KiCad

Vom Schaltplan zur fertigen Platine - 4 Layer als Standard

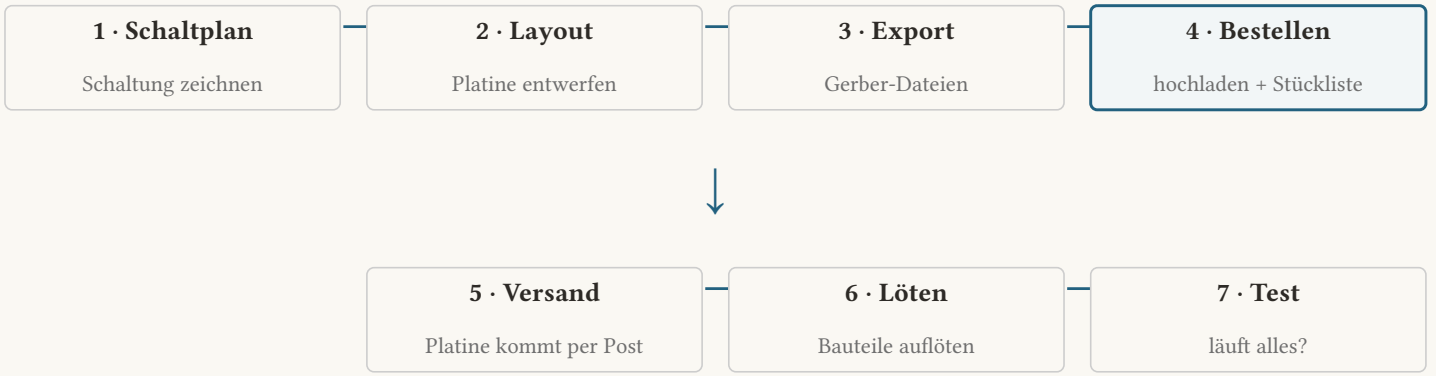
Your Name · 1980-01-01

ABLAUF

- **Was ist KiCad?** · Projekt & Schaltplan
- **Layout**: Platzieren · Routen · Zonen · Prüfen
- **4 Layer** statt 2 - warum
- **Praxis**: USB-C & 5 V · 0-Ω-Retter · Löten
- **Fertigung**: Gerber · Kosten · China vs. EU

1. Präsentation + Live Demo parallel
2. Wir zeigen euch parallel was ihr wissen wollt
3. Fragt direkt nach, spart euch die fragen nicht bis zum Ende
4. Zum Schluss: Wovon wollt ihr mehr sehen?

VON DER IDEE ZUR PLATINE



Heute: Schritte 1-3 in KiCad - ab 4 übernimmt die Fertigung.

