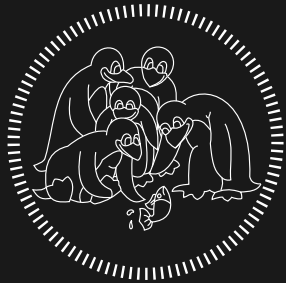




PCB Design course 2022

Introduction to PCBs

The final form of your circuits



POLITECNICO OPEN
unix LABS

Luca Biscaldi
bisca@poul.org



POLITECNICO OPEN
unix LABS

Who are we?

We are a non-profit *association* and *hacking community*
formed by students from **Politecnico di Milano**



POLITECNICO OPEN unix LABS

What do we do?


We promote the use of **Free and Open Source Software**
through courses, talks and workshops



POLITECNICO OPEN unix LABS

What do we do?

We tinker with **software and hardware** and share knowledge with anyone who is curious about technology



POLITECNICO OPEN unix LABS

What do we do?

...but also

Creative ~~Cazzeggie~~ Development

POLITECNICO OPEN unix LABS

Where are we?

Politecnico di Milano (Leonardo campus), building 9A

Come visit us!

Make sure to check **BITS** to know if the headquarters are open!



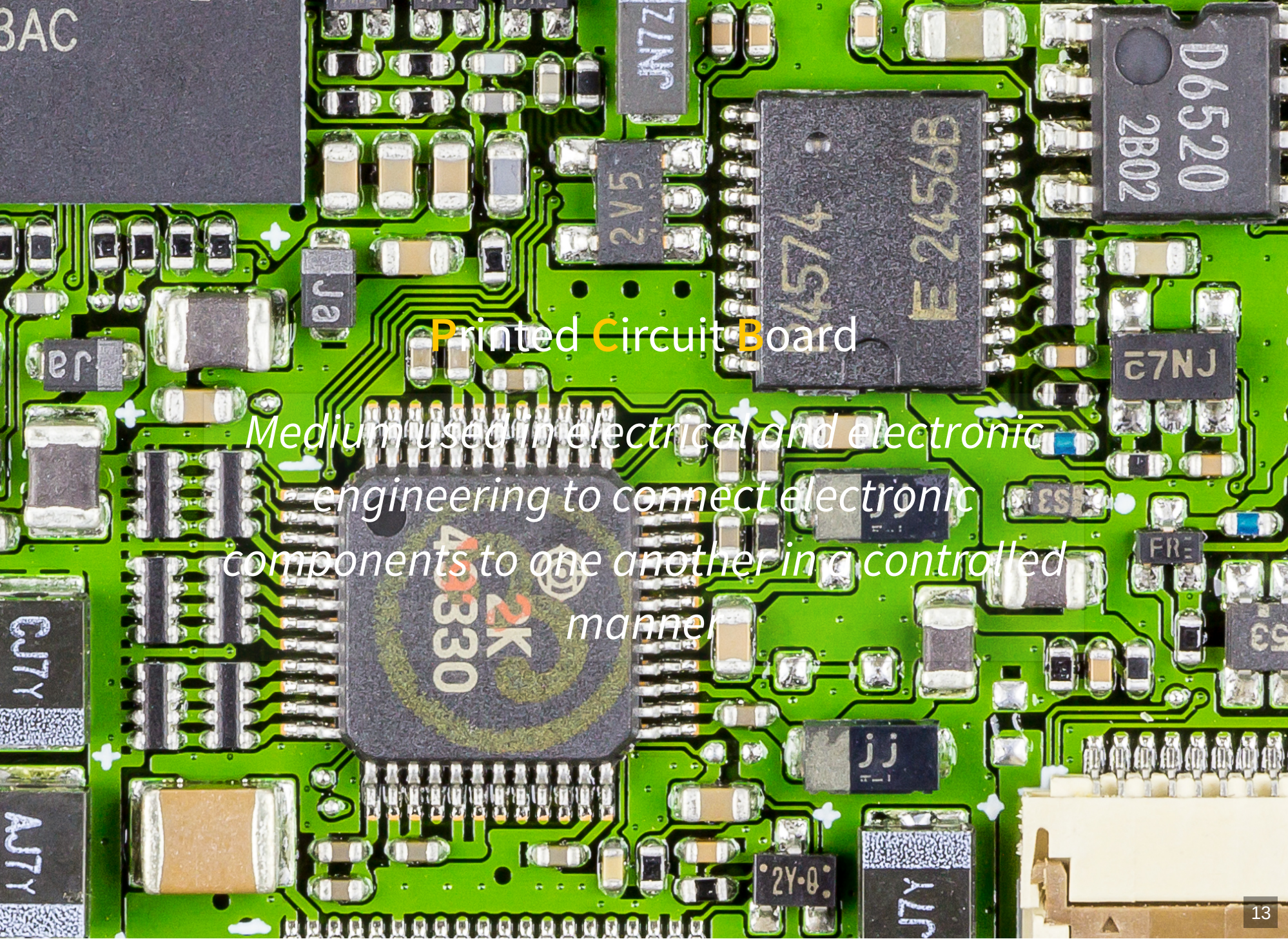
This course is...

- Suitable for every (aspiring) engineer (not necessarily EE)
- *As practical* as possible
- Oriented to hobbyists
- Complementary to PoliMi courses

This course is *NOT*...

- "How to design a circuit"
- Held by professional PCB designers
- Professional level advice
- Even remotely suitable for serious PCB design

What is a PCB?



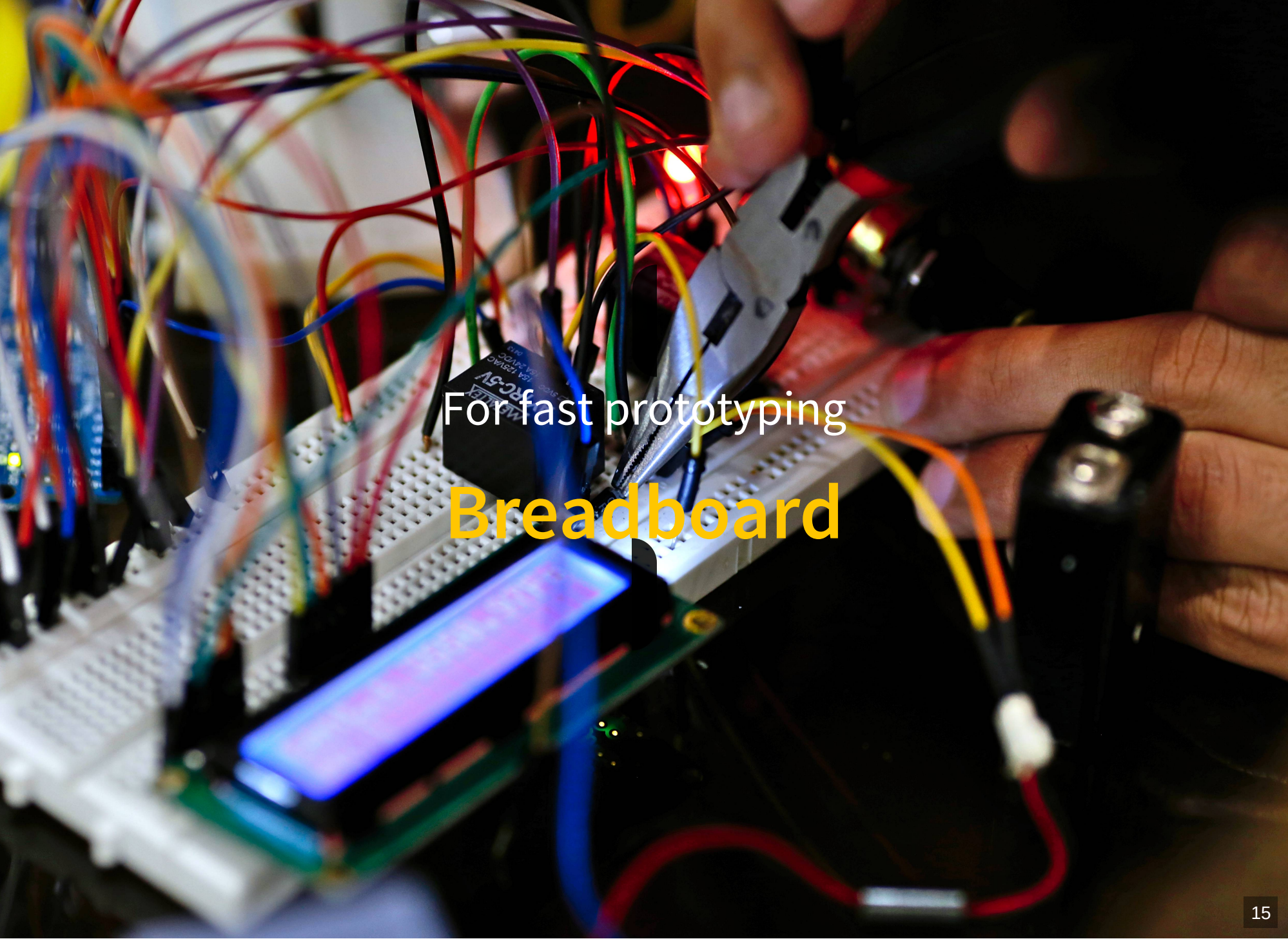
Printed Circuit Board

Medium used in electrical and electronic engineering to connect electronic components to one another in a controlled manner.



