pro² network+ Summer School 2024

Lancaster University

PCB Design W[illegible] with KiCAD

Becoming Dangerous with KiCAD

July 2024

John Vidler & Aron Eggens

Session Goals

Lancaster University

This is a ‘hybrid’ session – feel free to just listen in and work with the worksheets; and we’ll be stepping through each stage.

- Get ‘hands on’ with KiCAD with an existing design
- Basic schematic updates
- Synchronise PCB layout with schematic changes
- Implement some simple layout techniques
- Generate a Bill Of Materials (BOM) – *for the next session!*

Who are we?

Dr. John Vidler is a Senior Research Associate at Lancaster University and is working across a wide range of systems topics from AI/ML to embedded device development. His current research involves working with Psychology and Linguistics to bring devices to research scientists

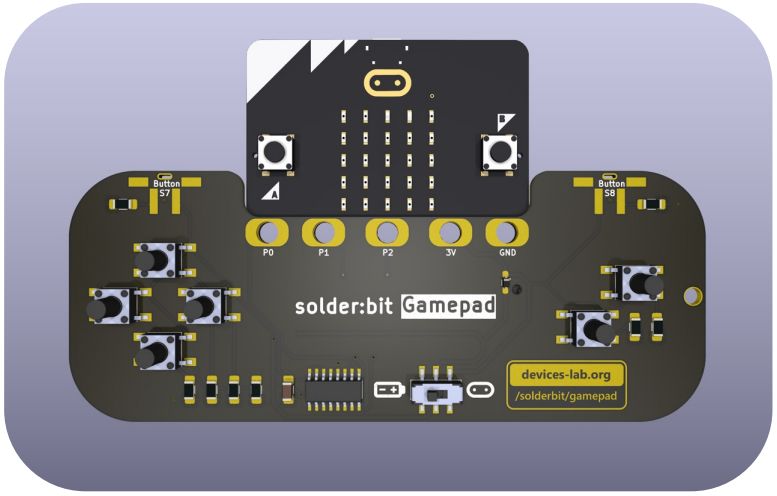
Aron Eggens is a PhD student at Lancaster University, working in digital device prototyping, consumer electronics, and embedded systems. His projects include the solder:bit initiative and Gerber manipulation tools for PCB design, aiming to make hardware creation accessible to everyone.

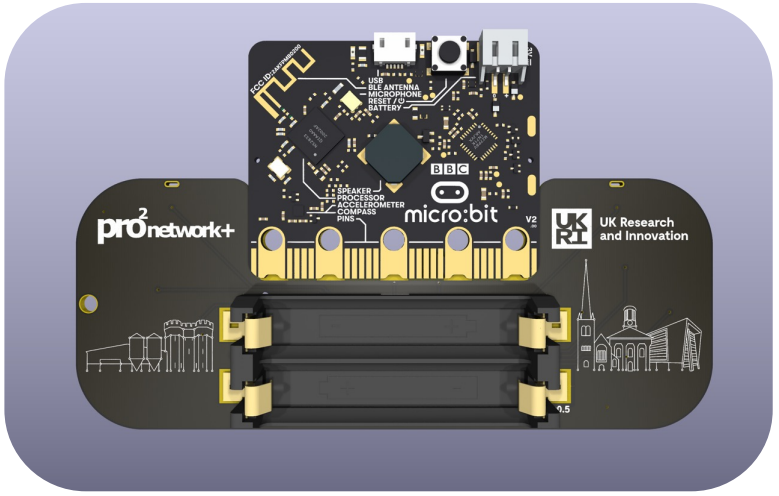
Download the project!

Start by downloading a copy of the repository containing the example project; a version of the solder:bit

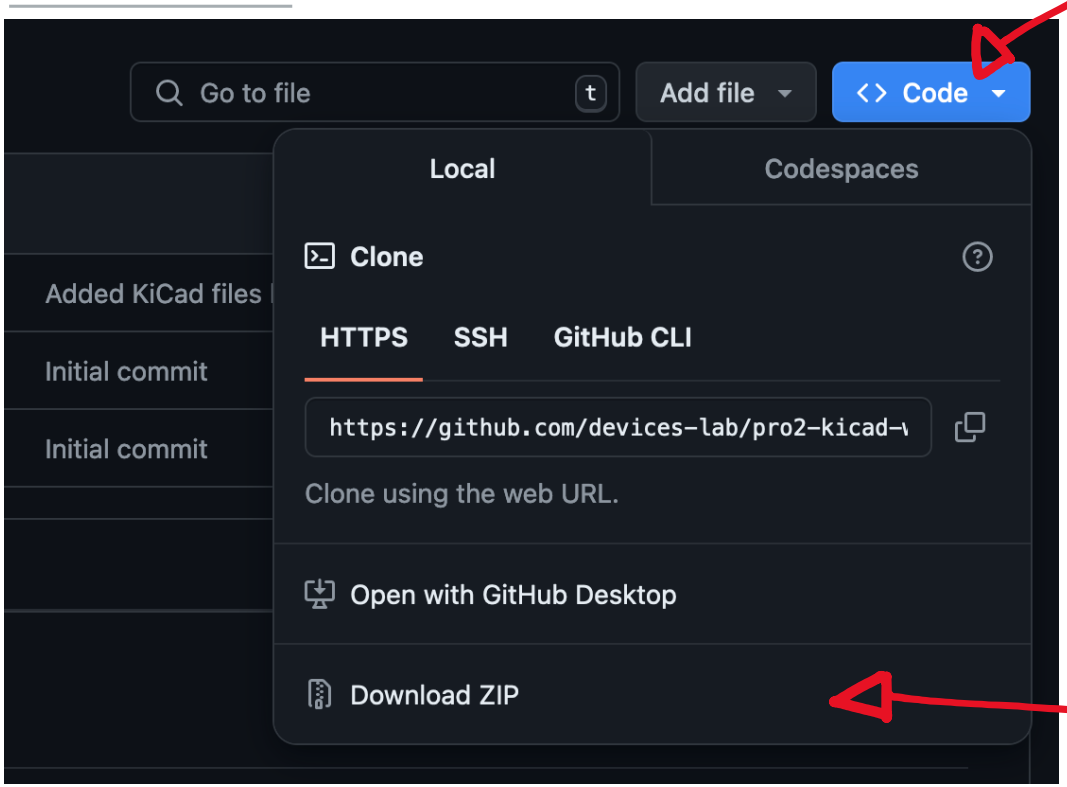
`github.com / devices-lab / pro2-kicad-workshop`

(This address is also on your worksheets)





Download the project!



To clone with git:

1. Select this button
2. Copy the address in the textbox
3. Run in a terminal:
   `git clone <address here>`

To download without git:

1. Select the ‘Download ZIP’ button
2. Extract the downloaded ZIP file into your home drive

github.com / devices-lab / pro2-kicad-workshop

Download the project!