SECTION

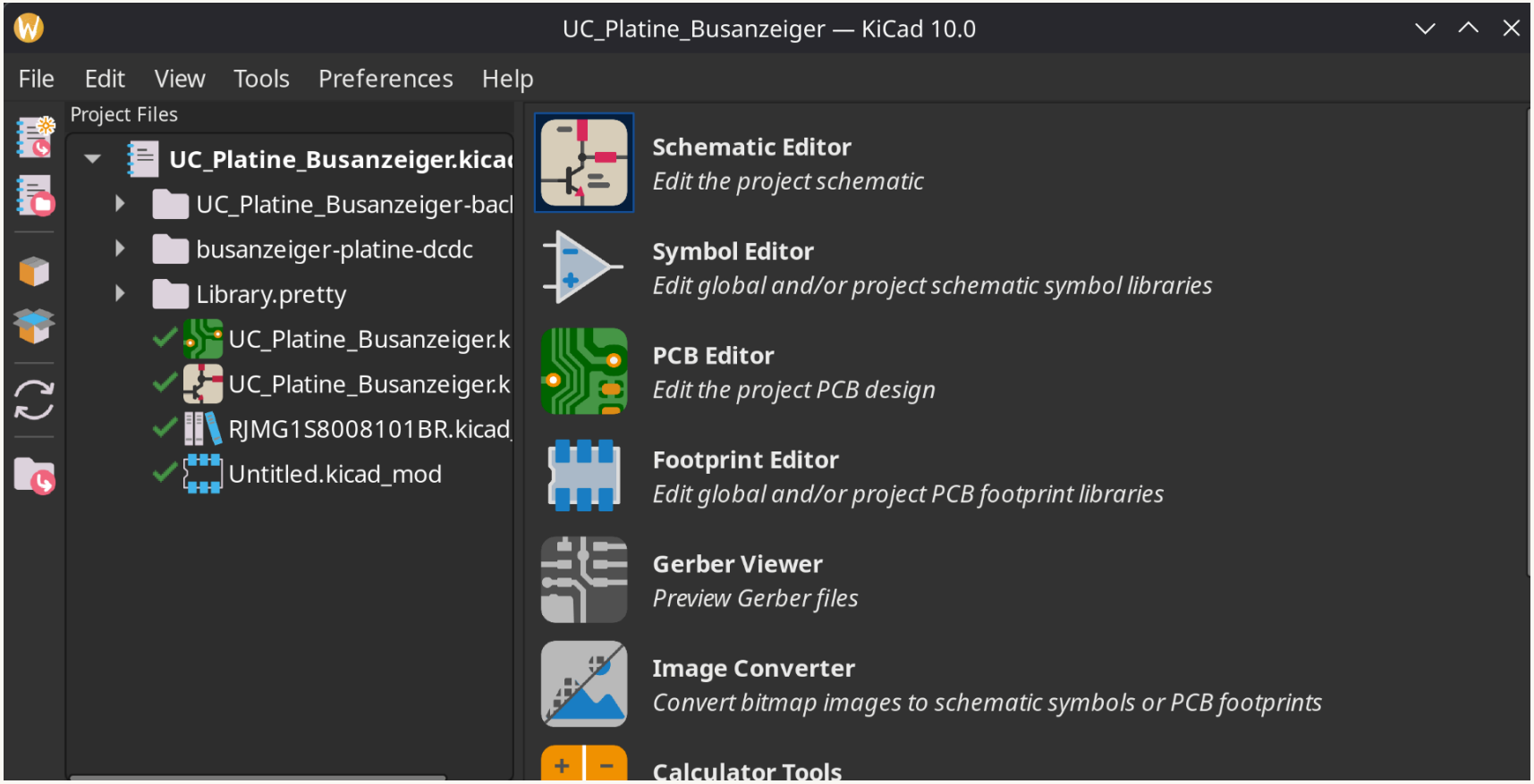
01

KiCad-Überblick

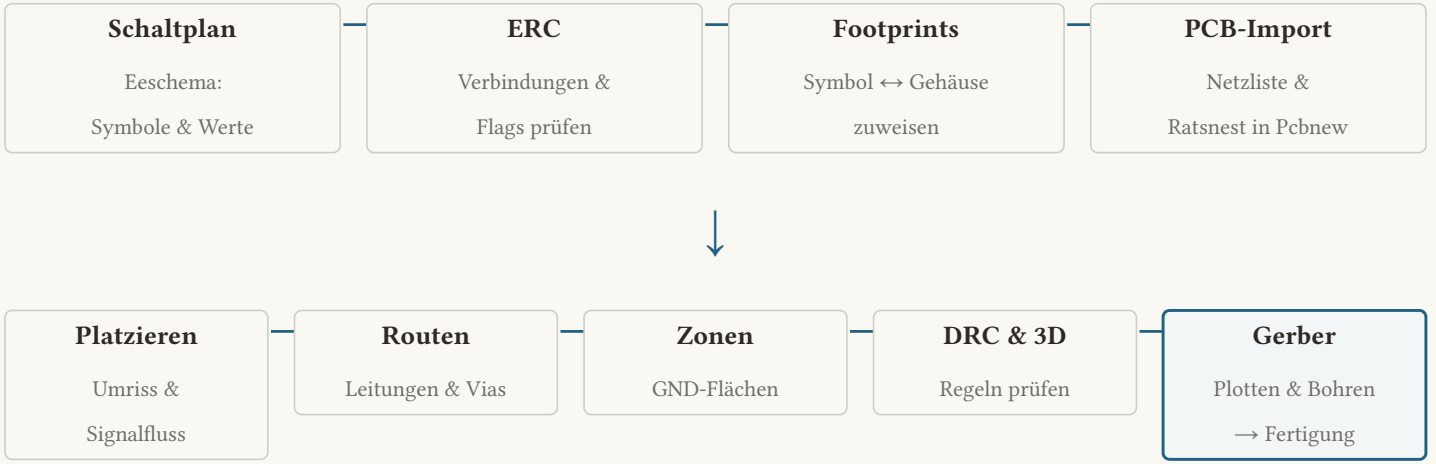
Werkzeuge & Workflow

WAS IST KICAD?

- Leiterplattendesignsoftware
- Schaltplan und Layout
- git support - Projektdateien sind Text
- Windows / Linux / Mac
- vom CERN, open source
- Eigentlich eine Sammlung von Programmen
- Python-Api für Automation


UC_Platine_Busanzeiger — KiCad 10.0
File
Edit
View
Tools
Preferences
Help
Project Files
UC_Platine_Busanzeiger.kicad
UC_Platine_Busanzeiger-bacl
busanzeiger-platine-dcdc
Library.pretty
UC_Platine_Busanzeiger.k
UC_Platine_Busanzeiger.k
RJMG1S8008101BR.kicad
Untitled.kicad_mod
Schematic Editor
Edit the project schematic
Symbol Editor
Edit global and/or project schematic symbol libraries
PCB Editor
Edit the project PCB design
Footprint Editor
Edit global and/or project PCB footprint libraries
Gerber Viewer
Preview Gerber files
Image Converter
Convert bitmap images to schematic symbols or PCB footprints
Calculator Tools