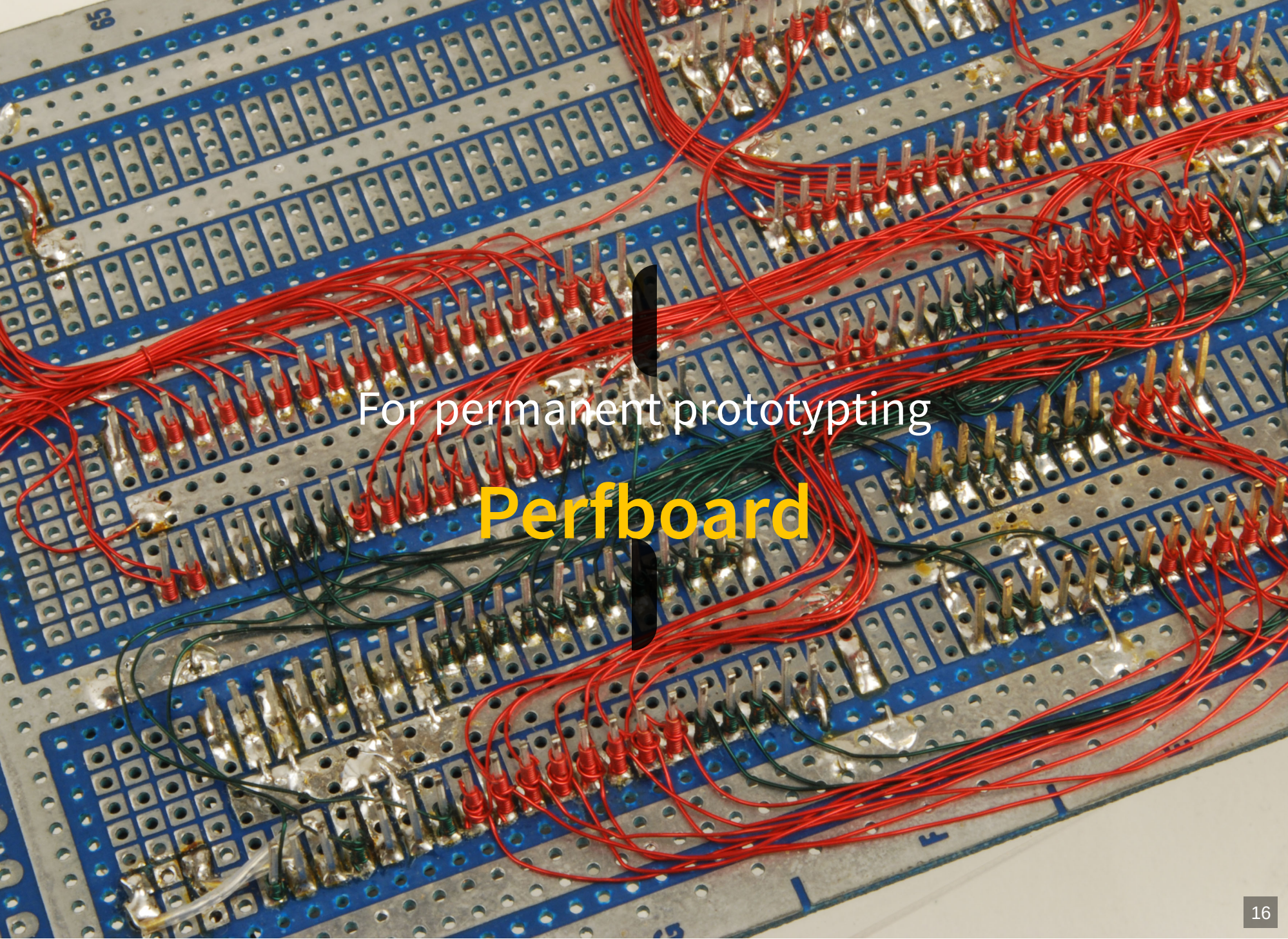
How did they do it before?

Point-to-Point connections



For fast prototyping

Breadboard



For permanent prototyping

Perfboard



For serious prototyping

Custom PCB

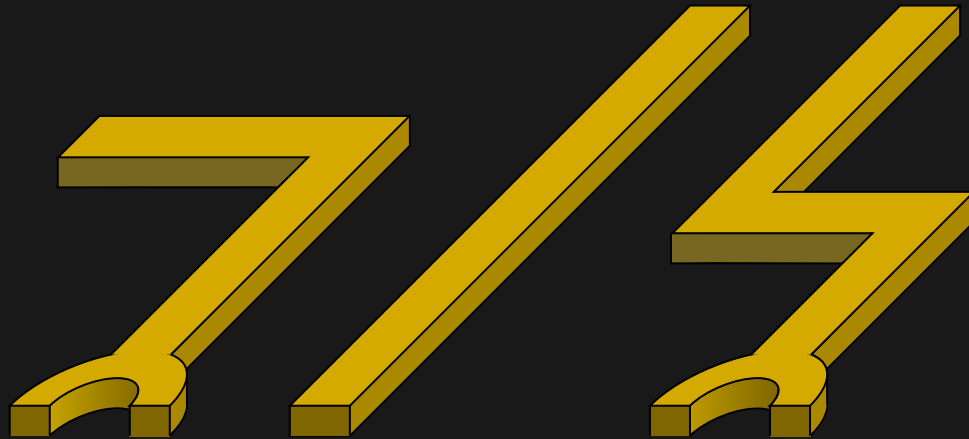
Single layer PCB

Composition ☒

Traces

Single layer PCB

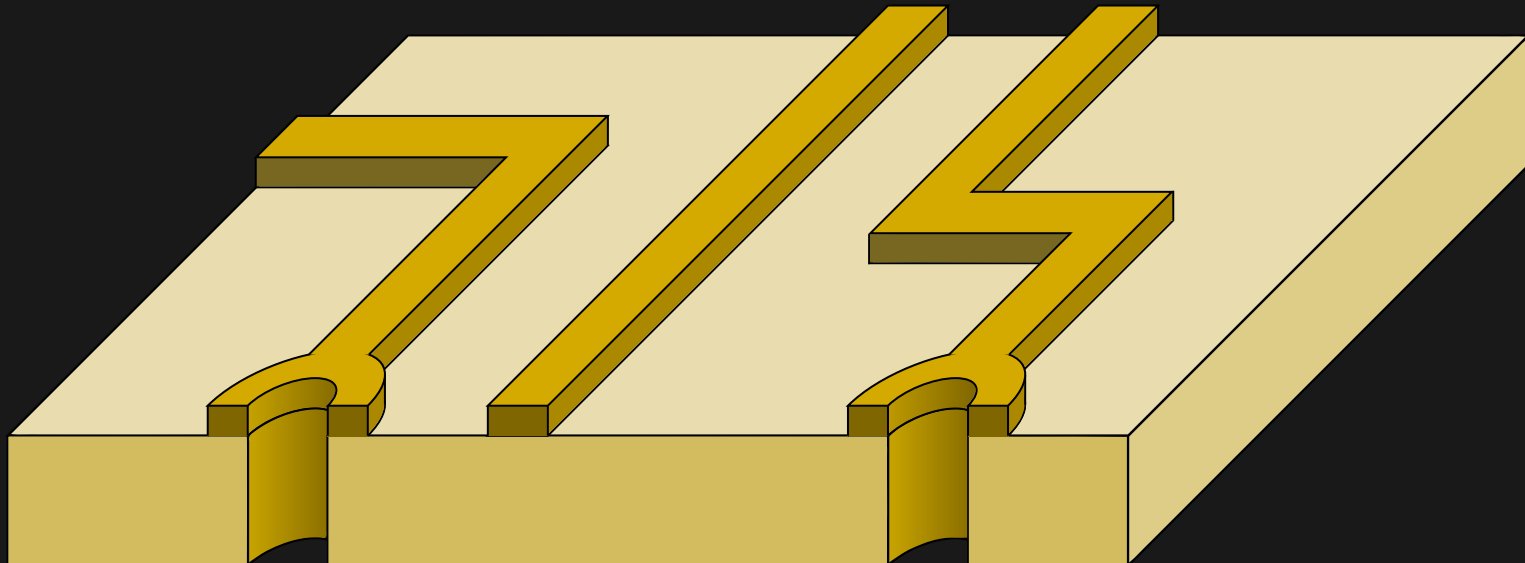
The electrical signal is carried by a thin **path of conductive material**, connecting different components.



Substrate

Single layer PCB

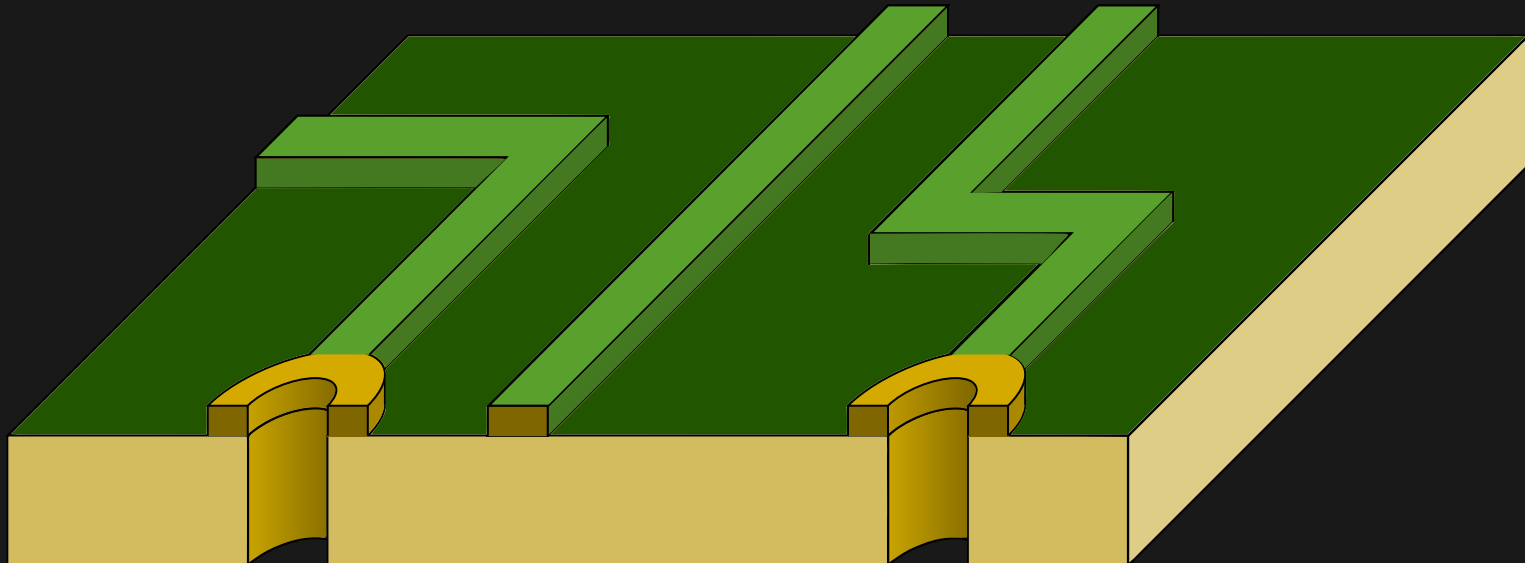
The traces sit on a **non-conductive sheet**, providing structural integrity.



