

Open Source Circuit Design w/KiCAD

Jenner Hanni

<http://jennerhanni.net>

KiCAD

Started in 1992 by Jean-Pierre Charras

Written in C++

Cross-platform (Linux/Mac/Win)

Expanded to three main developers

Lots of additional contributors

Licensed under GNU GPL v2

CERN



“We think that KiCad can do to PCB design what the gcc compiler did to software: ensure there are no artificial barriers to sharing so that design and development knowledge can flow more freely.”

Contributions



- differential pairs, matching track length
- more mechanical layers
- interactive routing
- improved graphic abstraction layer using OpenGL, Cairo
- starting to change the library management system to move away from cvpcb

Future Plans

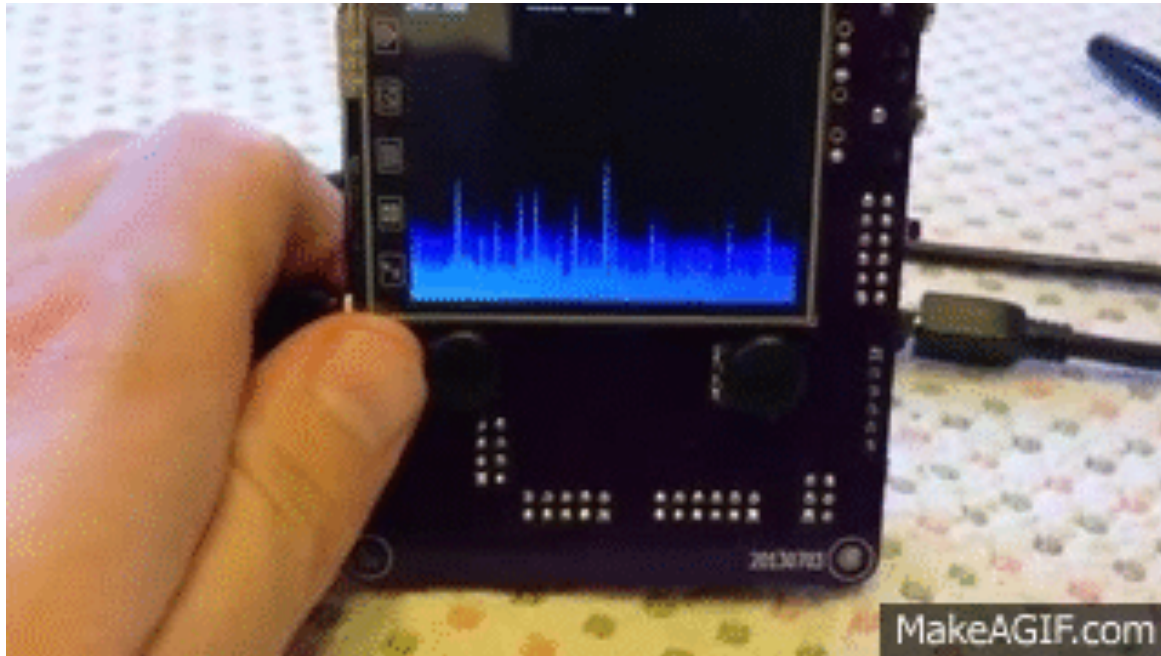
- production-ready design rules check (DRC)
- clean up UI and improve usability
- support split planes, differential pairs
- "circuit simulation as easy as LTSpice"
- cleaner codebase
- more behind-the-scenes netlist attributes



