your board house's fabrication limits for your traces. These are hard limits.
 - ❖ Care more if you are using automated assembly
- ❖ When not to care:
 - ❖ Do not care if your text violates the rules: smudgy text is irrelevant
 - ❖ Ignore clearance errors when you are really trying to connect two things and EAGLE doesn't understand

Demo: Sanity Check

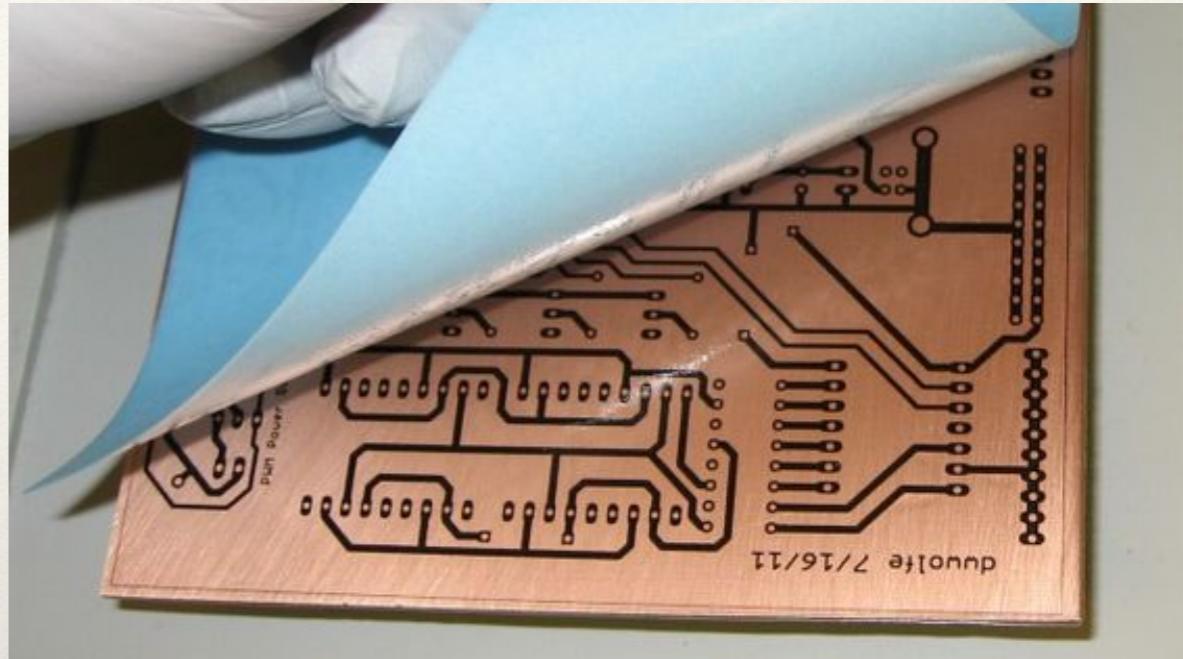
- ❖ Print out board and place components on the printout (stick through hole components through the printout!)
- ❖ Double check mechanical interfaces:
 - ❖ Make sure there's room for wires that come out of connectors
 - ❖ Make sure that the connectors face the right way
 - ❖ Check if there is some order that you have to place components (e.g. a large connector near a small surface mount part)



Getting it Done

Fabrication Methods

- ❖ A steady hand and an Exacto knife (small boards only, don't do this)
- ❖ Do it yourself, using bare copper board, an iron, and nasty chemicals (pretty icky)
- ❖ Send the design to a commercial firm which makes two layer, plated-through-hole prototype boards for dirt cheap:
 - ❖ Advanced Circuits deals
 - ❖ OSH Park