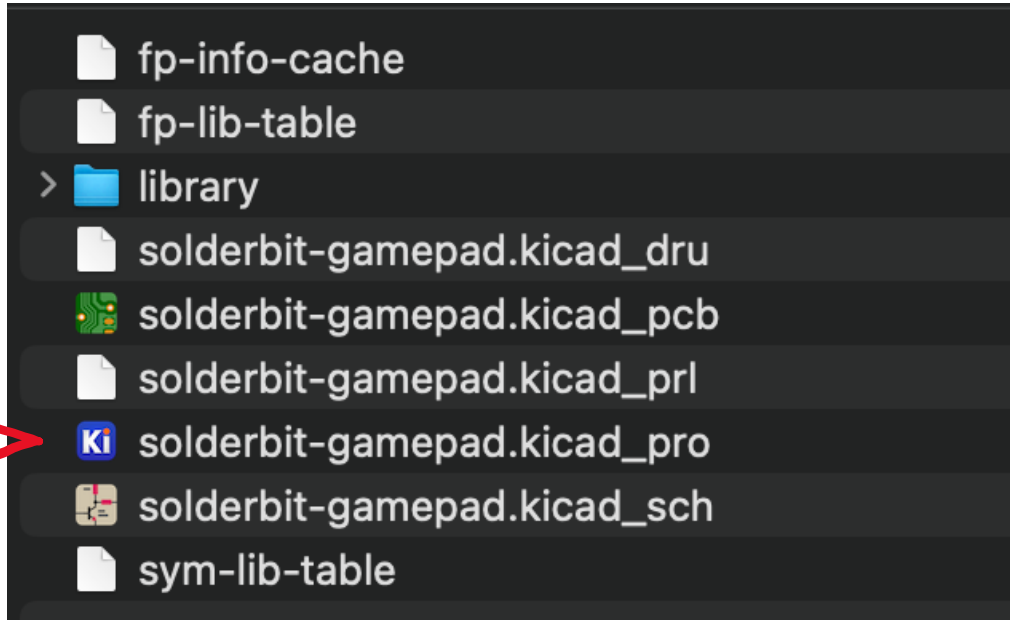
Lancaster University

We should now have a folder with approximately these files!

(Aron may have pushed another update!)

Open up the project by opening the '`.kicad_pro`' file

You can open schematic and board layout files individually, but as we need the whole project, lets stick to that this time



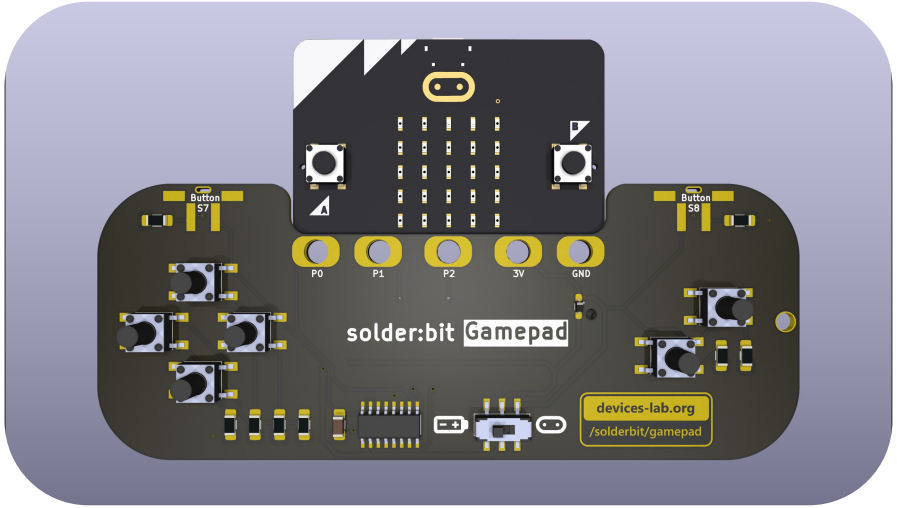
`github.com / devices-lab / pro2-kicad-workshop`

Requested Changes

Lancaster University

The story so far:

From a design review, you have determined that the 5x5 LED matrix on the Micro:bit is simply *not enough blinky lights* and we need to include some Neopixels onto the design ☺