Made with KiCAD

Portapack for HackRF

<https://github.com/sharebrained/portapack>



Mooltipass

<https://github.com/limpkin/mooltipass>



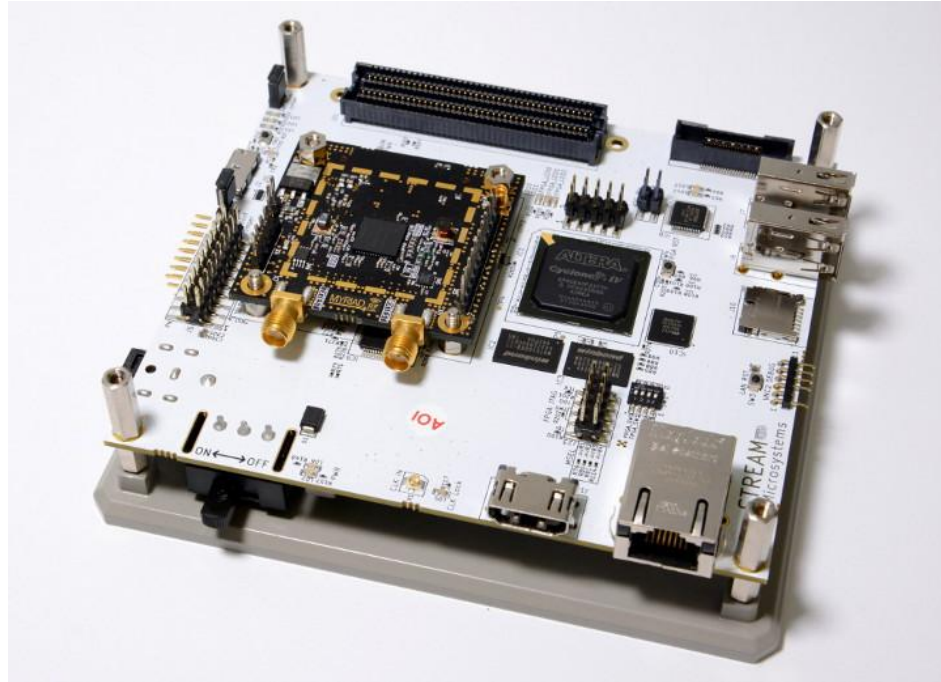
Atreus Keyboard

<https://github.com/technomancy/atreus>



MyriadRF STREAM

<https://github.com/myriadrf/Stream/>



Getting Started

Getting it

Old stable from 2013, pre-CERN

New stable coming in July 2015

[Linux Helper Script](#)

[Winbuilder](#)

[OS X Nightly Builds](#)

**if compiling from source, [read this first](#).

Read the Docs

[Getting Started in KiCad](#) - "Essential and concise guide to mastering KiCad for the successful development of sophisticated electronic printed circuit boards."

[Tutorials](#) - Chris Gammell (Contextual Electronics) and Ashley Mills

- [Simulating KiCAD circuits in SPICE](#)
- [3D Parts using OpenSCAD and Wings3D](#)

[KiCAD Documentation](#)

- | | |
|------------------------------------|---------------------------------------------|
| - <u>KiCad</u> | - <u>Eeschema</u> |
| - <u>Cvpcb</u> | - <u>Pcbnew</u> |
| - <u>Gerbeview</u> | - <u>Page Layout Editor</u> |

Demo

Importing Eagle projects

Importing an existing project -- Arducorder Mini

1. Run eagle2kicad.ulp in Eagle
2. Now a KiCAD .pro file exists with a .sch and .cache_lib
3. Open pcbnew directly and open the Eagle .brd as .kicad_pcb
4. Close pcbnew and open that KiCAD .pro file
5. Now the KiCAD .pro file has .sch, .cache_lib, and .kicad_pcb
6. Open .sch and start connecting the symbols to footprints

Library management

- Schematic component (symbol) library
- PCB layout module (footprint) library
- Do yourself a favor and make new libraries as soon as you start.

Projects on Github

Bart Massey's Desert Lizard example!

This is what a KiCAD github repo looks like.

- Forking, opening, contributing
- Files are not binary!
- Where's .cache_lib?

Grand plans

- production-ready design rules check (DRC)
- clean up UI and improve usability
- support split planes, differential pairs
- "circuit simulation as easy as LTSpice"
- cleaner codebase
- more behind-the-scenes netlist attributes

Stop Swearing
and Do Something

Community...

IRC: #kicad on freenode

Development Status

Forums: <http://kicad.info>

File Bugs! Fix bugs! Launchpad, Dev's Mailing List

KiCAD C++ Style Guide

Communication

Community seems to have good intent.

Remember the language and cultural barriers!

Don't bring hostility with you.

Example Bugfix

Finish

It sounds to me like this could be the \$10,000 program equivalent. It isn't yet. But I'm an optimist.

Wait until the next stable release to really commit....
... but start thinking about it now, open it up and play.