# Embedded systems

Exercise session 3

Important Circuit Components

Circuit Design

# Today

Part 1: Common components and circuits

- Voltage Regulator
- RC Lowpass Filter
- Optocoupler
- MOSFET Switch, MOSFET Drivers
- H-Bridge
- Brushed DC motors
- Stepper motors
- Servomotors

Part 2: Electronic circuit design procedure

- Components choice
- Electronic schematic
- PCB design

# Datasheet

On the market, there are <u>lots of</u> different components available. Some of them might look similar (or even identical) from the outside while actually having different specifications. But how can one find out which are those differences and which component fits its need? The answer is by looking in the datasheet.

Datasheet = document holding all useful information on a given electronic component. Each component has its own.

What kind of information can be expected to be in it?
- Physical and electrical properties;
- Normal working conditions;
- Absolute maximum ratings;
- Recommended connections...

Live examples : PIC16F1789, Sharp, LM7805

# Voltage Regulator

**Concept**

The voltage regulators can be used to generate a stable voltage output from a wide range of input voltages.

**Why?**

- The power source of a circuit can have an output voltage that depends on the power drawn by this circuit (typically batteries).
- Some components need a stable and clean voltage reference to work correctly.

**Note**

- Most regulators have a low efficiency and are not designed to transmit a lot of power ⇒ Be careful on what you plug on them.
- When making power electronics, prefer the use of DC/DC converters.

# Circuit Clean Voltage Input

**Stabilization**

- The microcontroller should be powered with a stable DC power source.
- Power lines can be noisy or unstable.
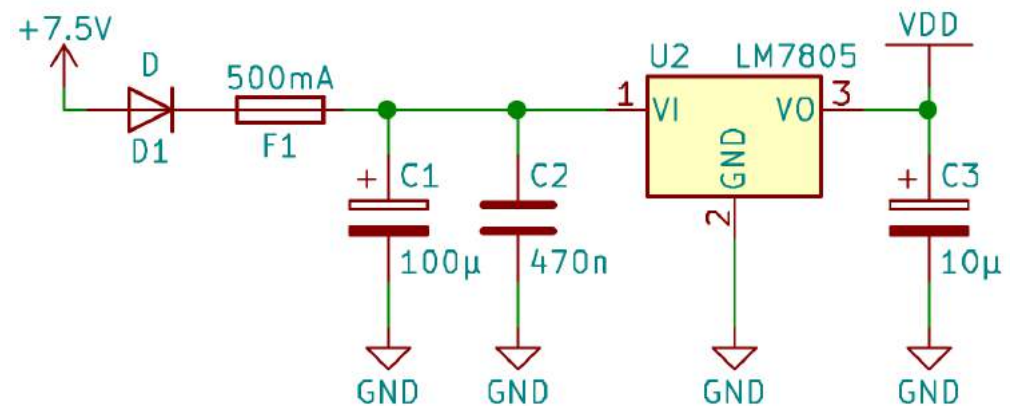
⇒ Stabilizing the input voltage is required

**Input protection**

Needed to prevent circuit components from destruction in case of bad manipulations.

The diode D1 protects the circuit from reversed input polarity.
The fuse F1 protects the components in case of short circuit.
The reservoir capacitors C1, C2 and C3 Stabilize the input voltage.

# RC Lowpass Filter

**Concept**

- Simple circuit made of a capacitor C and a resistor R used to filter high frequencies
- Can also be an undesirable effect of the circuit design which can lead to signal transmission issues