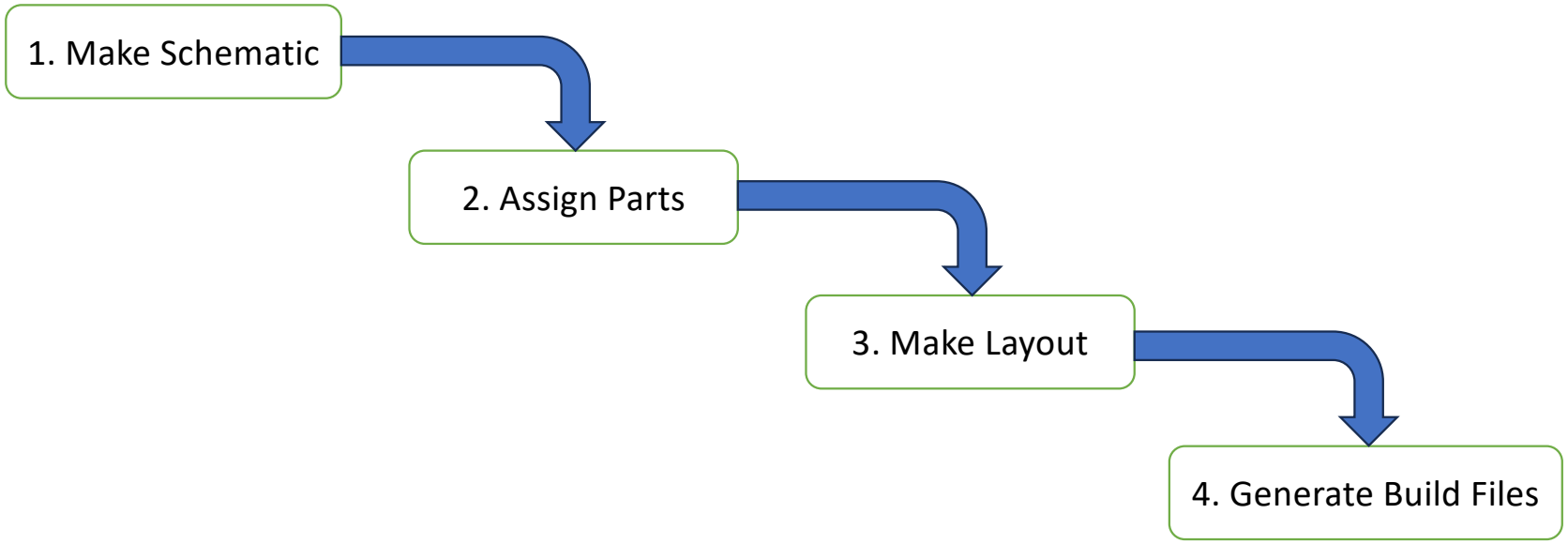


Lancaster University

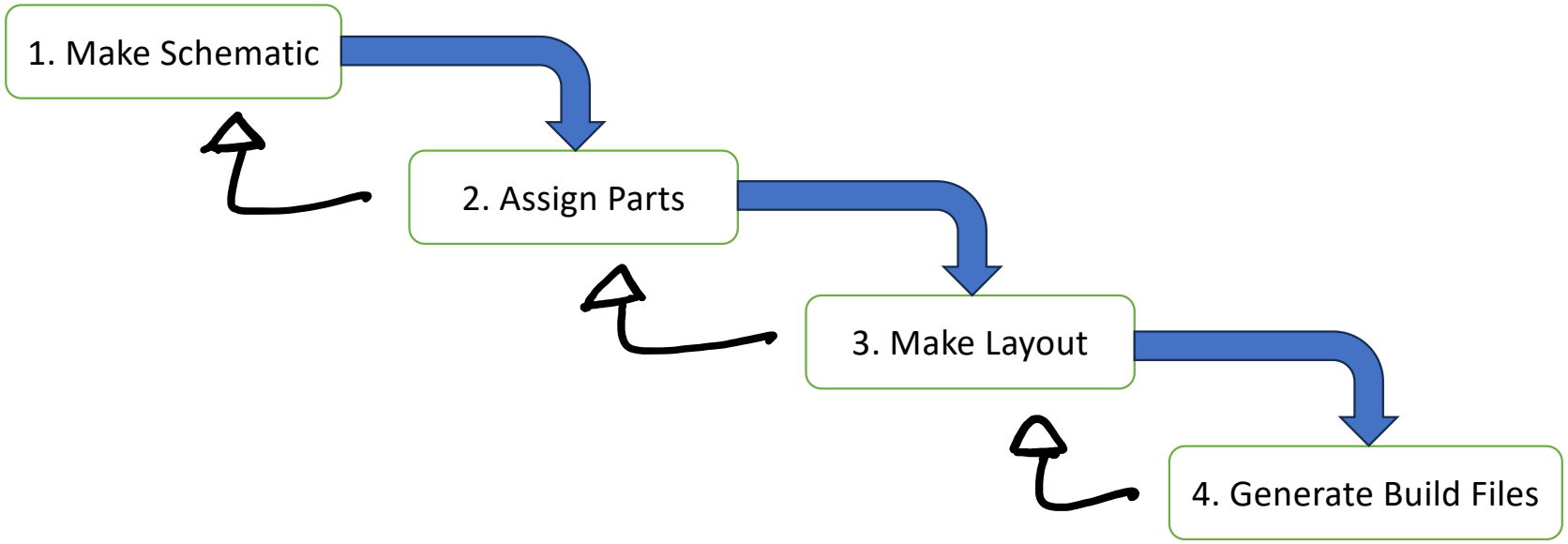
0. Viewing the Schematic and Layout

The Design Process

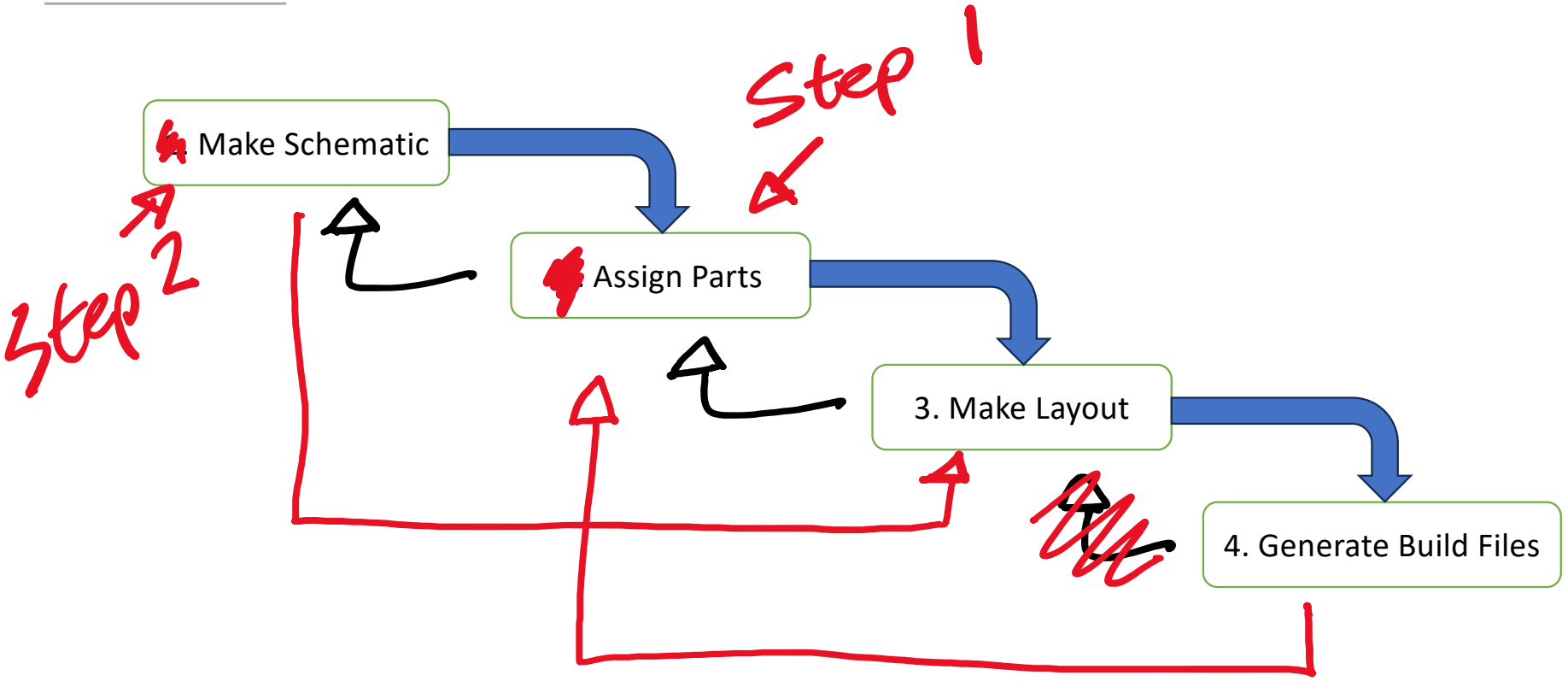
Lancaster University



The Design Process



The Design "a" Process"