WERKZEUGKETTE

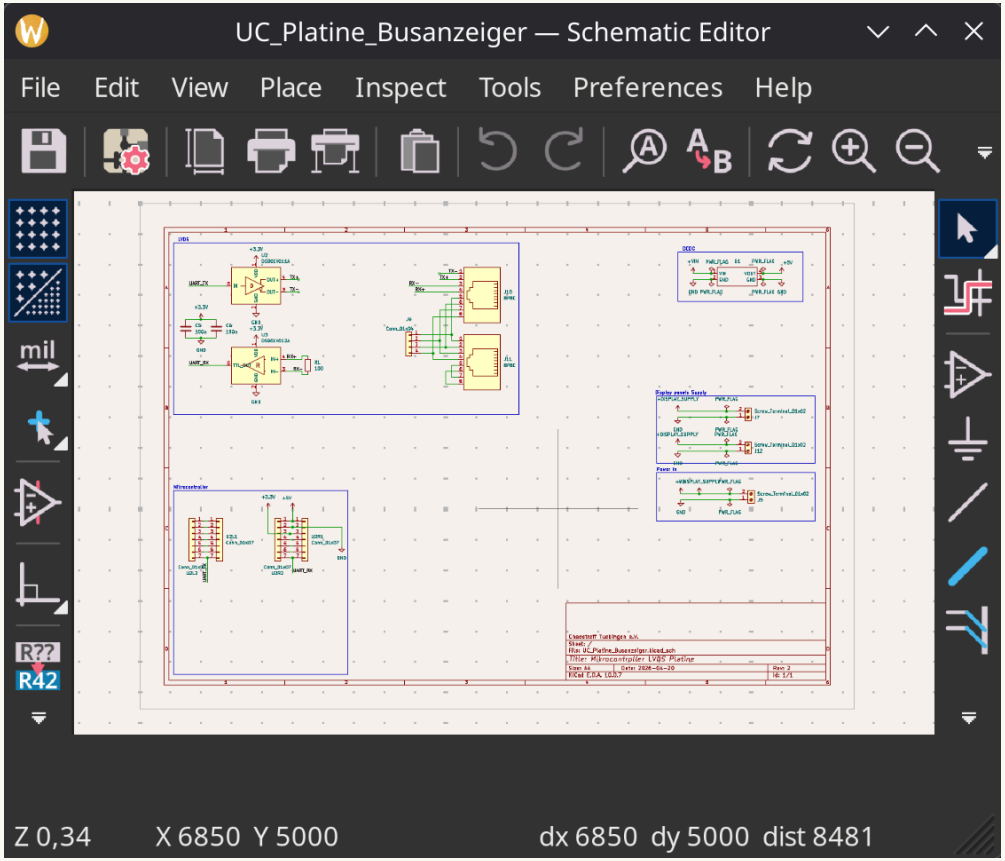


- Später etwas geändert? Ein Klick: Schaltplan → Platine aktualisieren
- Erst Electrical Rule check sauber haben, dann Layouten

PROJEKT & SCHALTPLAN

- An Beispielschaltungen aus dem Datenblatt orientieren - platzieren, verdrahten
- Sorgfalt und Zeit investieren - sauber gruppieren, nach Funktion:
  - Stecker & Montage
  - Stromversorgung - geht an alle anderen Gruppen
  - Zentraler Prozessor + Debug
  - Funktionale Gruppen - z. B. Schaltergruppe 1: Schalter + Sensor + LED + …

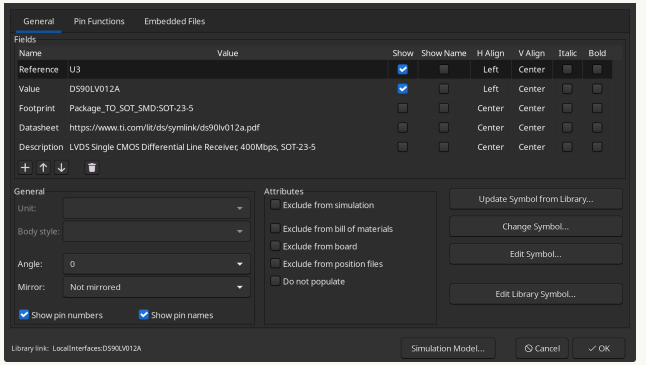
BEISPIEL: GRUPPIERTE SCHALTUNG



Jede Funktionsgruppe bekommt ihren eigenen Kasten - so bleibt der Plan lesbar.

BAUTEILE & LIEFERBARKEIT

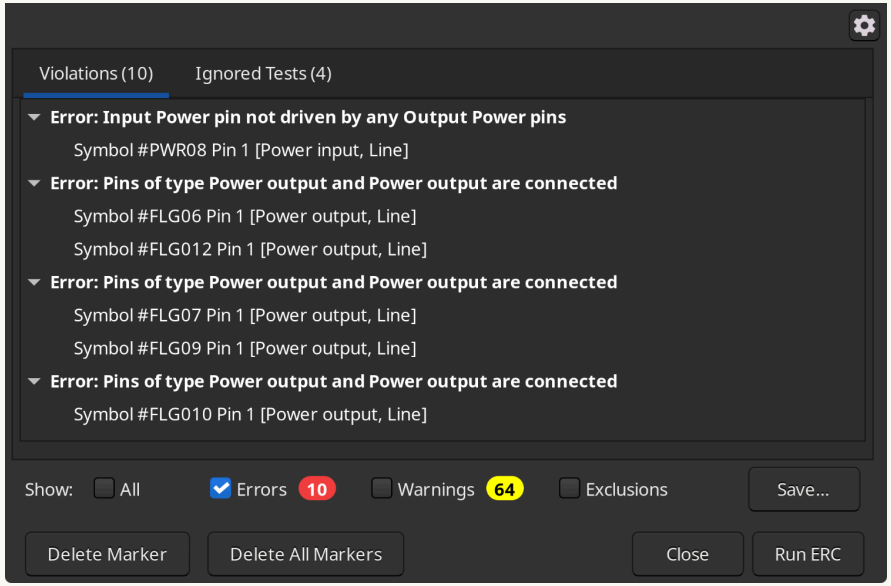
- Direkt beim Zeichnen konkret aussuchen - was es gibt und was lieferbar ist
- “irgendein 100 nF” ist okay - “der Schaltregler, aber in der 5-V-Variante” nicht
- Check: LCSC · Mouser · DigiKey - Lagerstand, Preis bei 1 / 10 / 100 Stück
- LCSC-Nummer als Feld ins Symbol ⟶ fertige Stückliste
- LCSC gibts als Plugin
- Schönster Schaltplan nützt nichts **ohne Bauteile**