Solder mask

Single layer PCB

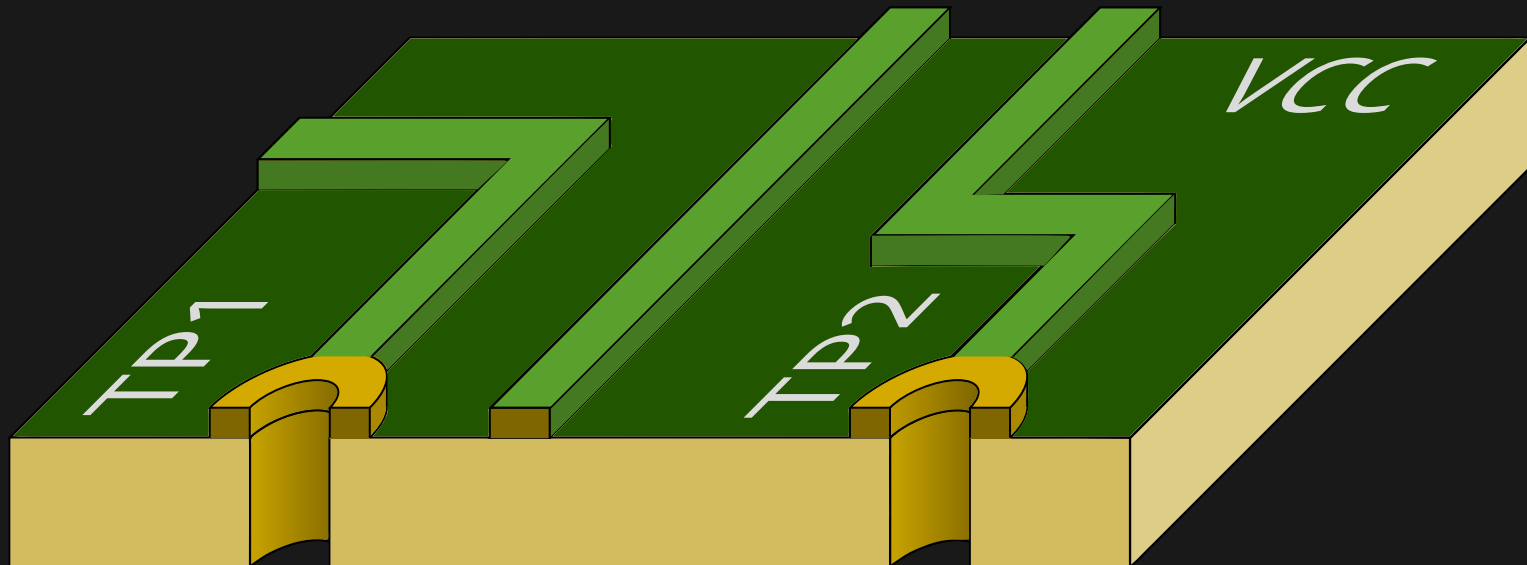
The traces are covered by a **protective layer**, preventing traces oxydation.



Silkscreen

Single layer PCB

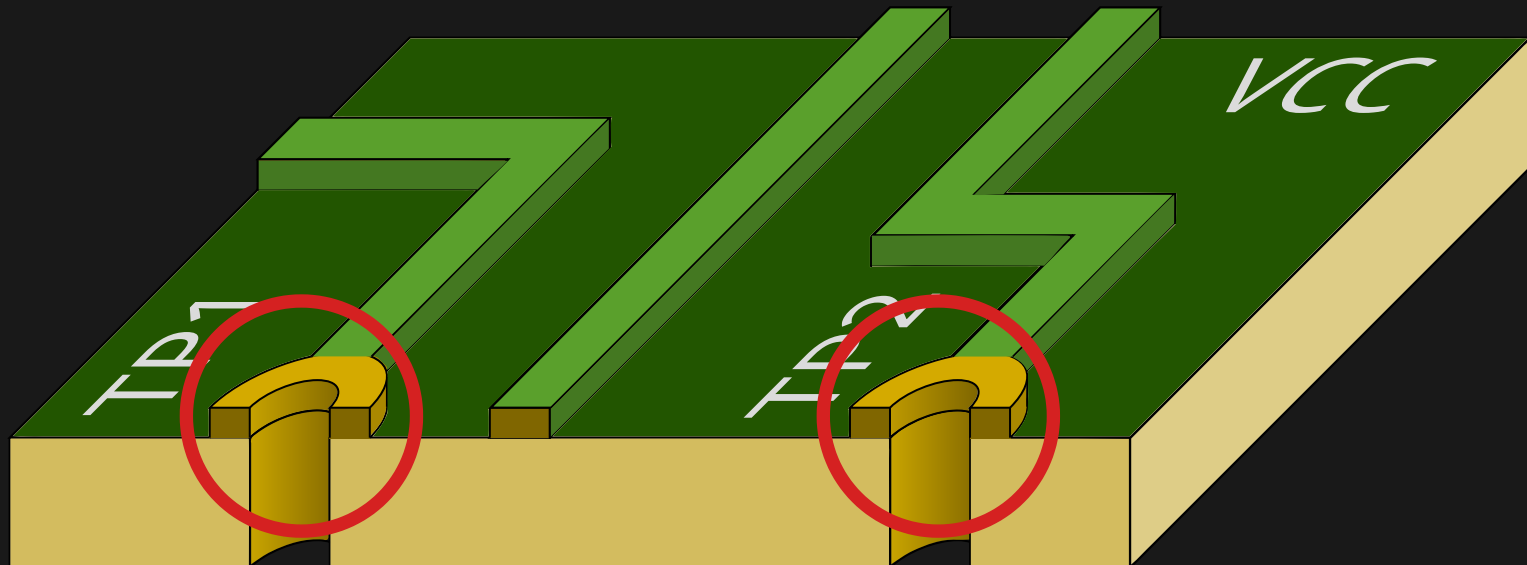
The solder mask can be covered in **ink traces**, useful to identify components or significant points in the circuit.



Pads

Single layer PCB

Electronic components are soldered to the traces in **specific points**, not covered by the solder mask



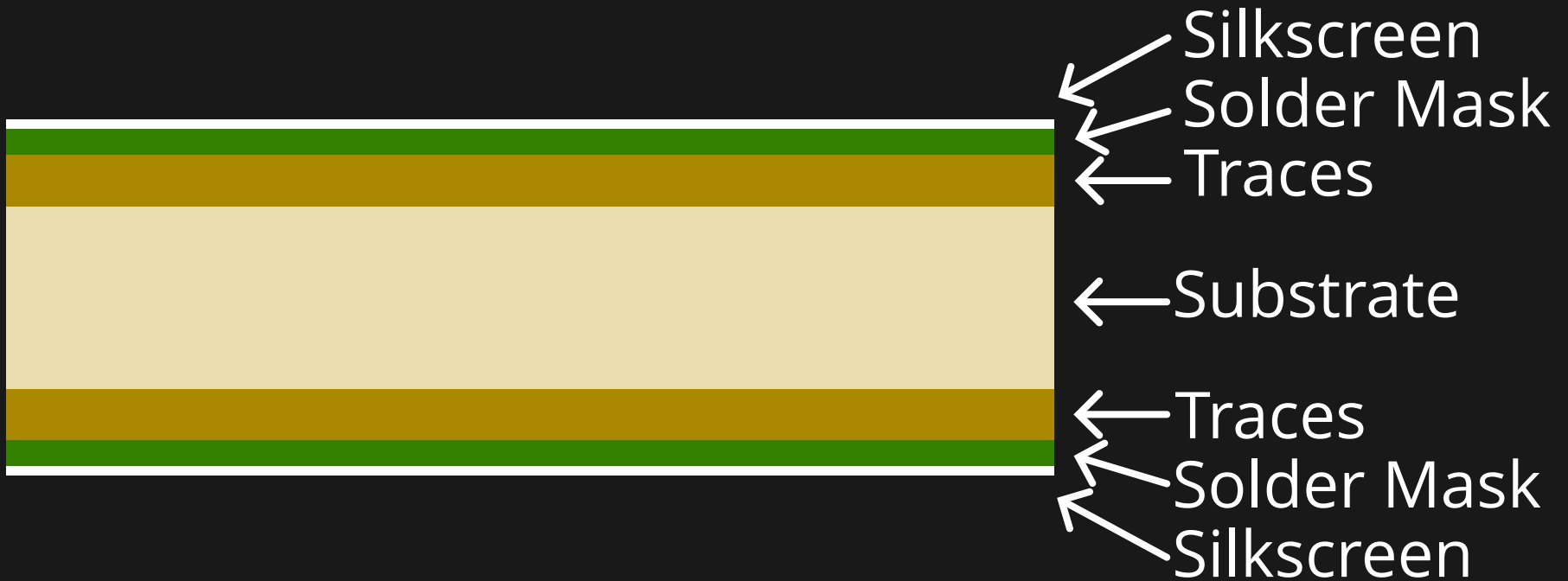
A one layer PCB is very difficult to work with! ⚠

- Determining traces **paths** is difficult even for simple designs
- The circuit is more prone to **noise interferences**

Solution:

Dual layer PCB

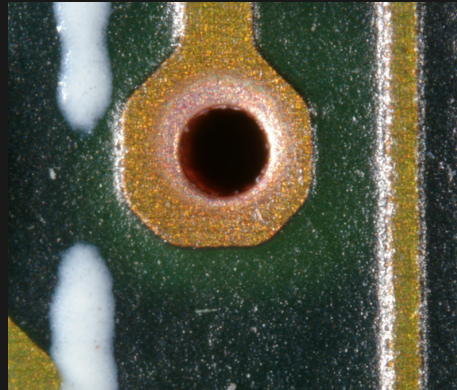
Sandwich to the rescue! ☒



How to connect the two layers?