Lancaster University

1. Editing Your Schematic

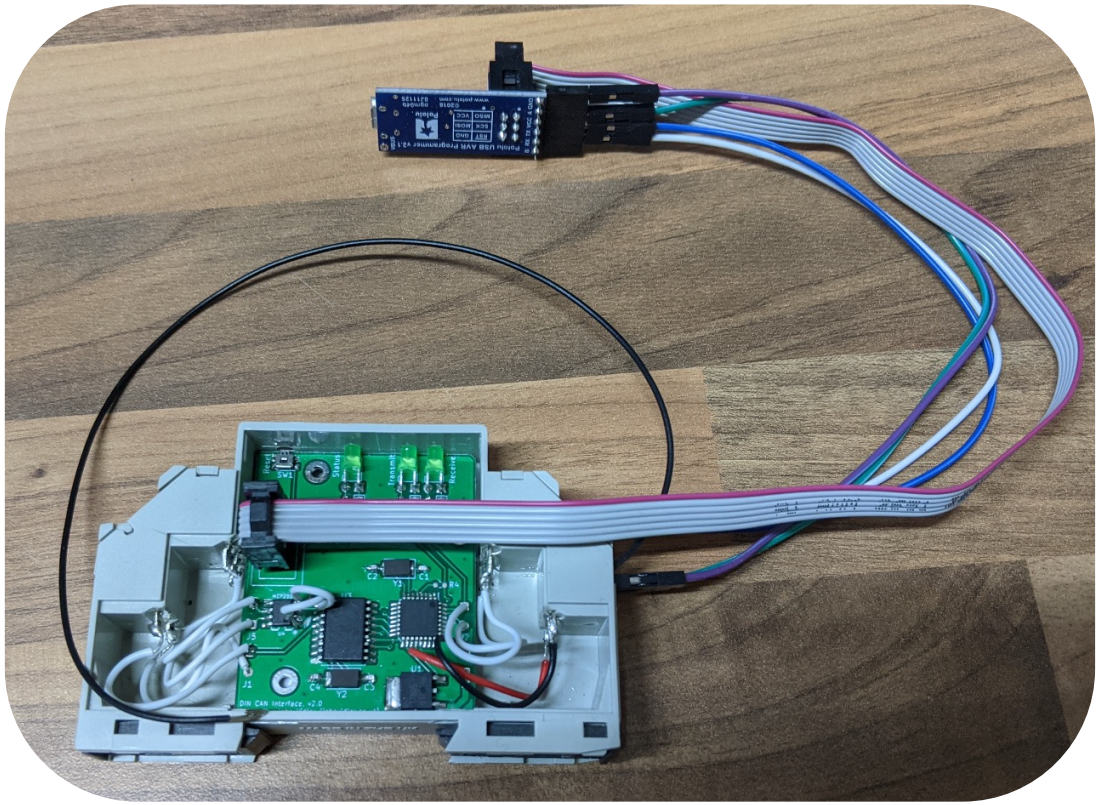
Pin Names and Functions

Lancaster University

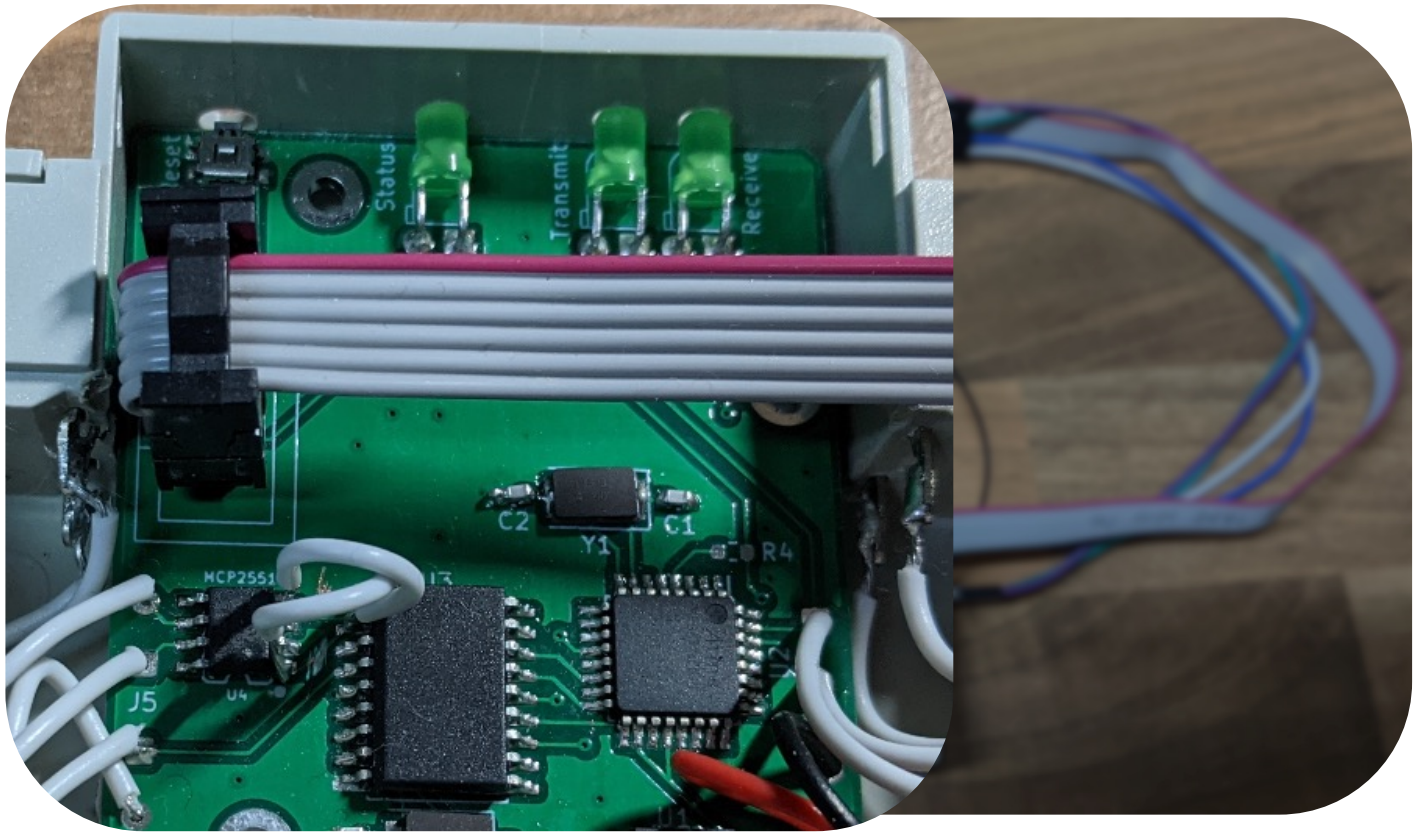
- Common ‘Gotcha’s
  - TX and RX … but which device are we talking about?
  - MISO/MOSI/SCK – better, but now we also have:
    - SOMI/SIMO, SPI_TXD/SPI_RXD, etc. etc.
  - Know your Vs - VCC, VDD, VSS, VEE
    - I prefer to use +5v, +3v3, unless the voltages are unknown (in a range)

