Designing with Kicad

This is not a Tutorial

The web is full of tuorials: just pick one

- If unhappy pick another

Or you can run the application and read help

- As a general rule, I tend to prefer real documentation
- If documentation accompanies the program, that is the best one

This is just a collection of ideas and suggestions

- You won't became an expert kicad user by reading this
- Moreover, you may as well disagree with me

Overview of Kicad

Kicad is a Free Sofware application to design PCBs

- It covers both schematic design and PCB drawing
- It has some extra features like 3D models, that we ignore for simplicity

The main schematic tool is called "eeschema"

- Schematic files, called "*.sch", are ASCII texts
- It relies on a library of components
  - There is a rich installed library of components
  - Many components are found online, with a free license
  - You can design yours, as usual
- The output you most use is a netlist file

The PCB design tool is called "pcbnew"

- The file is called "*.kicad_pcb", and it is an ASCII text
- It relies on a library of footprints
  - There is a rich installed library of footprints
  - Many footprints are found online with a free license
  - You can design yours, as usual
- The output you most use is a set of gerber files for production

Tools and Common Workflow

Kicad includes other two important tools:

- "cvpcb" is used to associate footprints to components
  - This is a boring but necessary step in the design workflow
  - A well-written component already lists the few possible footprints
- "gerbview" (or "gerbv", from the geda suite) can be used to verify your output
  - Never ship a gerber without looking carefully at it

So the standard workflow looks like this:

- eeschema: make the schematic diagram and save the netlist
- cvpcb: read netlist, associate footprints to components
- pcbnew: read netlist and footprints, make the PCB and plot gerber files
- gerbv[iew]: check the output
- email/web: ship the files and wait for the PCB to reach you

Finally, mounting

- Either you buy components, and solder yourself
- Or, at extra cost, you buy components, pack them again and ship for mounting
  - For this you need the BOM an position files, saved by pcbnew
- Or, at extra cost, your manufacturer does everything for you
  - You must be most precise with your BOM file

The one-man-company design workflow

For simple projects, I prefer a different workflow

- 1- Lay out the schematics with the most important components
- 2- Draft mechanics and place core components
- 3- Add more items to the schematic diagram
- 4- Adjust mechanics if needed
- 5- Place and route the new items
- 6- Goto 3
- 7- Add anything optional that may be useful and fits mechanics
- 8- Place and route these extra things
- 9- If something else fits, goto 7
- 10- Remove unused "options" before the mounting step

Everything must be performed with later steps in mind

- Please consider the cost and software effort of anything you add
- Please wonder what the client/user may unexpectedly ask
- Think about mechanics, power consumption, heath, noise, ...

And please be helpful to your software mate

- I/O channels, logging, debugging, ...

Suggestions from a frequent user (1/2)

It's trivial stuff, but usually untold

In schematic drawing

- Make drawings small, or you'll use up too many pages (hard to manage)
- Default resistor, inductor, etc may be too big: make yours
- Use smaller-than-default text: 0.04 or 0.03 inches is best
- Use labels rather than wires, so you can change pins while routing
- Add diagnostic leds for your microcontroller
- Always include a power-on led
- If it fits, add self-test circuitry
- Be generous with 0R resistors for options and tests
- Be generous with 0R1 or 1R0 resistors as current-meters

Suggestions from a frequent user (2/2)

In PCB drawing

- Place the pin1 marker outside of the component
- Write component values outside of the component
- Place the grid origin in a meaningful place (and mark it!)
- Always place pin-strip devices on a 2.54mm grid
- Place mounting holes at an even distance (e.g, 5mm grid)
- Add extra "mechanical" copper for all connectors
- Make pads bigger than suggested, for hand soldering
- Use square-angle routing between different layers
- Bless the board with a unique name, funny if possible
- Provide a "sticker" area for the serial number
- Proudly sign your device

My personal Suggested sizes

- References and values: 0.7mm (1.4 tick) or 0.5mm (0.1 tick).
- Tracks: 0.2mm, 0.4mm, 0.8mm
- Clearance: 0.018mm
- Grid: 1mm (or 0.5mm) for components, 0.25mm for tracks
- Vias: 0.8/0.6 or 0.4/0.2 if needed; bigger for power.