Symbol-Eigenschaften: Felder, z. B. für die LCSC-Nummer

ERC - SCHALTUNG PRÜFEN

- `Prüfen → Elektrische Regeln prüfen`
- Findet: offene Pins · vergessene Drähte · fehlende `PWR_FLAG`
- Warnungen **beheben** - nicht wegklicken
- ERC grün ⟶ weiter zum Layout



ERC-Dialog: 10 Fehler - genau die müssen weg

SECTION

02

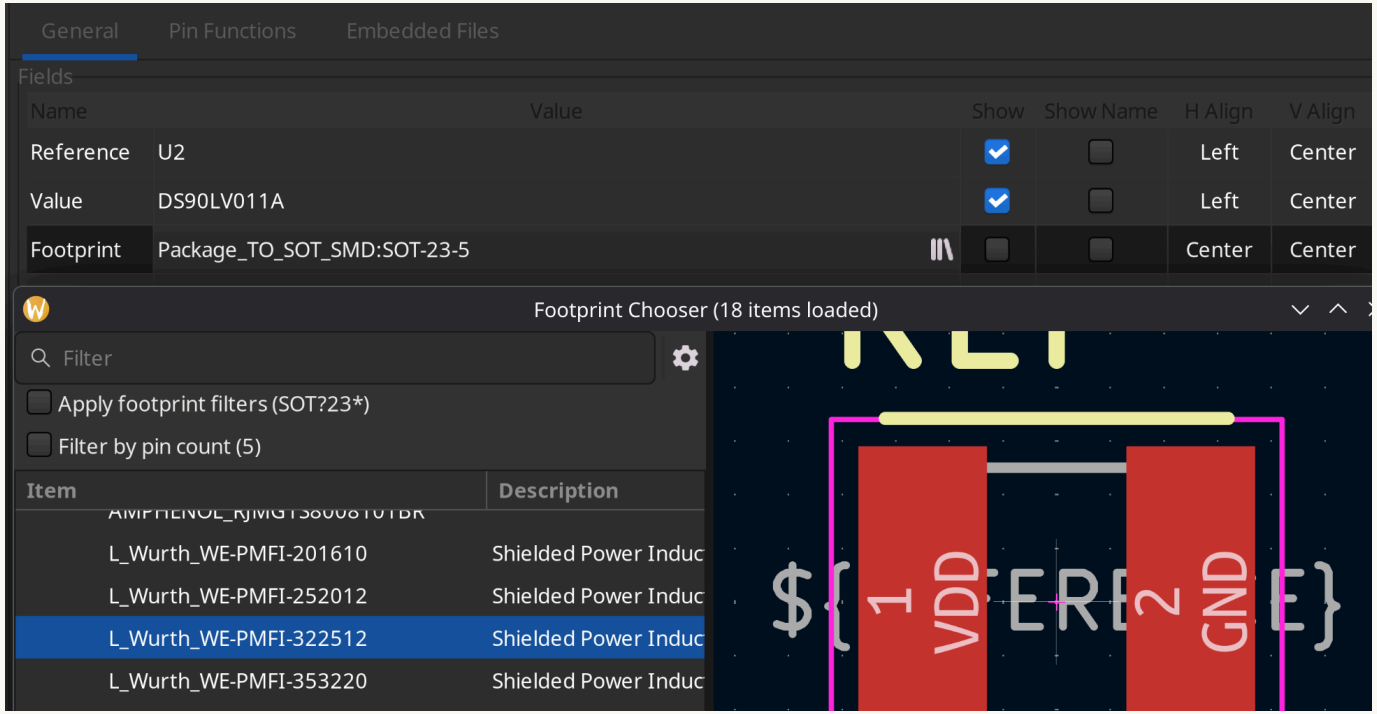
Layout-Workflow

Der Standard-Workflow

FOOTPRINTS ZUWEISEN

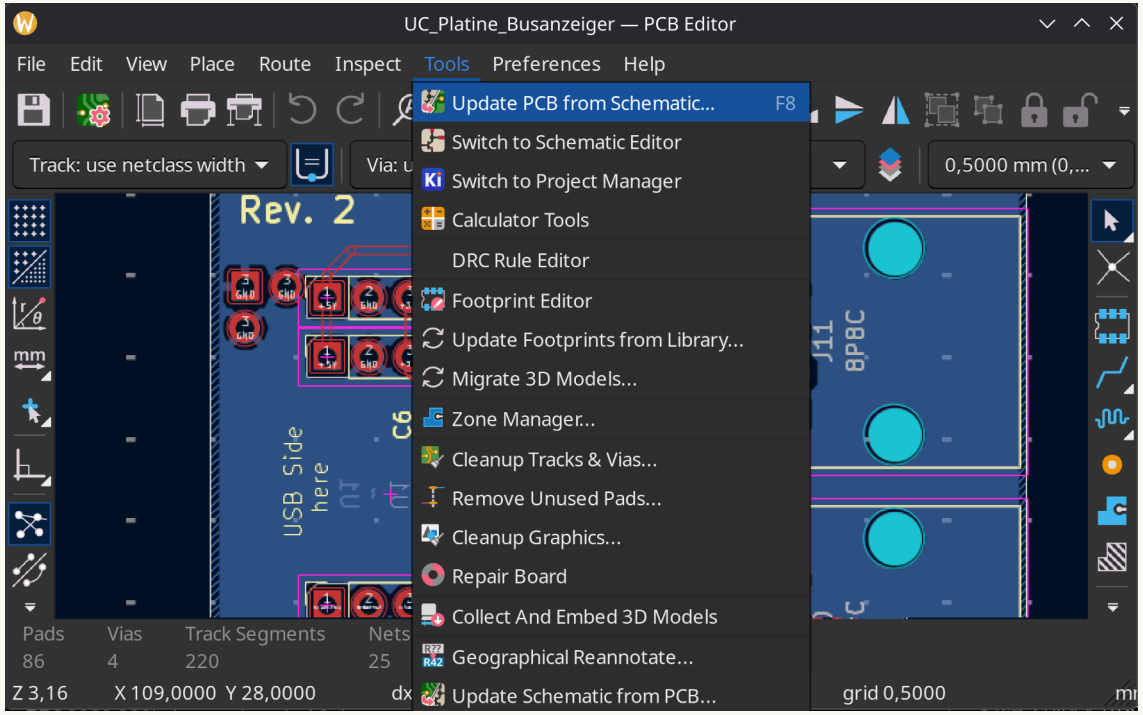
- `Werkzeuge → Footprints zuweisen`: Symbol ⟶ Gehäuse
- Mechanische Maße vorher im Datenblatt prüfen

FOOTPRINT CHOOSER



Filter + Vorschau: passt das Gehäuse zum Symbol?