Sometimes you just aren’t sure with new/prototype components, so make it easy to swap it later – Test pads and spacing are your friends

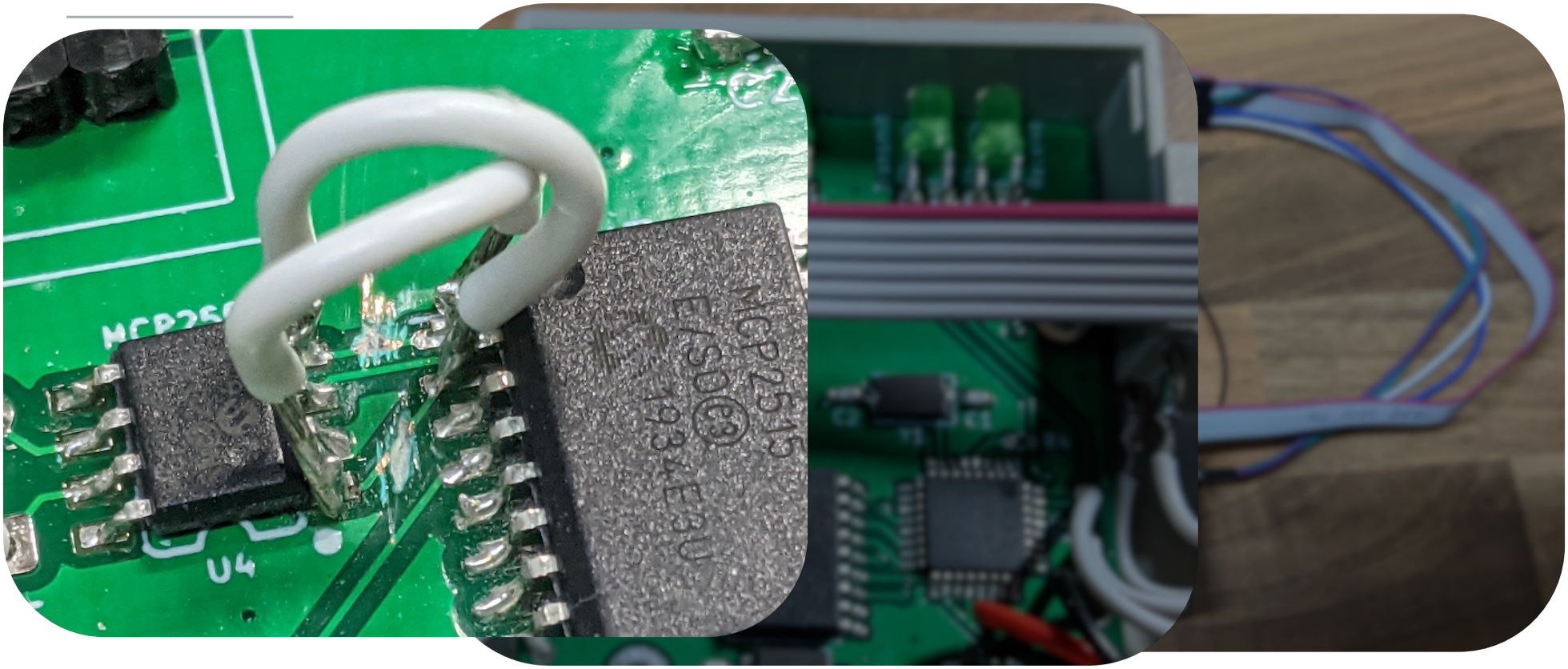
Pin Names and Functions



Pin Names and Functions



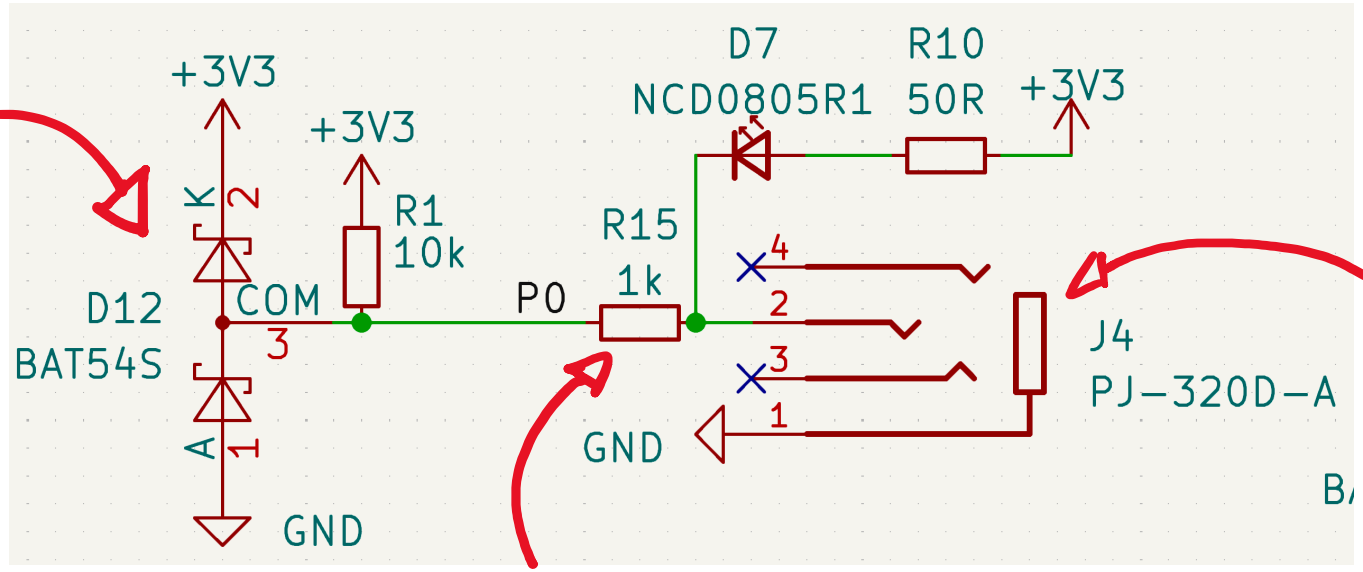
Pin Names and Functions



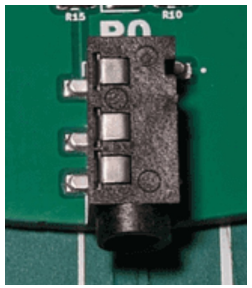
Protect and Serve

TVS diodes and Decoupling Capacitors

TVS Diode Pair



Jack Socket



Current Limiting Resistor

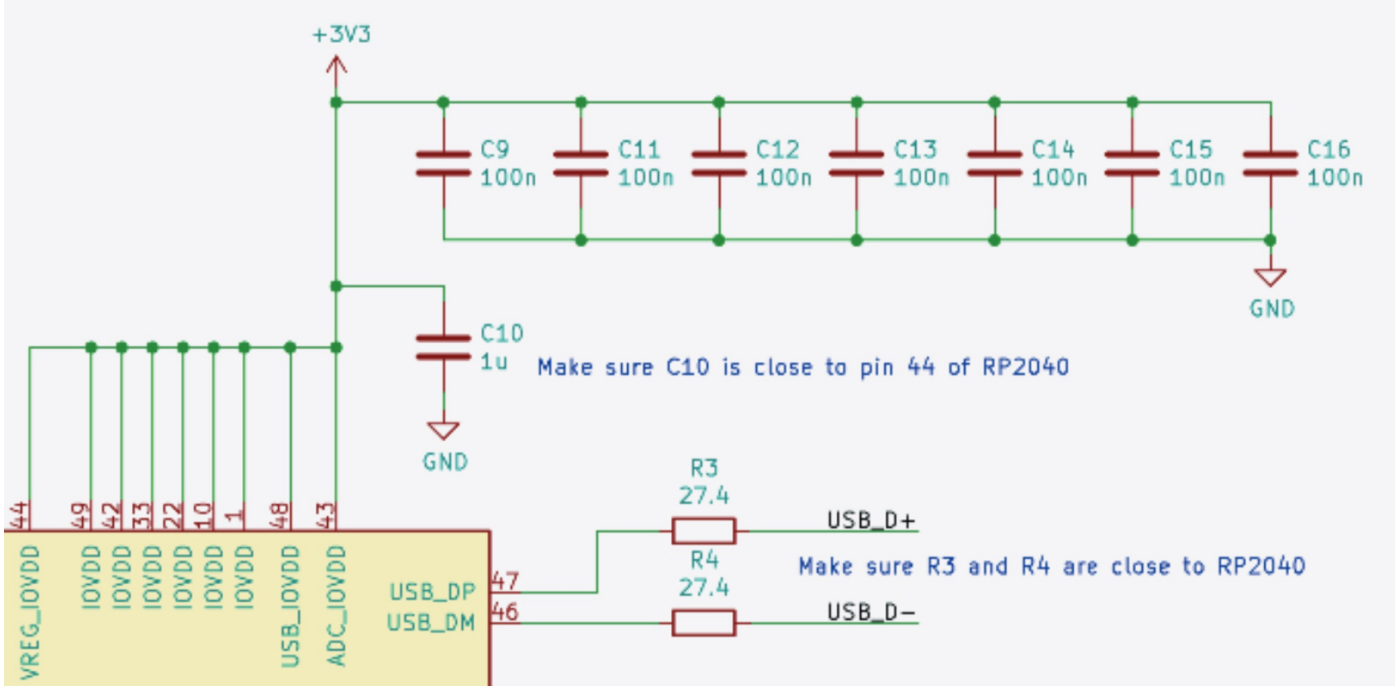
Protect and Serve

TVS diodes and Decoupling Capacitors

Sometimes the schematic can be misleading

The capacitors C9 to C16 are decoupling the IOVDD supply pins, but look like they're just a big parallel bank