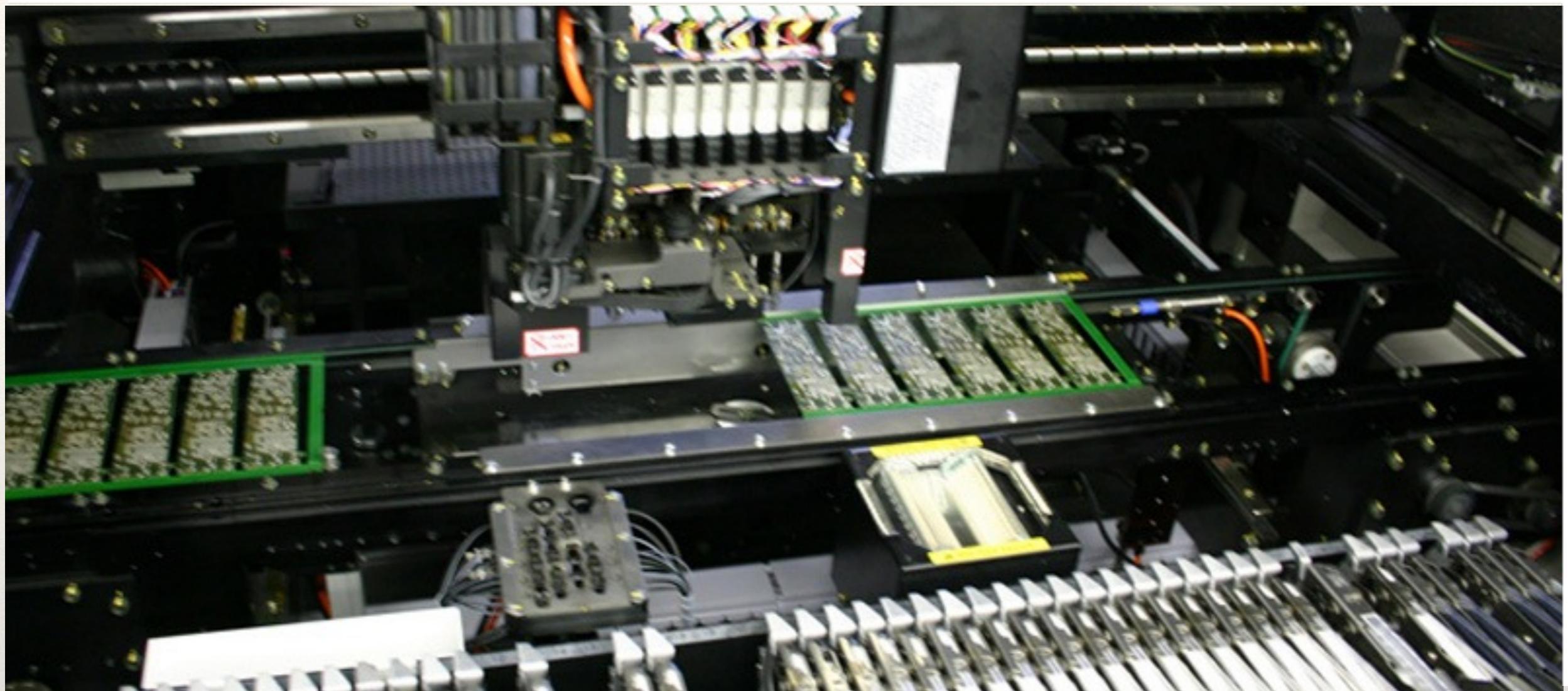


Demo: Exporting Gerbers

- ❖ Download Sparkfun's CAM file
- ❖ Load the CAM file and run it on your board
- ❖ Check the outputted Gerber files in an external viewer
(gerbv or MCN viewer)
- ❖ Zip gerbers and upload to Advanced Circuits'
Barebones page

Demo: Uploading to OSH Park

- ❖ Drag .brd file into OSH Park
- ❖ Check the preview images
- ❖ Done!