PCB AUS SCHALTPLAN AKTUALISIEREN



`Tools → Update PCB from Schematic` (F8) - Bauteile landen auf der Platine

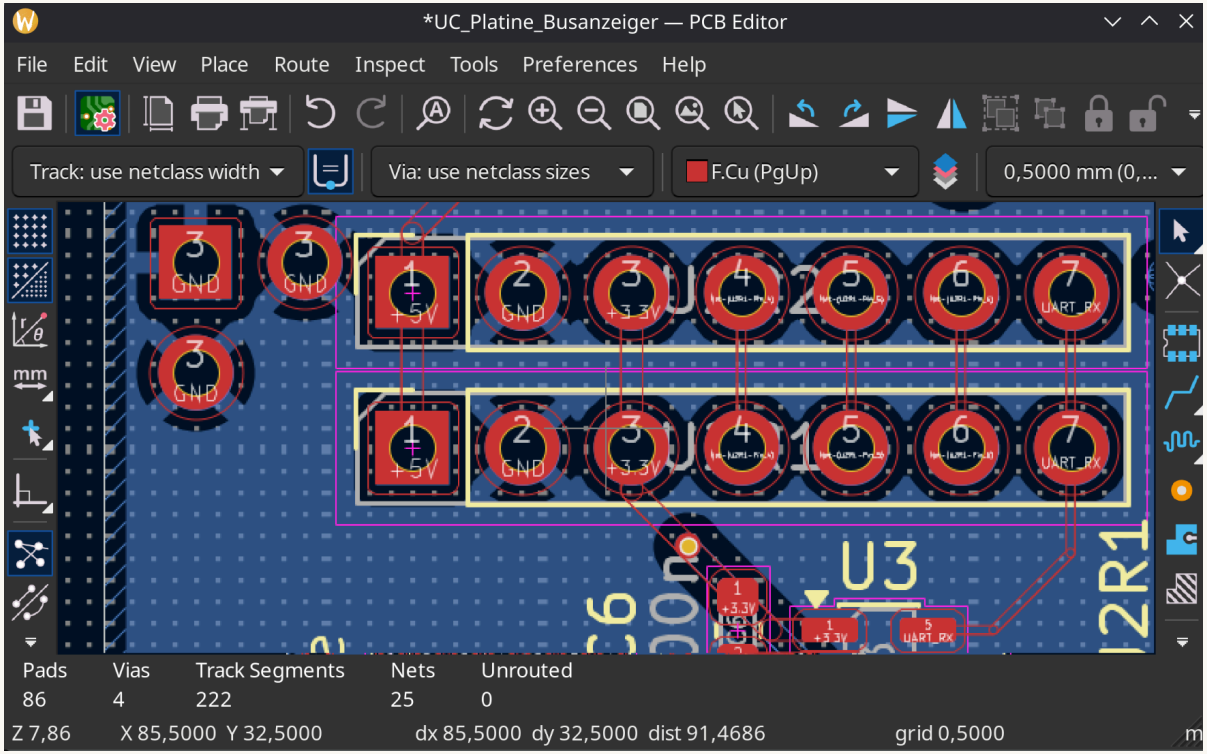
SETUP, UMRISS & PLATZIEREN

- Umriss auf **Edge.Cuts** (z. B. 80 × 50 mm)
- **Board-Setup** (Standard):
  - Signal 0,25 mm · PWR 0,5-0,8 mm
  - Via 0,6/0,3 mm
- Platzieren nach **Signalfluss**:
  - Stecker & Befestigung zuerst - als fix markieren
  - Dann Bauteile grob zusammenschieben wie auf dem Schaltplan gruppiert
  - Entkopplungs-Cs direkt an die Versorgungspins
- **Platinenbeschriftung**: auf wichtige Eckpunkte reduzieren
  - z.B. Stecker-Polarität
  - max Spannung
  - Versionierung