In reality these would each be placed close to an individual IOVDD pin, to locally decouple, but only experience would tell you this here.



Schematic and photo example from the Raspberry Pi RP2040 Reference Design

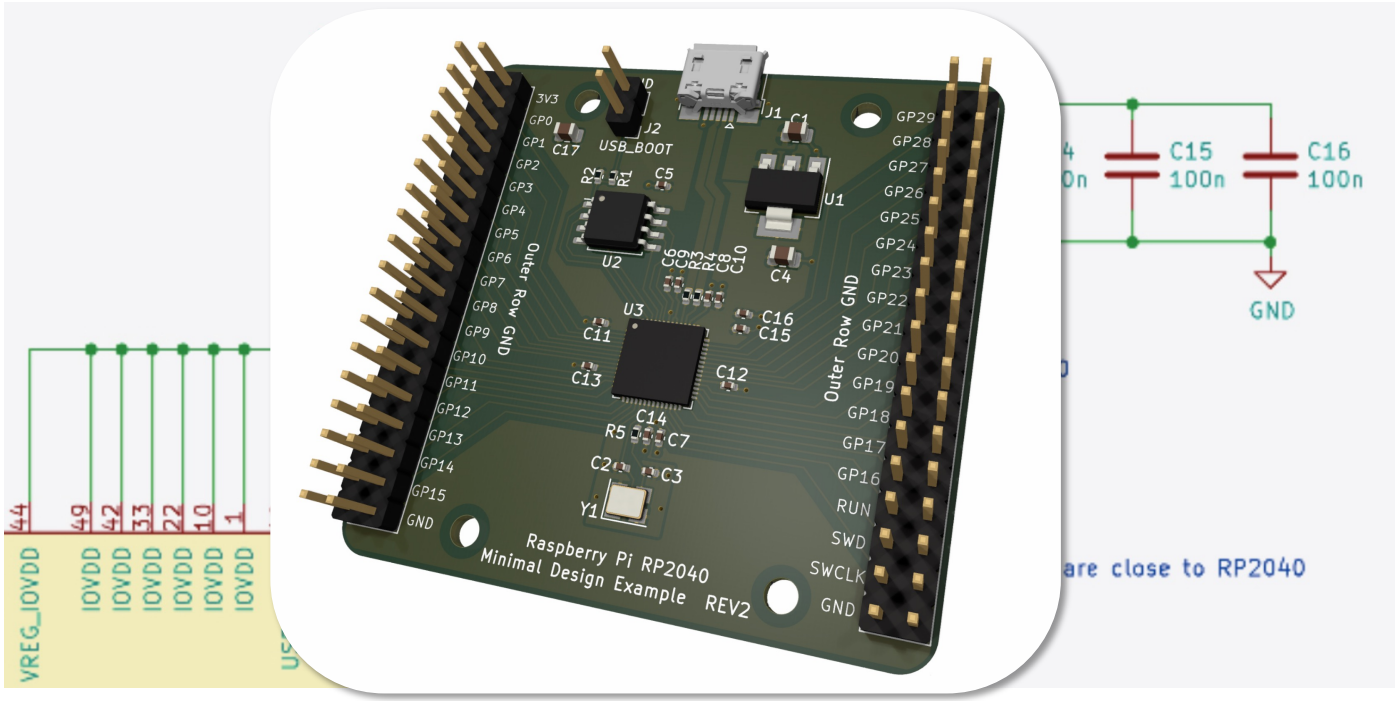
Protect and Serve

TVS diodes and Decoupling Capacitors

Sometimes the schematic can be misleading

The capacitors C9 to C16 are decoupling the IOVDD supply pins, but look like they're just a big parallel bank

In reality these would each be placed close to an individual IOVDD pin, to locally decouple, but only experience would tell you this here.