Vias

Traces in different layers are connected by **small plated holes** going through the substrate.



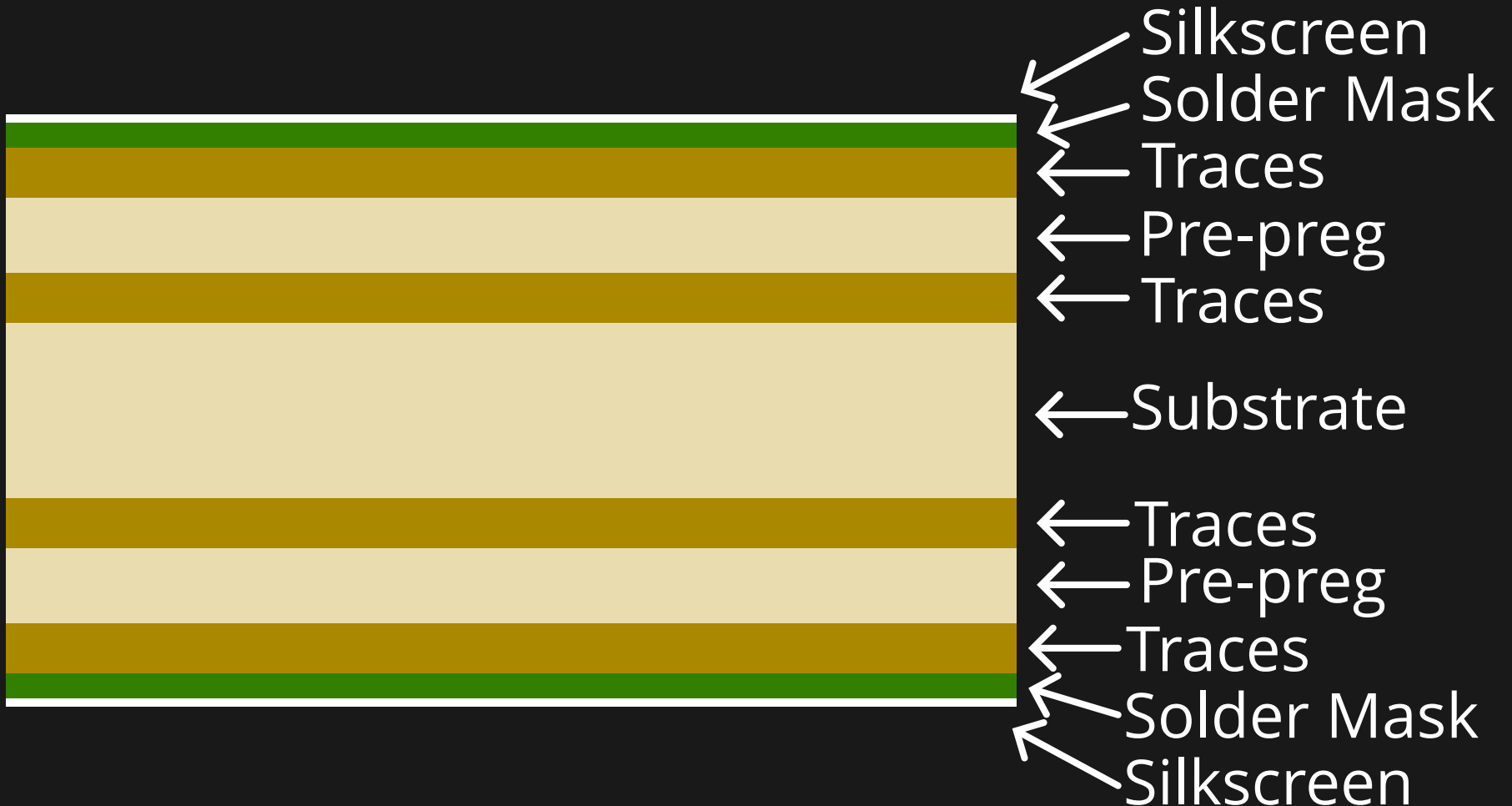
2 layers are not enough?



MORE

Multilayer PCB

The *club* sandwich ☒

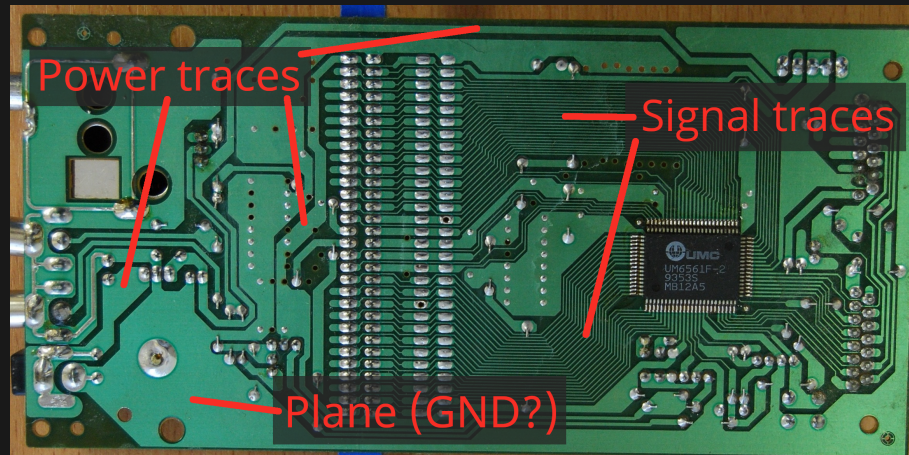


Let's get more specific ☒...

Traces

General rule of thumb to classify them:

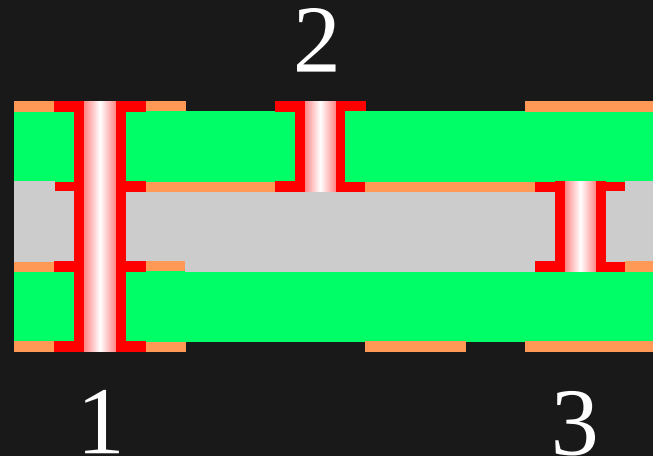
- **Narrow** traces carry signals
- **Wide** traces carry power
- **Very wide** -looking traces (almost covering the entire layer) are called **planes** (connected to ground or power)



Vias

They can be divided in 3 types:

1. **Through-hole** vias
2. **Blind** vias
3. **Buried** vias

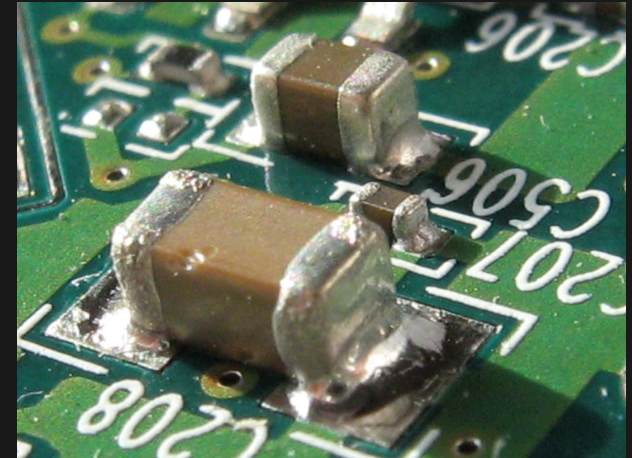


Blind and **buried** vias are difficult (and expensive!) to manufacture

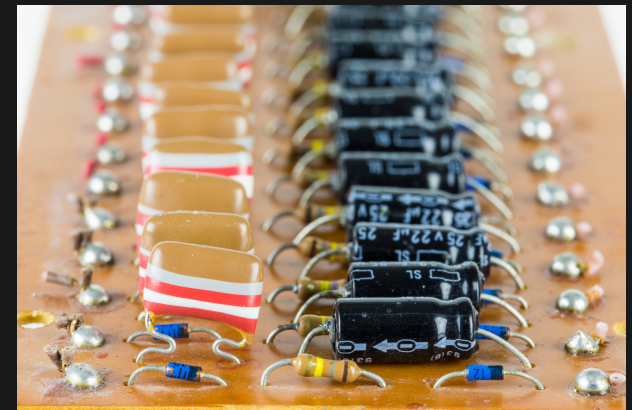
Pads

There are two types of pads:

Surface Mount Pads: component terminals are soldered to the surface of the pcb.

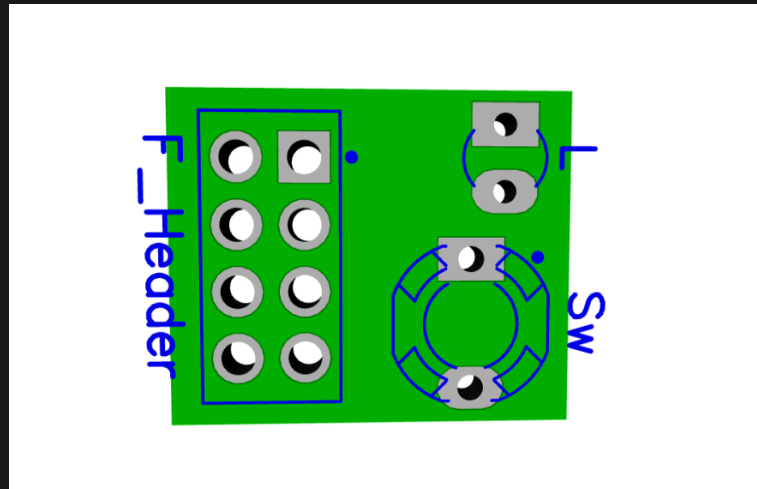


Through hole Pads: component terminals pass through plated holes in the pcb



Footprint

The arrangement of pads compatible with a specific component



They are (mostly) standardized, but there are a lot of standards.