ROUTEN

- x - **kritische Signale zuerst**: Takt · PWM · Analog
- Kurze Wege, wenige Vias
- v beim Routen ⟶ Via, Lagewechsel
- PWR breiter (Netz-Klasse) · GND ⟶ Massefläche
- Raster: 1,27 / 0,635 mm (THT) · 0,25 mm (SMD)

ROUTEN IN PCBNEW



F.Cu - die dünnen Linien (Ratsnest) zeigen, welche Verbindungen noch fehlen

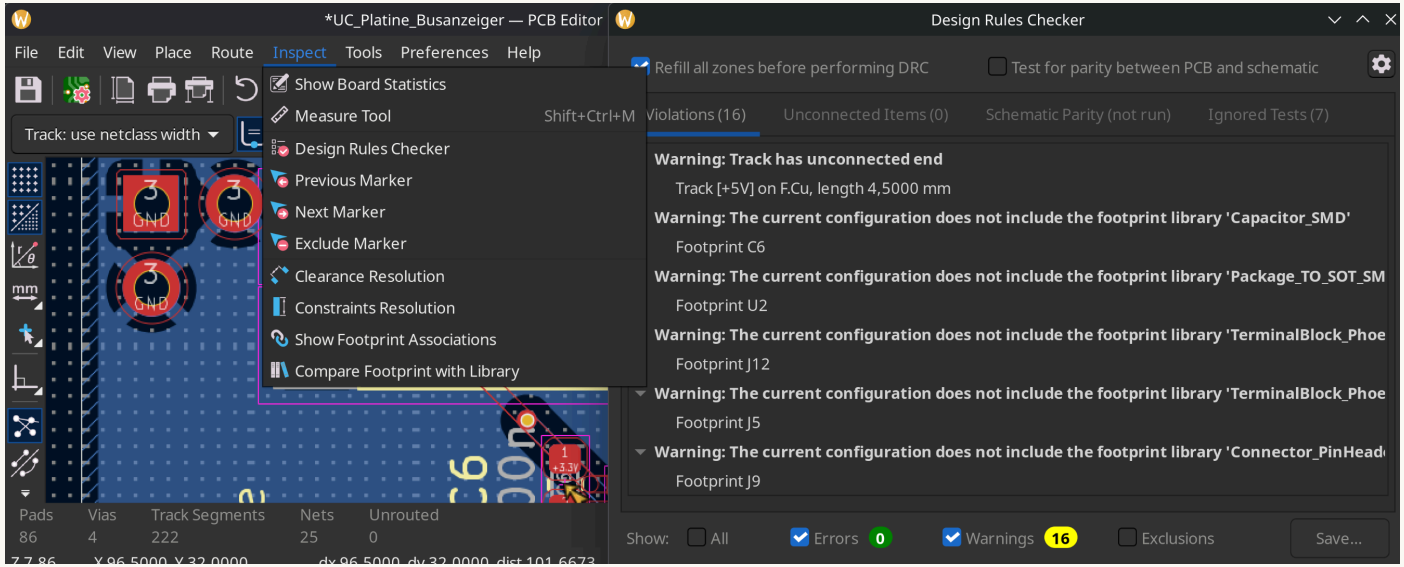
ZONEN - GND-FLÄCHE

- Werkzeug **Kupferfläche**: Netz **GND** · Lage wählen
  (4 Layer: In1.GND · 2 Layer: B.Cu)
- Thermische Reliefs für Pads
- **Stitch-Vias** nahe kritischen Bauteilen
  (Entkopplung · Quarze · Spannungsregler)
- Nach jeder Änderung: `B` (Zonen auffüllen) → erst dann DRC

PRÜFEN & EXPORTIEREN

- **DRC - nie vergessen!** Nach dem Layout + **nochmal vor dem Export**
- **3D-Viewer**: kollidieren Gehäuse & Stecker?
  - Hilft aber auch nur wenn die 3D-Modelle korrekt sind
- Export: Gerber X2 + Bohrdaten → ZIP → Fertigung