**Note:** RC filters are not suitable for power transmission!

Step response of the circuit : $V_C(t) = V\left(1 - e^{-\frac{t}{RC}}\right), \quad \tau = RC$
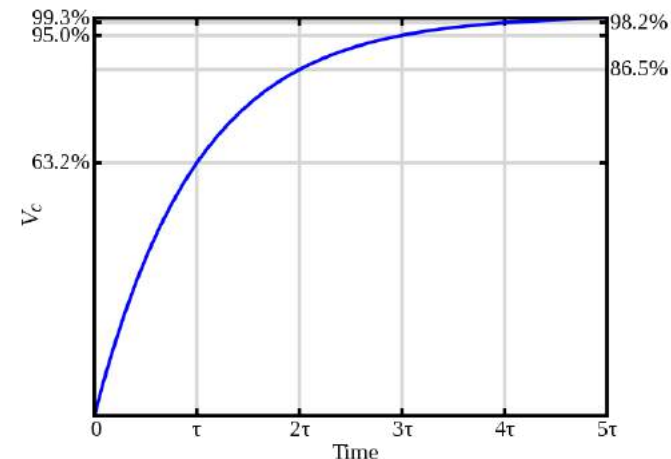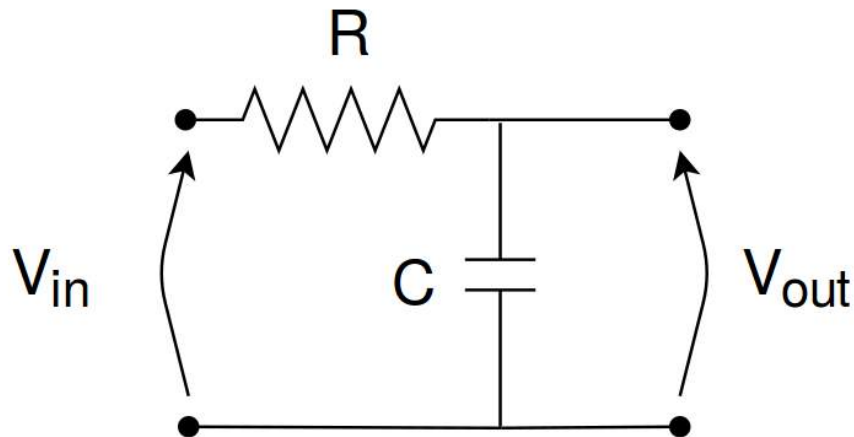
# Optocoupler

**Concept**

Component (grey box on the circuit) used to transmit digital signals between devices that don't have the same ground reference.

**How does it work?**

The device contains a LED and a phototransistor placed in front of each other:

1. The LED converts electrical input signal into light
2. The phototransistor generates or modulates an electrical current when exposed to light emitted by the LED
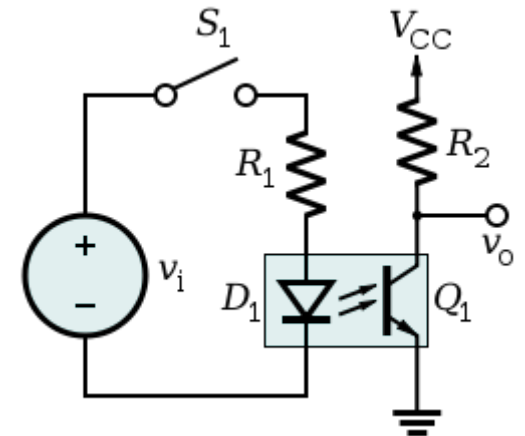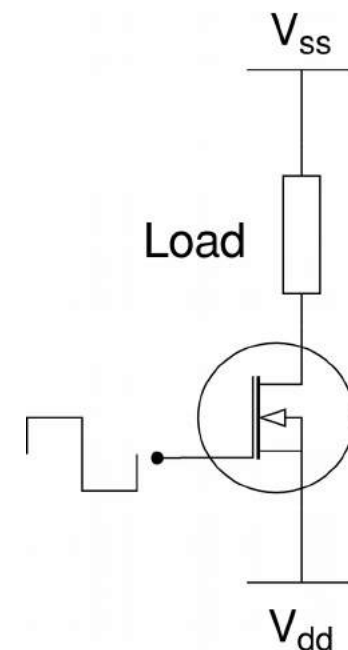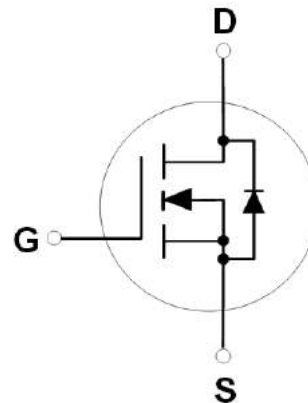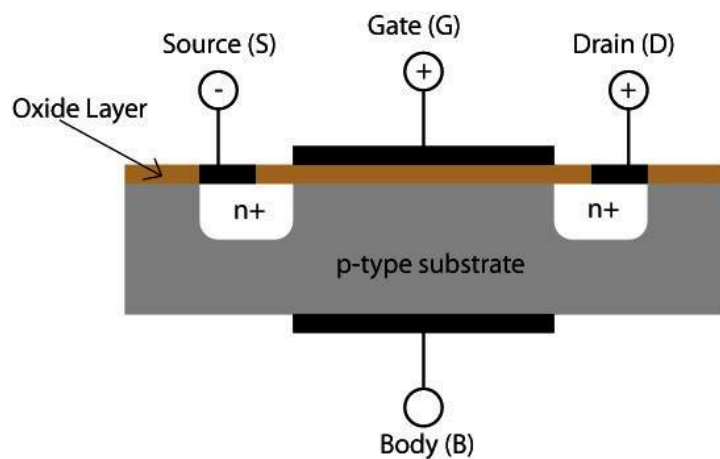
# MOSFET Switch

**Concept**

MOSFET transistors can be used as switches to turn on and off high-power loads with a microcontroller signal.