KiCad default libraries contain over **10k** footprints!

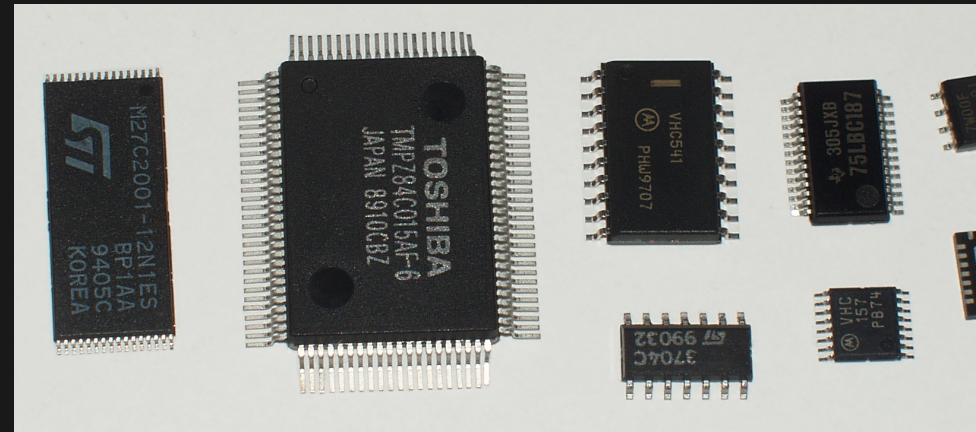
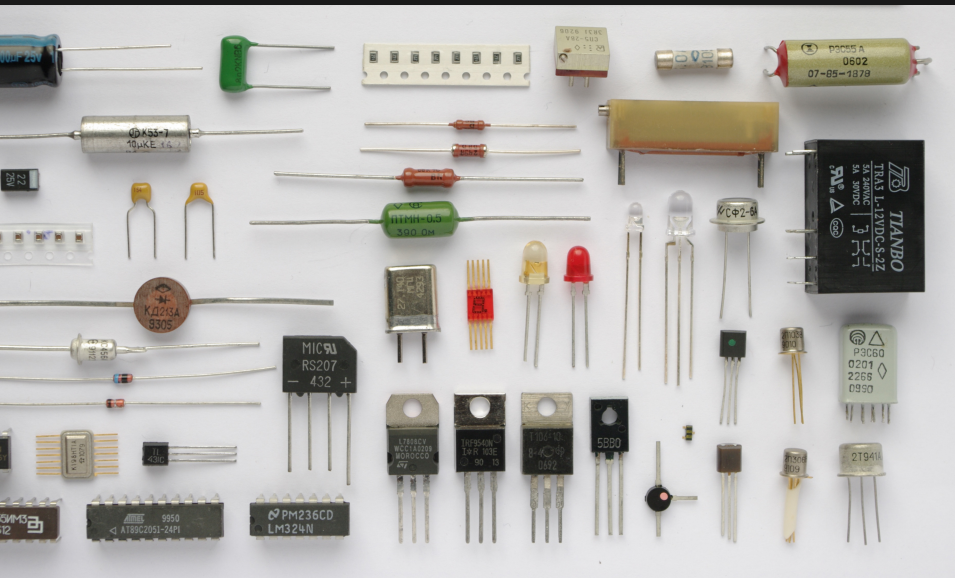
Units of measurements ☒

Both **millimeters** and **thousandths of an inch** are widely used.

1 mil = 0.0254 mm 1 mm = 39.37 mil Damn 'Murricans ☒...

Package

The form factor of the components on the PCB

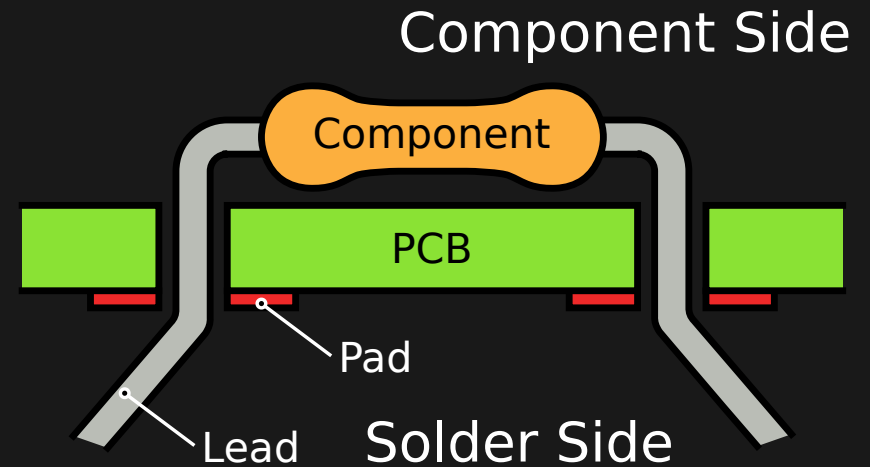
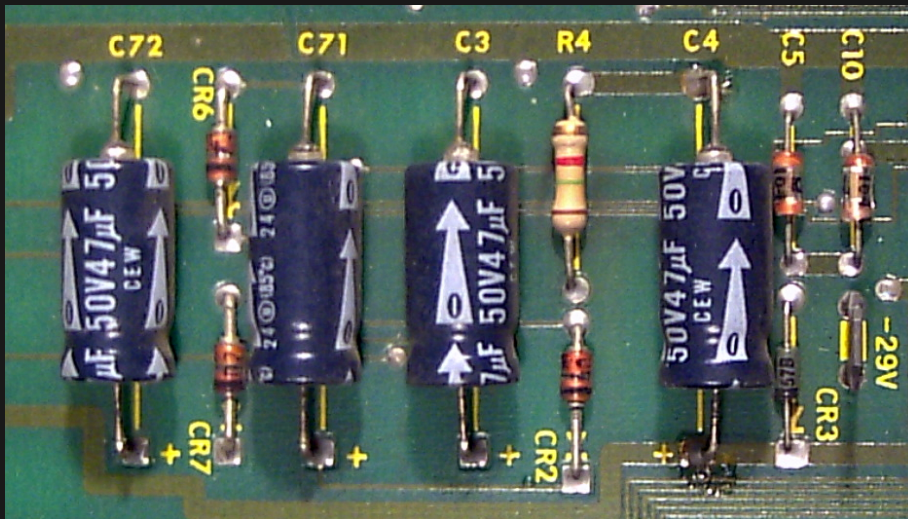


Each **package** has an associated **footprint**

Common packages

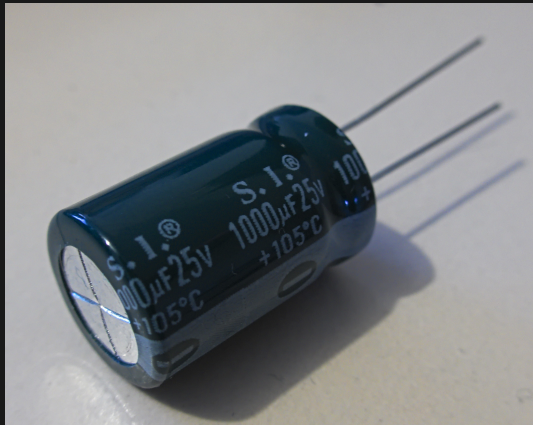
Axial leads (through-hole) on Packages

Used mainly by resistors, diodes, small inductors and (less commonly) capacitors



Radial leads Common Packages

Used mainly by capacitors and many other components

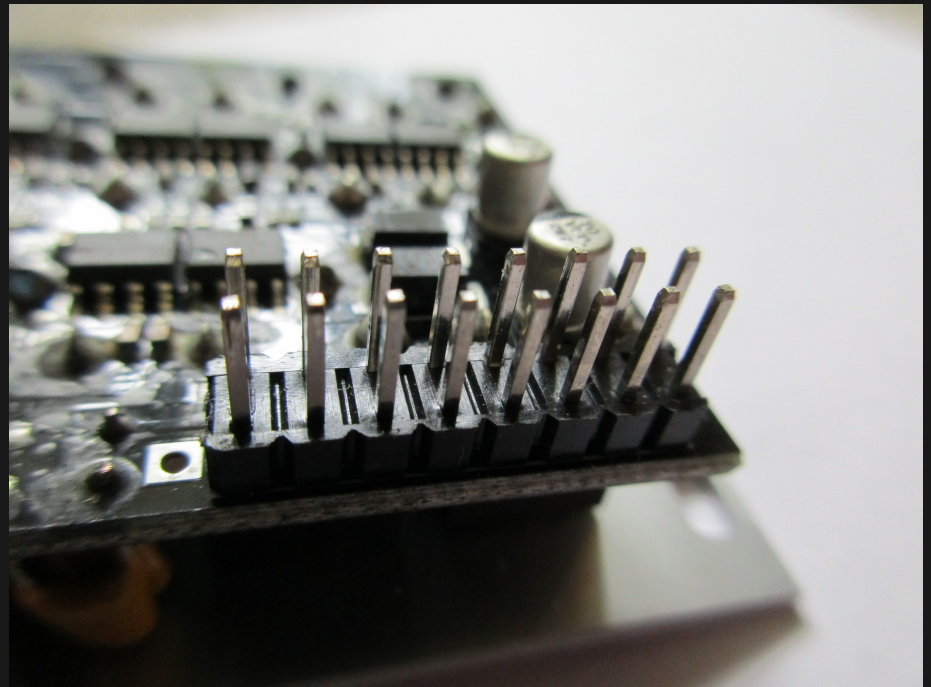


Pin header Common Packages

The simplest type of connector

They can be categorized by:

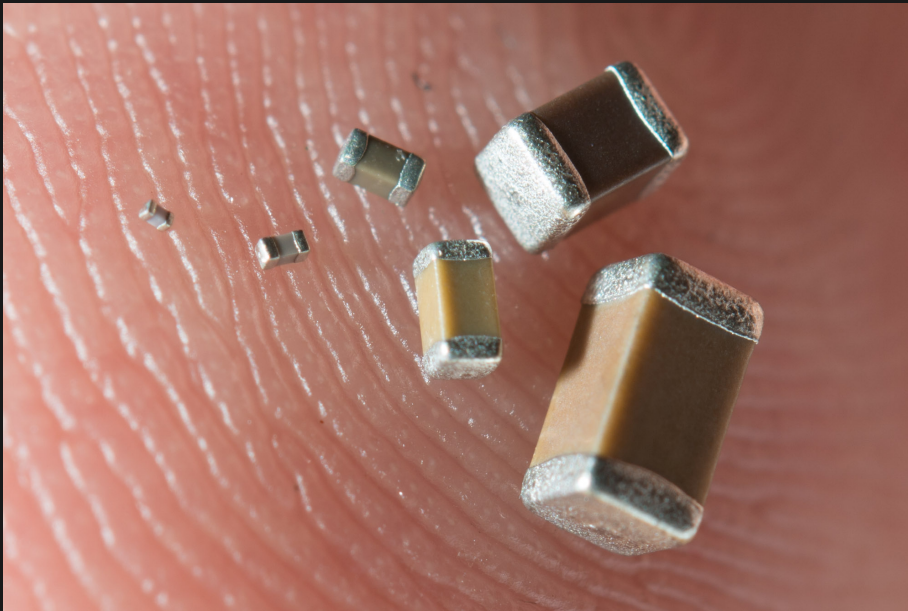
- **Male** or **female**
- **Angle**: bent or straight
- **Pitch**: distance between the pins (standard sizes are *2.54 mm* and *1.27 mm*)



They are commonly found with **through-hole** mounting.