DRC



16 Warnungen - jede anschauen und verstehen, nicht sofort wegklicken

SECTION

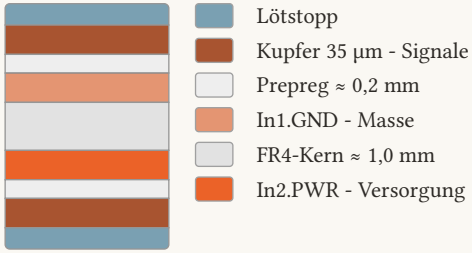
03

4-Layer

Stackup, Masse & Versorgung

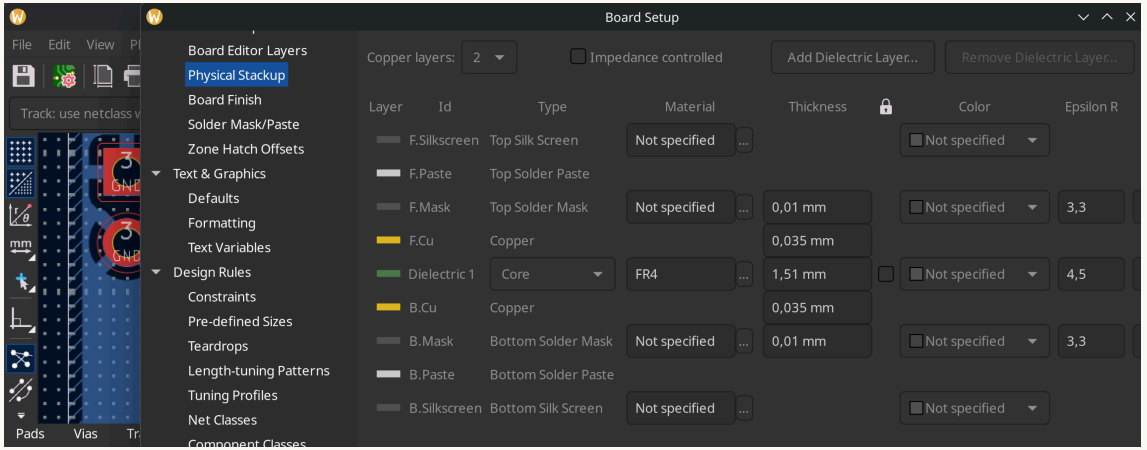
2 ODER 4 LAGEN?

- **4 Lagen als Standard** - Aufpreis klein (oder nicht vorhanden), Nutzen groß
- Massefläche unter jedem Signal → weniger Abstrahlung
- Impedanzkontrolle (USB · Ethernet · Funk)
  - Tools in Kicad!
- Versorgung als eigene Lage
- 2 Lagen: nur Adapter- & Breakout-Boards
- Billigvariante des Stackups reicht fast immer



ADVANCED: STACKUP IN KICAD EINRICHTEN

- Board-Setup ⟶ **Stapelaufbau**: Innenlagen anlegen, benennen, Dicken eintragen
- **Impedanz** im selben Dialog: Ziel 50 Ω ⟶ empfohlene Bahnbreite
- Netzklassen & Abstände wie gehabt



Physical Stackup: hier über “Copper layers” auf 4 Kupferlagen erhöhen

SECTION

04

Praxis-Fragen

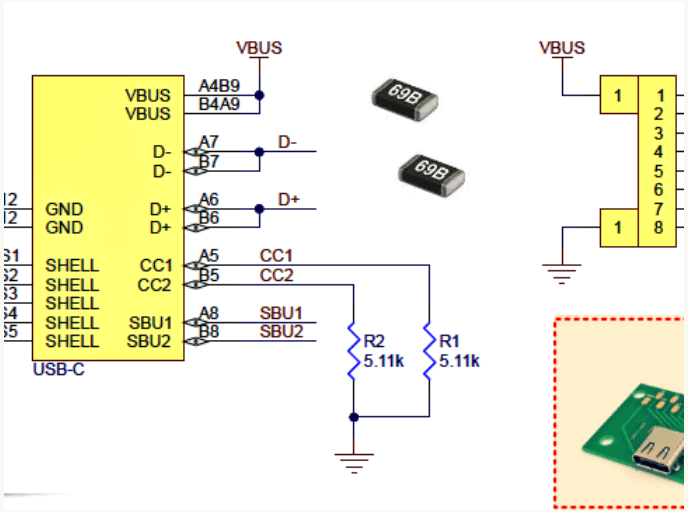
Erfahrungswerte fürs erste Board

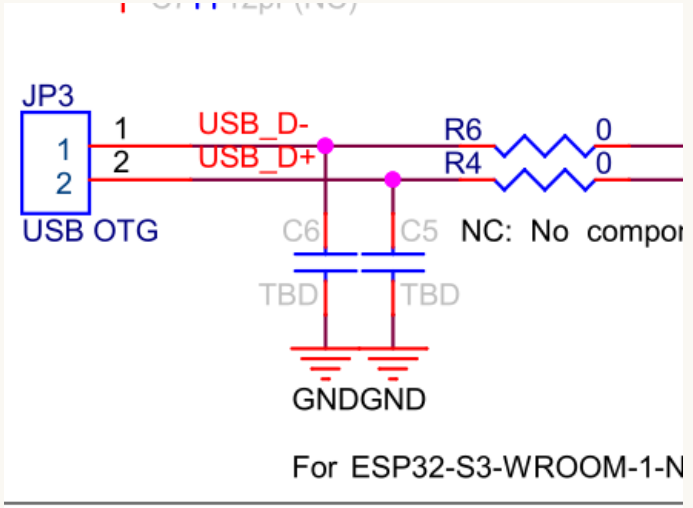
WIE STARTE ICH?

- **Klein anfangen**: eine Platine, ein Ziel
  (z. B. Versorgung + MCU-Modul - das Vortrags-Beispiel)
- **Bewährtes kopieren**: Referenzdesigns · Open-Source-Boards
- Review von Bekannten anfragen
  - z.B. Mitglieder im Chaos-Verein
- Revision 1 = Lernprojekt: **2-3 Anläufe** sind normal
- Beispiel Busanzeiger: Versorgung · Mikrocontroller · Display-Anbindung

USB-C & 5 V - EIN GUTES KONZEPT?

- **Ja - guter Standard**: robust · überall verfügbar · nur 5 V
- **CC1 & CC2: je 5,1 kΩ nach GND** - sonst kein Strom
- A/B-Pins **gespiegelt, nicht gedreht** - Footprint prüfen
- USB-2: A6↔B6 (D+) · A7↔B7 (D-)
- VBUS: **NIE zurückeinspeisen**
  - Schützen mit Verpoldiode + ESD-Netzwerk





EMV ALS ANFÄNGER?