Schematic and photo example from the Raspberry Pi RP2040 Reference Design

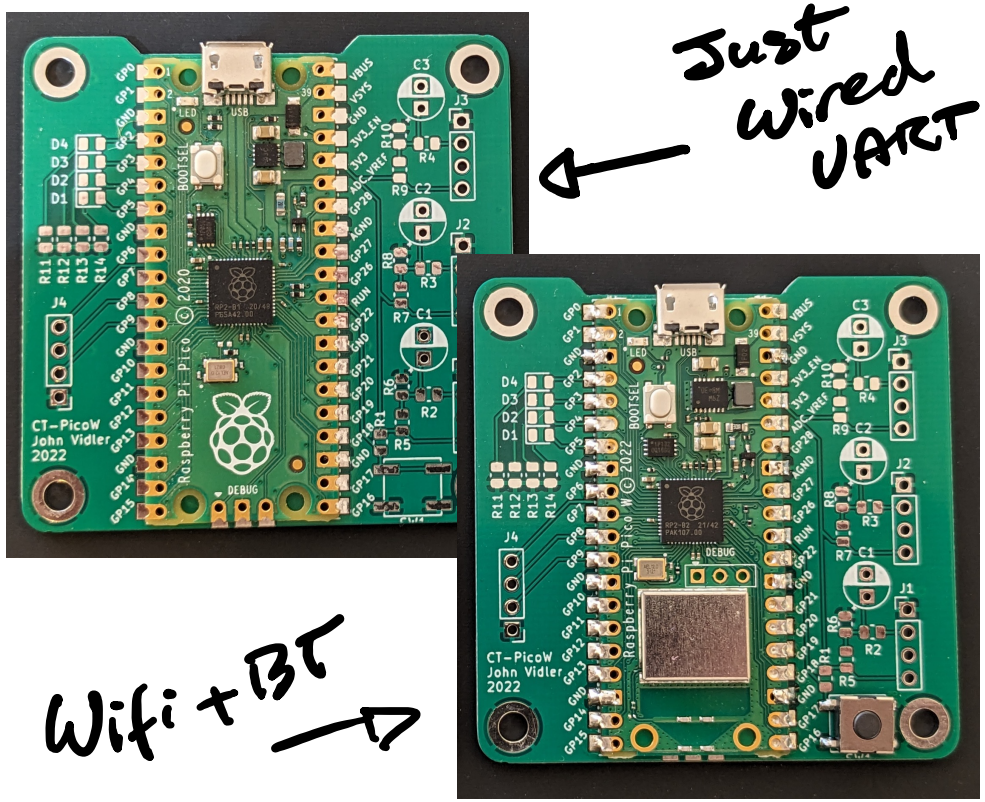
Lancaster University

2. Choose your footprints

Choosing components

Lancaster University

- Part of an iterative process!
  - Many components have numerous alternatives; try to be generic where you can so you can swap parts in and out.
    - Passive components in particular
    - Mark ones that matter for certain values!
  - Common footprints are your friend for prototypes
    - Arduino/Feather/Raspberry Pi Pico pinouts are similar enough to allow swapping (within the device family)



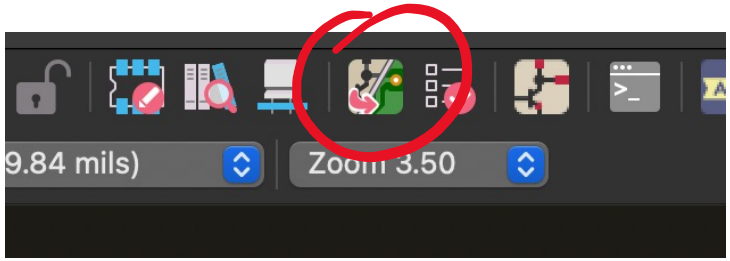
Lancaster University

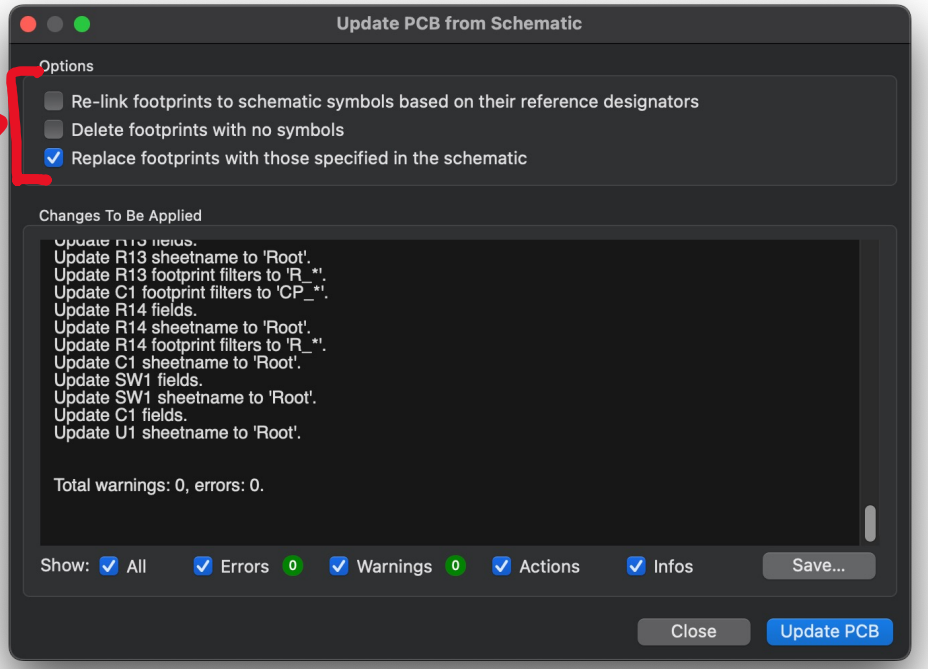
3. Synchronise your board layout

Synchronise your board layout

Lancaster University

This pulls any schematic changes into the layout editor





You may want to enable some of the extra options to ensure you don't have old parts from previous revisions hanging around

(Remember the fun of the 'process' from earlier!)