Passive SMD Packages Common Packages

Used mainly by resistors and capacitors



Package code specify sizes

l.e. package '0805' is
0.08x0.05 inches

Sometimes metric sizes are specified

If you plan to work on these by yourself:

- 0805 can be soldered by (steady) hand under magnifying glass
- 0604 is trickier (microscope recommended)
- 0402 microscope and hot-air station are required (or *lots* of experience)

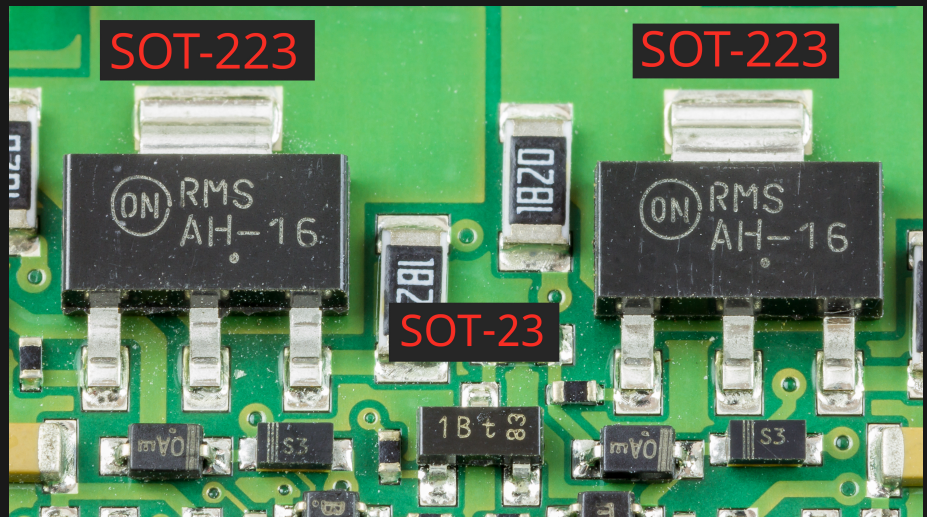
SOT Common Packages

Small Outline Transistor

Most commonly used by **SMD transistors**, but also by small **amplifiers, diodes** and **voltage regulators**.

Common variants:

- **SOT-23(-X)**: where X denotes the number of pins (up to 8)
- **SOT-223(-X)**: larger version, for more power dissipation

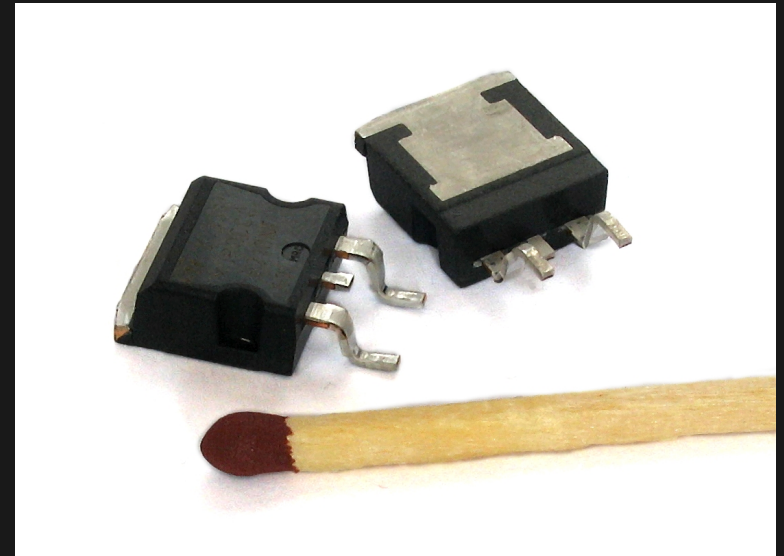


TO-252 and TO-269 Common Packages

(Double) Decawatt Package

Commonly used by **high power MOSFETs** and **voltage regulators**

They have a **metal tab** that can be soldered to the PCB, allowing an **high power dissipation**.



The soldering of the tab is tricky without a hot-air station or a high-power soldering iron

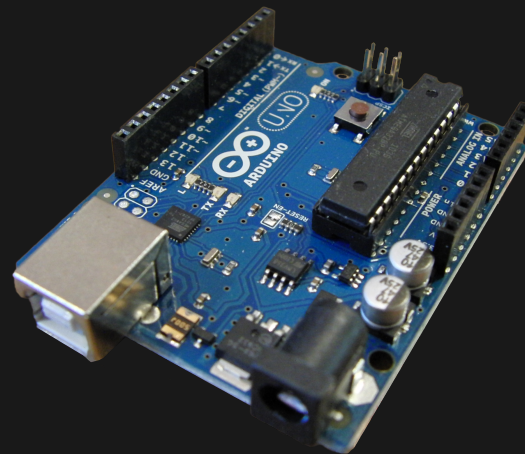
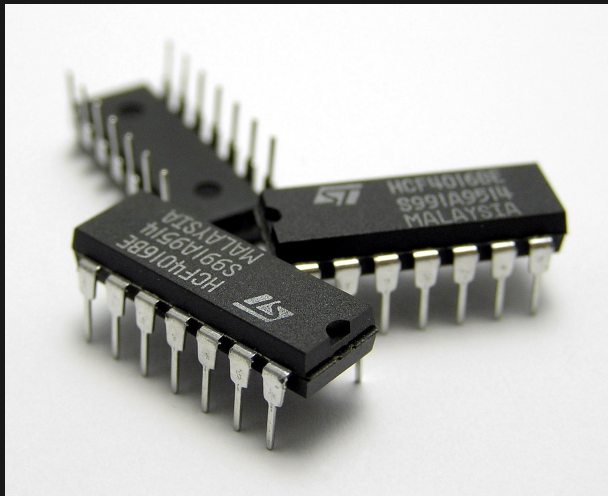
DIP

Common Packages

Dual In-line Package

Used by **Integrated Circuits**. They can be soldered or mounted on **sockets**.

The naming scheme is **DIP-X**, where X is the number of pins.



SOIC

Common Packages

Small Outline Integrated Circuit

Basically an SMD version of DIP.



The naming scheme is **SOIC-X** or **SO-X**, where X is the number of pins.

Alternative formats (wider) are **SSOP**, **TSSOP** (used by logic IC, EEPROM memory) and **TSOP** (used by old dram modules). **MSOP** is narrower.

QFP

Common Packages

Quad Flat Package

Basically a four-sides version of SOIC.

Multiple variants exist: the main are **LQFP** and **TQFP**, that are low-profile.

Some other variants are: BQFP, BQFPH, CQFP, EQFP, FQFP, LQFP, MQFP, NQFP, SQFP, TQFP, VQFP, VTQFP, TDFN, CERQUAD.



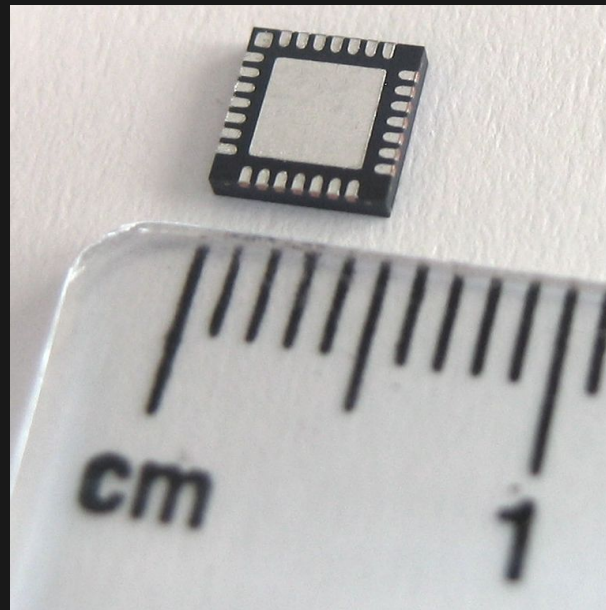
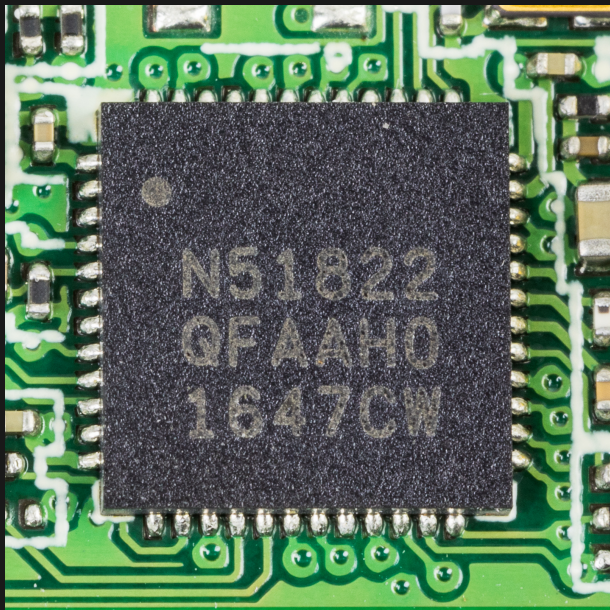
QFN

Common Packages

Quad-Flat No-leads

Basically a no-lead version of QFP.