- **Bewusste EMV-Optimierung (Funkstörungen): zu früh**
- Grundset bringt schon den Großteil:
  - durchgehende **GND-Fläche** (deshalb 4 Layer)
  - **Entkopplungs-Cs** an jeden Versorgungspin
- Einplanen: **2-3 Revisionen** + **Yellow-Wire** für Rev. 1

WIE FLEXIBEL DESIGNEN?

- Jumper **Ja** · Bestückoptionen des gleichen Teils? **Eher nicht**
  - aber z.B. optionaler pulldown Widerstand: **Ja**
- Die nächste Platine ist schnell & günstig
- Flexibilität kostet: Platz · Zeit · Nerven
- **Ausnahme: ab 100+ Stück** → Varianten lohnen

RETTER IN DER NOT: 0 Ω & 1 R

- **0 Ω in Serie** zu Strom- & heiklen Signalleitungen
  - im Notfall: austauschen · überbrücken
- Gehäuse: **0805 / 0603 - nicht kleiner** (handlötbar)
- **1 R in 0805 als Sicherung** - geht durch statt abzufackeln

SECTION

05

Bestückung & Equipment

Löten, Werkzeug & PCBA

BESTÜCKUNG: KLEIN NACH GROSS

- **Klein nach groß** - sonst kommt der Lötkolben nicht mehr dran
- **Erst SMD, dann THT** - Stecker blockieren sonst SMD-Plätze
- **Zoom nicht übertreiben**: Bildschirm ≠ Realität
- SMD in einem Rutsch: Lötpaste → setzen → Hotplate / Heißluft
- Bestückungsdruck mit Bauteilnamen?
  - Lohnt sich so wirklich - erst ab 1 000 Stück
- HTML-Bom-Viewer: Ein Traum-Werkzeug zum Bestücken
  - mit Abhakfunktion und Bauteilfinder
- PCBA (Bestückt von Fabrik) möglich, aber oft nicht nötig

EQUIPMENT - WAS LOHNT SICH?

Kaufen / Vereinsmaterial

- **Lötkolben** - temperaturgeregelt, feine Spitzen, aber auch eine fette Spitze für Steckverbinder
- **Lötpaste & Lötzinn**
- **Lupe**
- **Heißluftfön** - Rework · QFN/DFN

später bis GANZ SPÄTER

- **Bauteilordnung** - Schrank / Kästchen / …
- **Hotplate** - SMD-Lötpaste im kleinen Reflow
- **Reflow-Ofen** eher nicht
- **Automatischer Picker** - Pinzette tut's
- Großes **Mikroskop**

SECTION

06

Prüfung & Fertigung

DRC, Gerber, Bohrdaten & Kosten

ENDKONTROLLE

- **Innenlagen sichtbar machen**: GND wirklich geschlossen?
- **DRC** mit den Regeln des Herstellers (Clearance · Annular Ring · Bohrung)
- **3D-Check** + STEP-Export (`Datei → Exportieren → STEP`)
  - kann helfen, muss aber nicht, wenn die 3D-Modelle nicht genau sind
  - Hilft dann, um zu sehen ob alle Anschlüsse erreichbar sind
- Faustregel: erst Zonen auffüllen, dann DRC

GERBER & BOHRDATEN

- `Fertigungsoutputs → Plotten...`: **Gerber X2**
  - Kupfer · Lötstopp · Bestückungsdruck · Umriss
- Bohrdaten (Excellon) separat ⟶ beides ins ZIP
- Kontrolle im Webviewer beim bestellen

Alternativ

- direkt das Kicad-Projekt hochladen
- oder: Plugins für Fertiger verwenden

WAS KOSTET EINE PLATINE?

- Preis-Treiber: **Fläche · Lagenzahl · Mindestabstände**
- Kleine Platine (50x50) ungefähr

Stückzahl	Richtwert (kleine 4-Layer-Platine, China)
1-5	ca. 10-25 € - fast nur Grundgebühr, Panel & Versand
10	kaum teurer als 5 - Setup einmal bezahlt
100	ca. 1-2 € pro Stück
1000	ca. 0,3-0,8 € pro Stück

- **Kleinmengen = Fixkosten**: Setup · Panel · Versand

CHINA ODER EU?

China - JLCPCB, PCBWay, SEEED

- sehr günstig · riesige Auswahl
- Versand 1-2 Wochen teuer
- Versand 3-4 Wochen billiger
- lange Transportwege
- und: Import / Zoll
- Nicht automatisch billig für Prototyping

EU - AISLER, Beta layout, Würth

- schnell & unkompliziert
- Preisunterschied kaum vorhanden (bei ähnlicher Dringlichkeit)
- deutschsprachiger Support · Gewährleistung
- kurze Wege → besser für Umwelt & Klima
- Prototyping

- China: günstige Stückzahl · EU: Zeit, Service, CO_2, Prototyping

QUELLEN & LINKS

- `docs.kicad.org` - Handbuch & Getting Started (de/en)
- `kicad.org` - Downloads · Release Notes · Blog
- `forum.kicad.info` - Community-Forum
- Design-Rules der Fertigung (JLCPCB · PCBWay · AISLER)
  - vor dem Layout ins Board-Setup
- Tools:
  - https://github.com/hulryung/kicad-lcsc-manager
  - https://github.com/dsa-t/jlc-kicad-lib-loader

Danke - Fragen?