**How does it work?**

Either in *cutoff* or *saturation* mode depending on the voltage applied between the source and the gate (with $V_{GSthreshold}$ being different for each transistor model):

- Cutoff : $V_{GS} < V_{GSthreshold}$ : $I_{DS} \simeq 0 \Rightarrow$ Switch is opened
- Saturation : $V_{GS} > V_{GSthreshold}$ : $V_{DS} \simeq 0 \Rightarrow$ Switch is closed



8

# MOSFET Drivers

**Problem**

MOSFETs are not perfect :

- Parasitic resistance : $R_{DS(ON)} \neq 0\Omega \Rightarrow P_{dissipated} = R_{DS(ON)}I_{DS}^2$
- Parasitic capacitances $\Rightarrow$ Commutation time $\neq$ 0s $\Rightarrow$ Power loss (in form of heat) during switch transitions
  - $\rightarrow$ Can be problematic for high-frequency switching (too much heat dissipated)

**Solution**

Use a MOSFET driver to minimize the switching time.

  - $\rightarrow$ Can provide high-current spikes to quickly charge the parasitic capacitances and therefore to quickly operate the switch
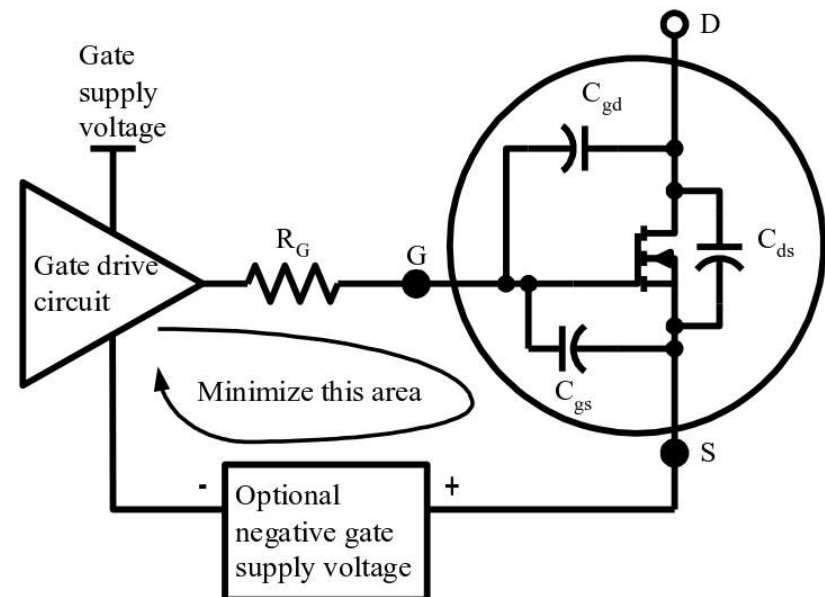
# H-Bridge

**Concept**

- Combination of 4 MOSFET switches
- Allows to control the direction and the amount of current powering a load.

**How does it work?**

- Direction of the current set by activating some given pair of switches
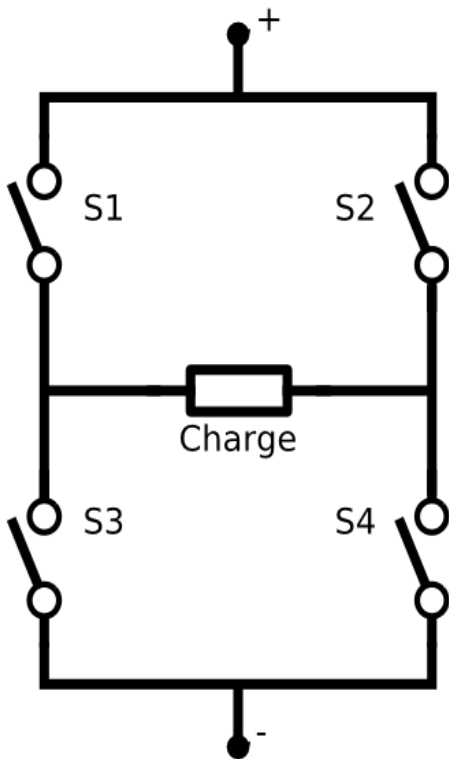- Power modulated thanks to PWM signal

**Note**

H-bridges can be crafted by hand by using several components or bought as a single component.