Here too, there are a lot of variants: DFN, DQFN, cDFN, TDFN, UTDFN, XDFN, QFN, QFN-TEP, TQFN, LLP, LPCC, MLF, MLPD, MLPM, MLPQ, DRMLF, V/WQFN, HVQFN, UDFN, UQFN.



Very difficult to solder even by hot-air. Microscope needed.

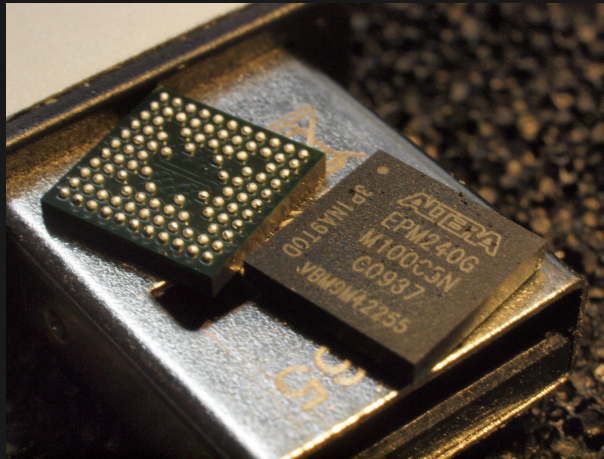
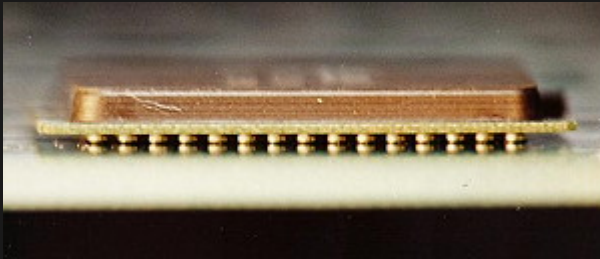
BGA

Common Packages

Ball Grid Array

The connections are made *on the bottom side* by small pre-applied balls of solder.

I refuse to list all the variants.



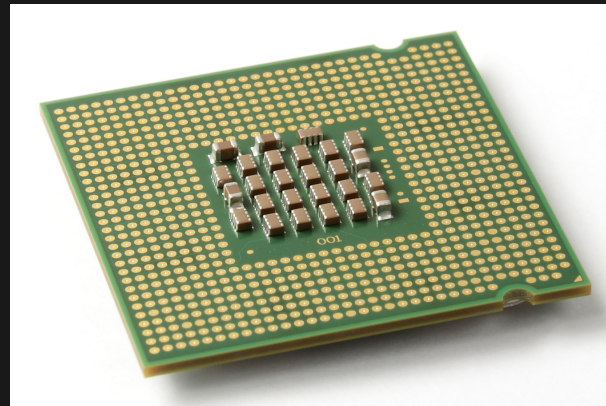
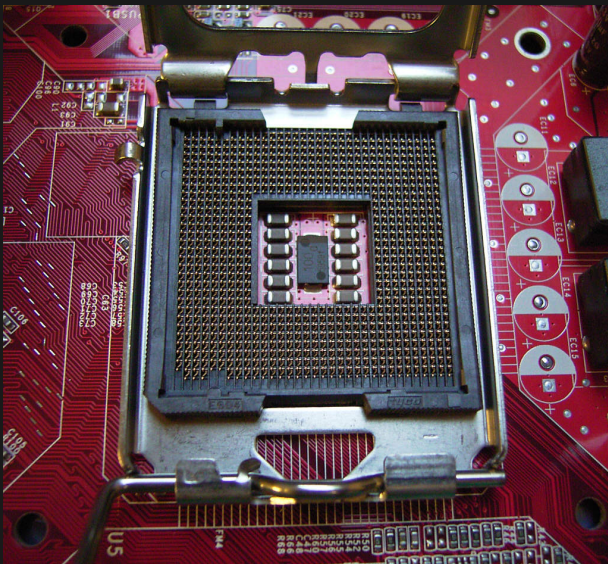
Large packages impossible to solder at home. Small packages require hot-air.
It's impossible to know the quality of the connections without x-ray.

Honorable mention Common Packages

LGA

Land Grid Array

Used by all modern desktop CPUs. They can be installed in a **socket** or soldered directly to the pcb.



Ok, we know what we are working with

Let's start designing a PCB!

The workflow

The workflow

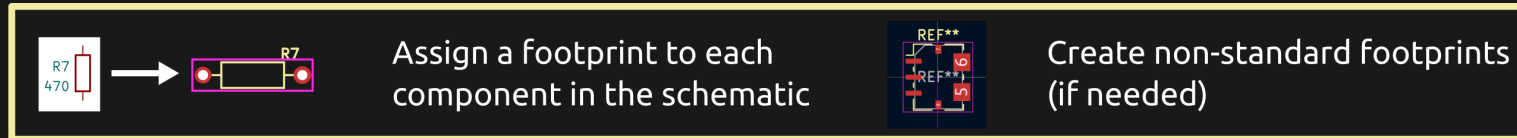
The workflow

The workflow

The workflow

The workflow

Design process



The workflow

Design process

The workflow

Design process

The workflow

Design process

The workflow

Design process

Prerequisite to all of this

**The schematic for your circuit
should already exist**

This course does *not* cover the circuit design process

For tips on how to design a circuit you can easily follow [this course](#)
(a bit expensive, but hopefully not scam).

First step in PCB design:

Evaluating constraints

*In theory, theory and practice are the same.
In practice, they are not.*

Benjamin Brewster

What is a constraint?

A limitation on how "freely" we can design our PCB

Who defines the constraints?

The real world

Application Example specifications of constraints

- Board/components **shape** and **size**
- Mechanical **stress** endurance
- High/Low **temperature** endurance
- **Connectors/components** position
- **Mounting** system
- Low **EM emissions**
- ...

Circuit Specifications of constraints

- **Impedance** requirements
- Immunity to **EMI**
- Support for **high-frequency signals**
- Thermal **dissipation**
- **Decoupling** requirements
- ...

Production capabilities of constraints

- Traces **spacing**
- Number of **layers**
- Supported **via types**
- Trace **thickness**
- ...

Soldering capabilities of constraints

- **Hot air** availability
- **Microscope** availability
- **Steady hands** availability
- ...

When should these constraints be evaluated?

Always

Before, during and after the design

...also while you eat, while you sleep...

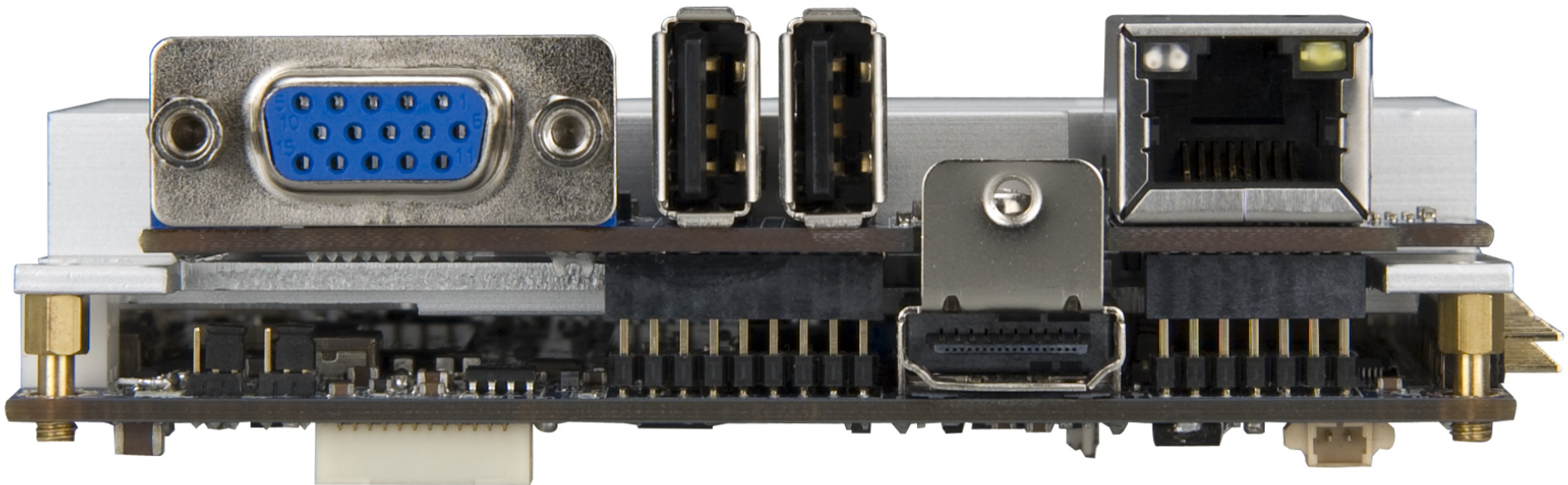
Component placing

Useful rules

1- Fixed components placement rules

The first things to place

Connectors and components with placement restrictions should be prioritized.