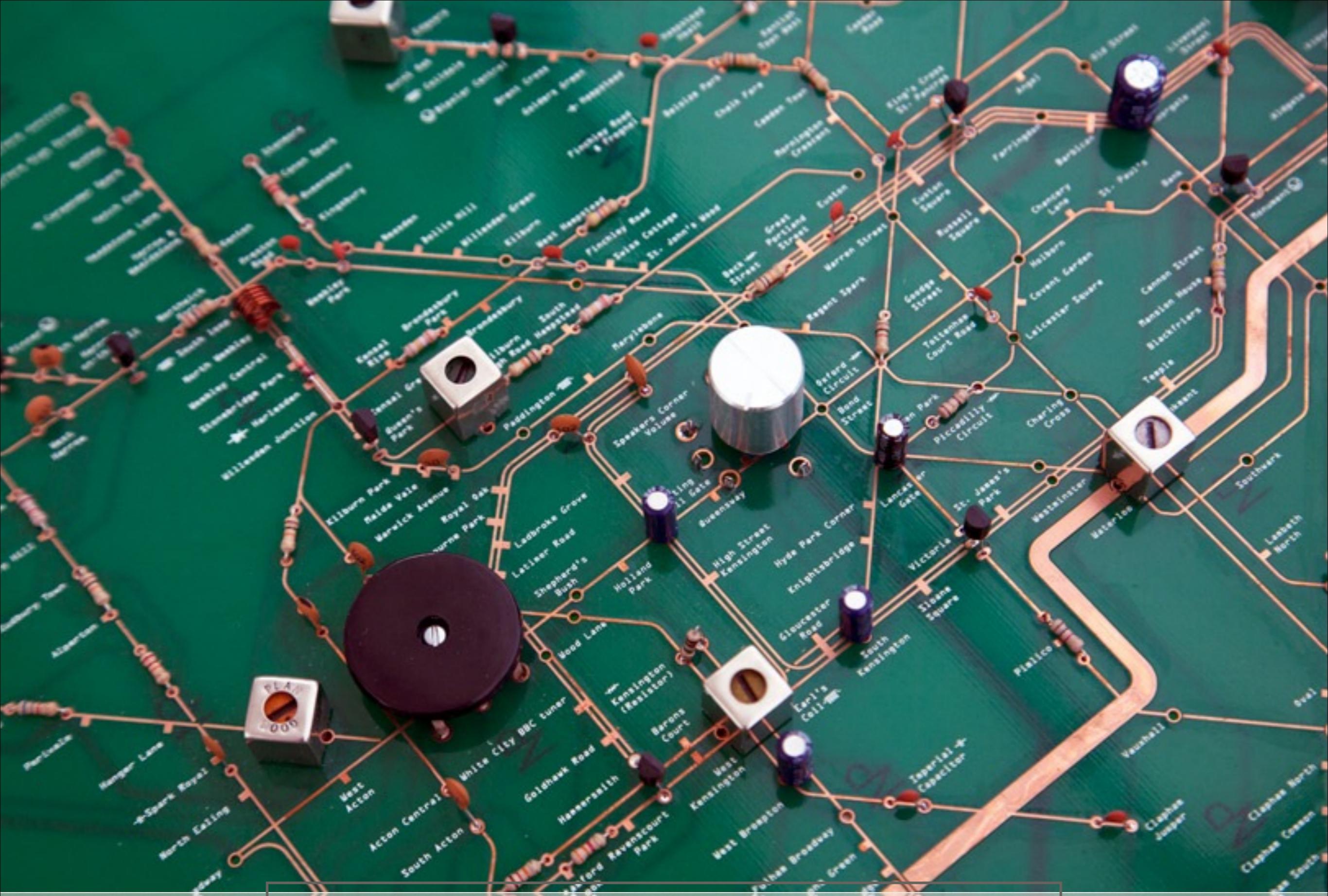


Next Steps

Assembling Your PCBs

- ❖ Hand Assembly
 - ❖ Easiest for small runs of boards with a limited number of components
- ❖ Automated Assembly
 - ❖ Send your Gerbers and Centroid files to an assembly house
 - ❖ Use a Pick and Place machine to populate your boards
 - ❖ Expensive, good for medium and large runs





Remember: It's an Art

Thank You

- ❖ Classes: NYCResistor.com
- ❖ Tutorials:
 - ❖ cadsoftusa.com
 - ❖ sparkfun.com
- ❖ Community:
 - ❖ NYC Resistor Microcontrollers Mailing List
 - ❖ NYC Resistor Craft Night (Thursday evenings)
 - ❖ forums.adafruit.com
 - ❖ forum.arduino.cc
 - ❖ electronics.stackexchange.com
- ❖ Me:
 - ❖ Max Henstell
 - ❖ max@mhmaaa.xxx

