

Crash-Course KiCad

Ivo Blöchliger

Tech-Lab, Kantonsschule am Burggraben

28. Mai 2021

Disclaimer

- Ich bin ein NOOB
 - in Elektronik
 - erst recht in PCB-Design
- Nicht nur mein marginales Wissen teilen
- sondern auch neues von Euch Hinzugewinnen

Inhalt

- Einführendes Blabla
- Übersicht des Workflows
- Kleines Anwendungsbeispiel:
 - Microcontroller (z.B. ESP32 oder Arduino Nano)
 - AHT10 (Temperatursensor 4 pins)

Warum überhaupt PCBs?

- Anstatt Lochplatinen selber löten
 - Mit ähnlichen Zeitaufwand PCB designen
- Billige Produktion in China (CHF 2.- pro Board für 5 Stk)
 - Lieferung in ca. 2-3 Wochen
 - oder 5 Tage (CHF 20.- für Lieferung)
 - JLCPCB, Seeedstudio, PCBWay
- Coolness-Faktor ;-)

Warum KiCad

- Freie Software
 - Unabhängigkeit, offene Datenformate, Nachhaltigkeit
 - Ist mehr als einfach nur gratis!
- De facto Standard gratis PCB-Design-Programmen
 - Wird auch professionell eingesetzt.
- Alternativen:
 - EasyEDA (mit JLCPCB assoziiert, Web-basiert
 - Viele Community-Footprints, einfache Bestellung.

PCB für Maker

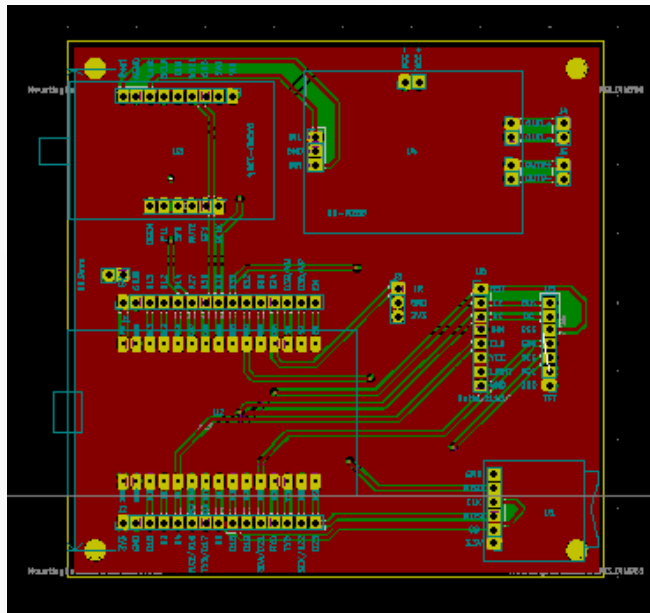
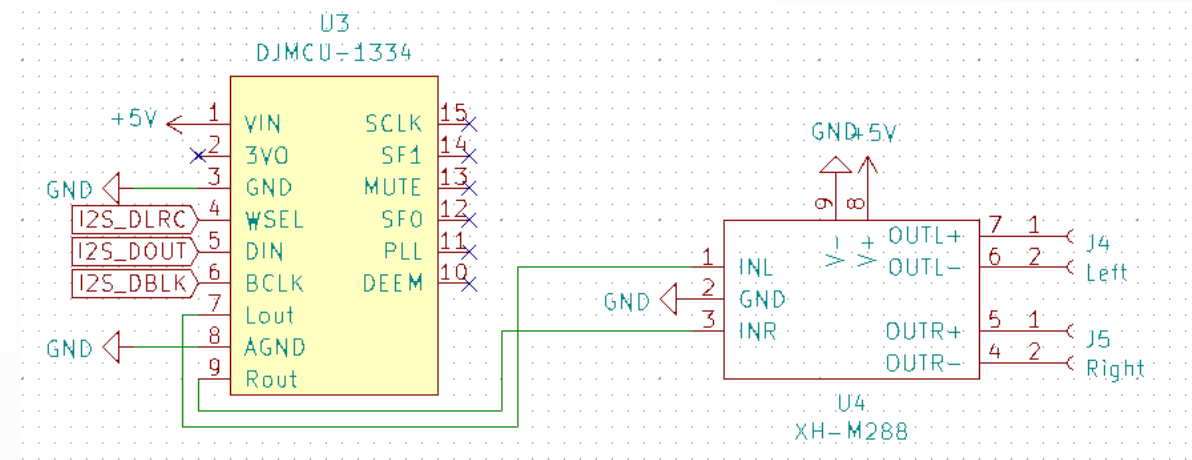
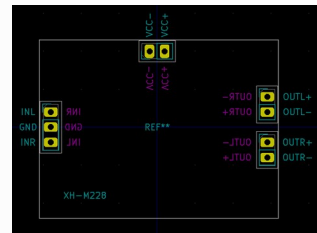
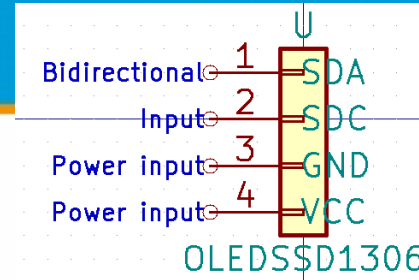
- Breakout-Boards
 - Praktisch, wiederverwendbar
 - Steckbrettfreundlich
 - Einfach mit Pinheader zu löten
 - Nicht für PCBs gedacht (macht auch niemand wirklich)
- Wenig bis keine Ressourcen für Breakouts in KiCad
 - Symbole und Footprints selber machen

Workflow

- Bauteile als Symbole
- Schaltung als Schema

- Footprints

- PCB



Export Gerber-Files

Organisation der Dateien

- Alle Dateien in einem Projekt-Ordner
 - Auch eigene Symbole und Footprints importieren
 - Können verändert, erweitert, verbessert werden, ohne das frühere Projekte kaputt gehen.
 - Zusammenarbeit/Backup ist einfacher (1 Ordner zippen, fertig).

Lets fetz!

- Video-Time
- AHT10
 - Temperatur
 - Luftfeuchtigkeit