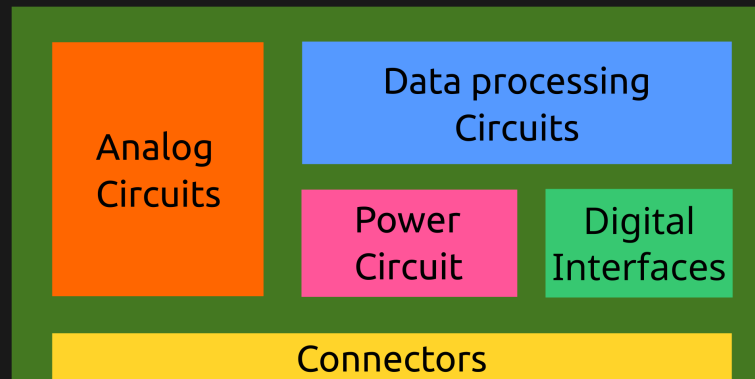


2- Floorplanning

Component placing rules

Divide the PCB into functional areas

- Separate analog and digital sections
- Noise sensitive paths should not traverse noise generating sections



3- Big components first placing rules

Go from large components to smaller ones

Large components usually have some circuitry associated to them



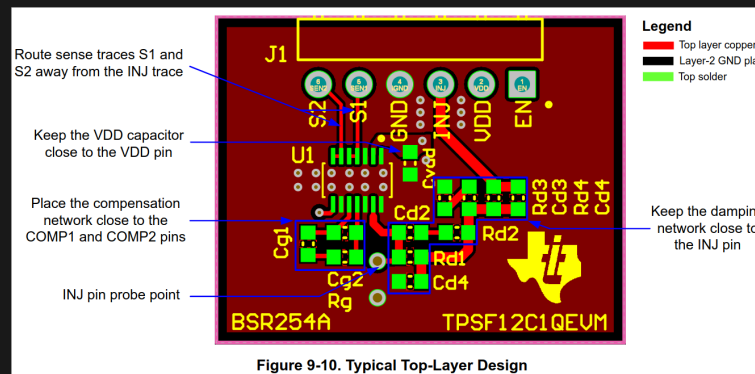
Good luck with this

4- Look at the datasheet replacing rules

Reference implementations are your Bible

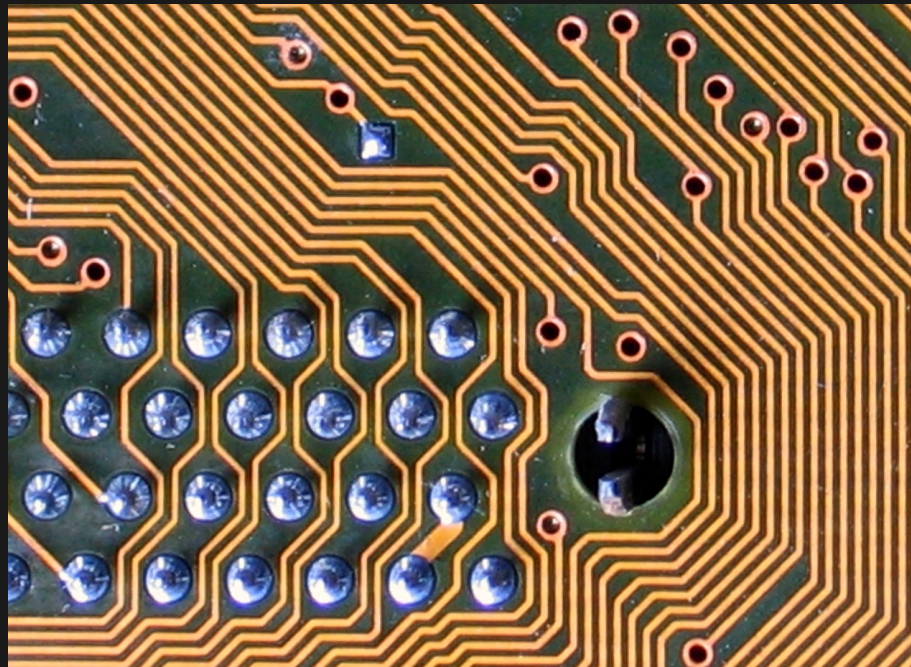
In the components datasheets you can find
examples of applications.

Study and understand the layout, the manufacturer knows
best!



5- Leave space for routing

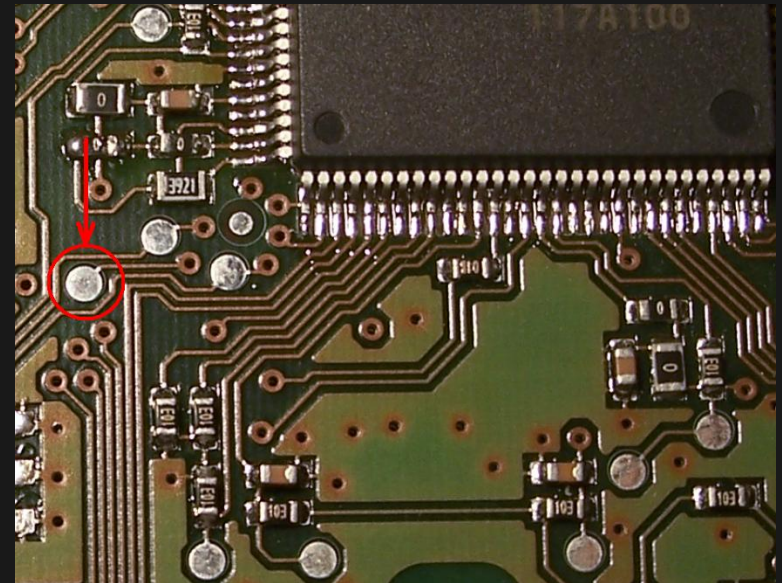
Traces need their space!



6- Testpoints component placing rules

Because you will need them!

Testpoints are exposed pads in the circuit that you can **probe**. They are useful to trace failures and the behaviour of your circuit



Other things ~~Component~~ placing rules

- Orient the components in the **same direction**
- Account for **soldering space**
- **Thermal** awareness
- **High-speed signals** requirements awareness
- **Every other constraint**

Routing

Useful rules

1- General planes Routing rules

Ground planes and Power planes are very useful

Since ground and power are needed by almost every component, it is useful to have them available everywhere on the board

If we work with 4-layers boards, we can afford to dedicate an **entire layer** to each plane.

We can **split** the power plane in multiple parts, one for each needed voltage.

It's also possible to split the ground plane, but don't do it if you don't know what you are doing!

2- Return path Routing rules

Each signal should be close to its return path to ground

This is needed to minimize noise coupling

Having a ground plane simplifies this.

3- Differential signals Routing rules

Should be routed as a pair

This is because common-mode noise will be rejected, so you want them to be exposed to the same noise.

4- Trace width

Routing rules

Power traces are wider than signal traces

It's important to know the **impedance** of the traces used for:

- high **currents**
- High **speed** signals
- **Analog** signals

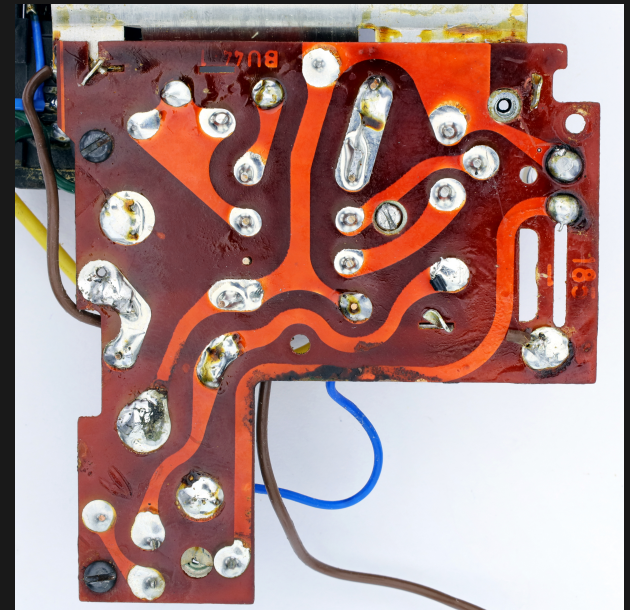
Trace width is correlated to trace impedance!

5- Avoid using 90 degree trace angles

Use 45-degree angles instead

The reasons are multiple (and very debatable).

Key reason (*for us*): DIY **manufacturing**
90 degree trace angles is difficult



If Pollock was a PCB designer

The other concerns are about EM emissions and signal integrity, topics out of the scope of this course

Other things

Routing rules

- **Overlaid traces** for higher current
- **Via fences** for better EMI immunity
- **High-speed signals** requirements awareness
- *Every other constraint*

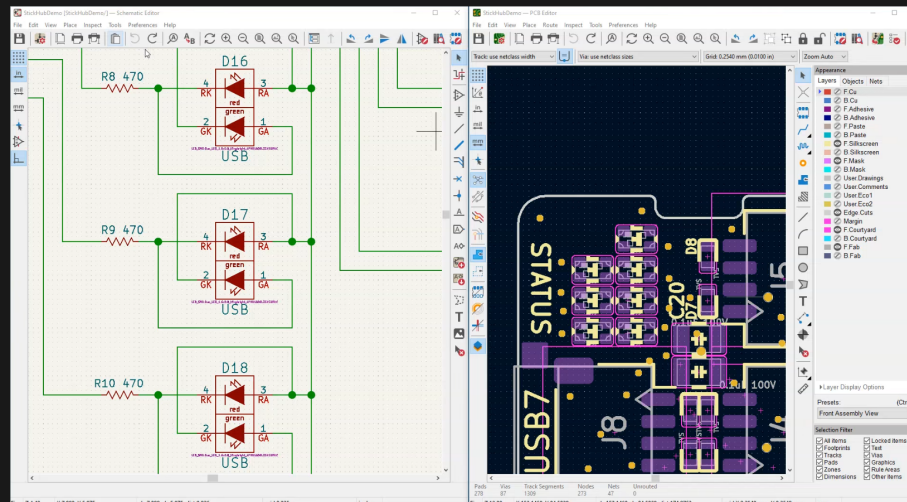
Do I have to do all of this with pen and paper?

Of course not!

Or...

You can use an EDA software!

Electronic design automation (EDA) (...), is a category of software tools for designing electronic systems such as integrated circuits and printed circuit boards.



Such as...

EDA softwares

Altium Designer

Such as...

Autodesk Eagle

Such as...

KiCad

What's next?

We will design a PCB with KiCad!

If you want to follow the steps yourself, *download and install
KiCad on your computer*

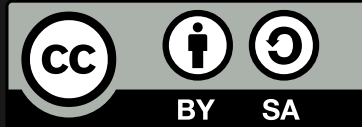
Link to Download

The download is 1.5GB, the required space on disk is 10GB.

Thank you for your attention!



POLITECNICO OPEN
unix LABS



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).

Source code available [here](#)

Luca Biscaldi
bisca@poul.org