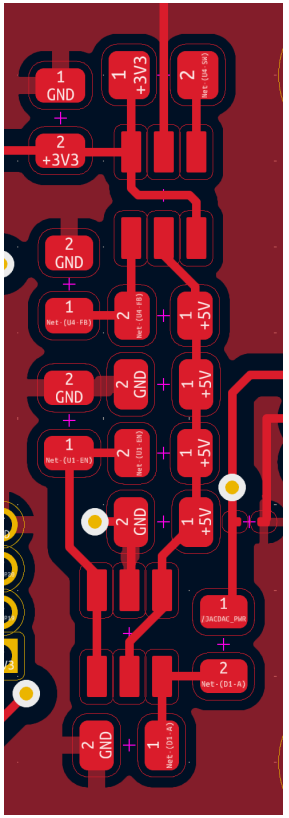
Lancaster University

4. Placing components

Placing components

Lancaster University

- Sometimes it is possible to achieve clean layouts with some combinations of parts.
  - Other times… less so...
- It can be worth playing around with some parts in isolation away from your board to see how they fit together first, then look to move them into place
  - This power supply layout took quite a few layout attempts to get ‘clean’
- But avoid *bikeshedding*[1]!
  - For prototypes, unless it’s a high-frequency, RF, or differential pair, it will *probably* work.

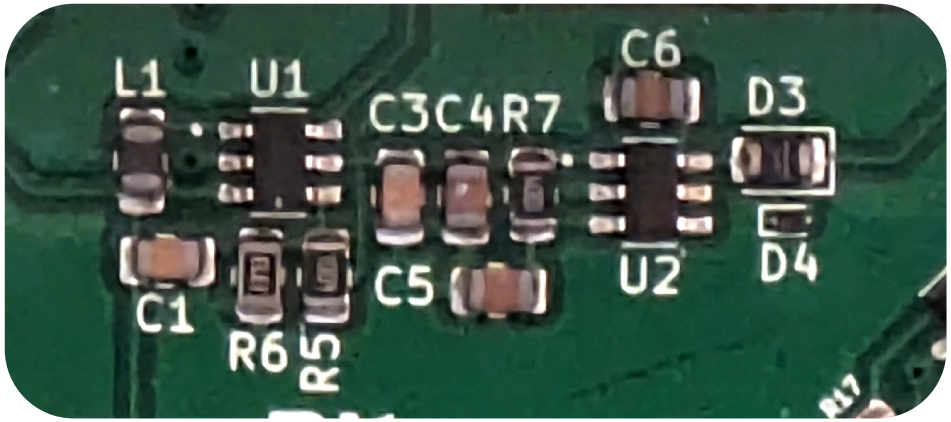


1 - https://en.wiktionary.org/wiki/bikeshedding

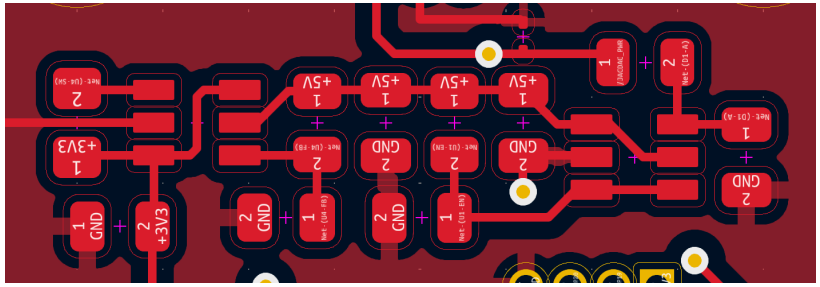
Placing components

Lancaster University

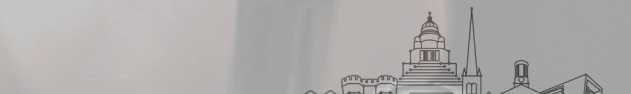
- Once you have known-good component placements, you can copy them to new designs.



- For the Access:bit I simply copied my known-good Jacdac power supply design from one schematic to another and replicated the layout ☺

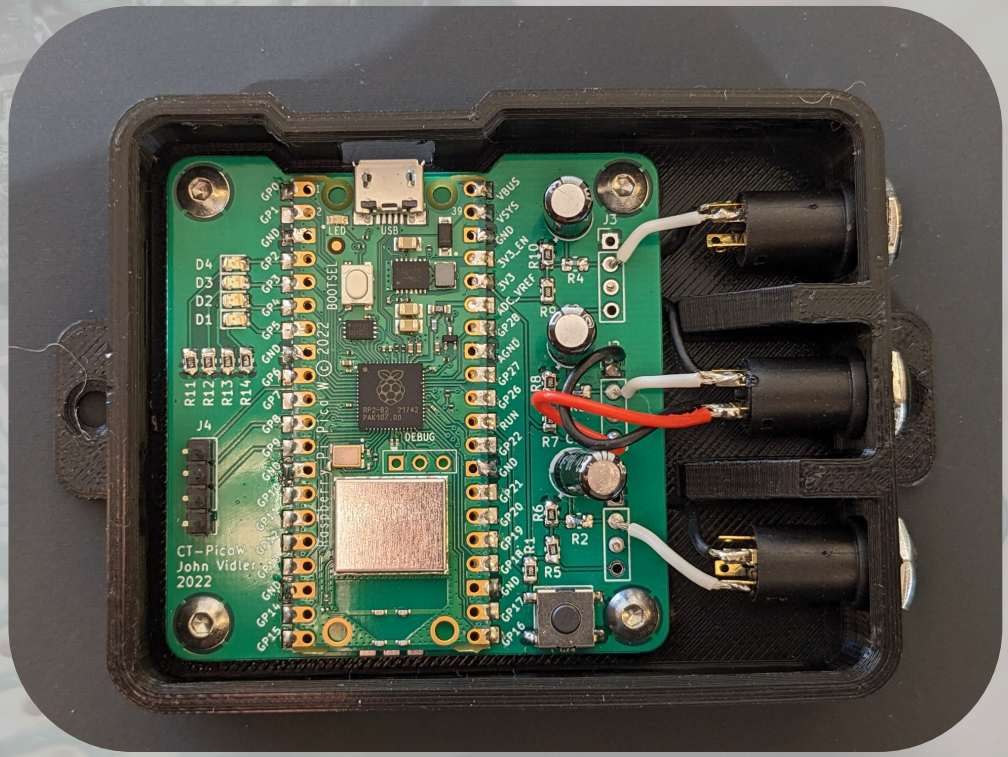


Placing Components



Lancaster University

- Sometimes the best option is to *not* place the component
- It wasn't clear where the jack connections were going to be for this board, as the case is 3D printed, so I simply dropped down 4-way 0.1" pin header footprints instead and air-wired them in place



Lancaster University

5. Generate Gerbers and BOM