# H-Bridge

**Concept**

H-bridges are a particular combination of 4 MOSFET switches which enable to control the direction and the amount of current which goes through the load.
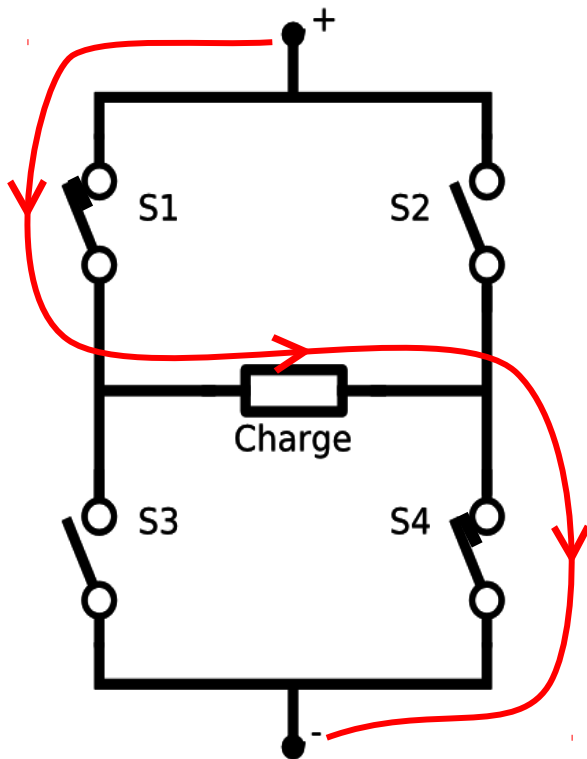
**How does it work?**



| S1 | S2 | S3 | S4 | State |
|----|----|----|----|-------|
| 0 | 0 | 0 | 0 | **No current** |
| 1 | 0 | 0 | 1 | **Positive current** |
| 0 | 1 | 1 | 0 | **Negative current** |
| 1 | 1 | 0 | 0 | **Motor short-circuit** |
| 0 | 0 | 1 | 1 | **Motor short-circuit** |
| 1 | 0 | 1 | 0 | **!!! Power Supply Short-circuit !!!** |
| 0 | 1 | 0 | 1 | **!!! Power Supply Short-circuit !!!** |

# H-Bridge

**Concept**

H-bridges are a particular combination of 4 MOSFET switches which enable to control the direction and the amount of current which goes through the load.
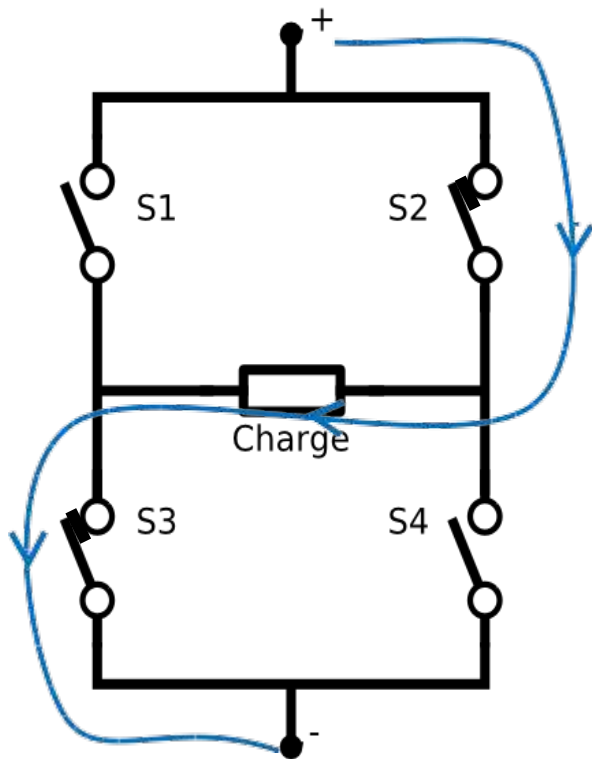
**How does it work?**



| S1 | S2 | S3 | S4 | State |
|----|----|----|----|-------|
| 0 | 0 | 0 | 0 | No current |
| 1 | 0 | 0 | 1 | Positive current |
| 0 | 1 | 1 | 0 | Negative current |
| 1 | 1 | 0 | 0 | Motor short-circuit |
| 0 | 0 | 1 | 1 | Motor short-circuit |
| 1 | 0 | 1 | 0 | !!! Power Supply Short-circuit !!! |
| 0 | 1 | 0 | 1 | !!! Power Supply Short-circuit !!! |

# H-Bridge

**Concept**

H-bridges are a particular combination of 4 MOSFET switches which enable to control the direction and the amount of current which goes through the load.
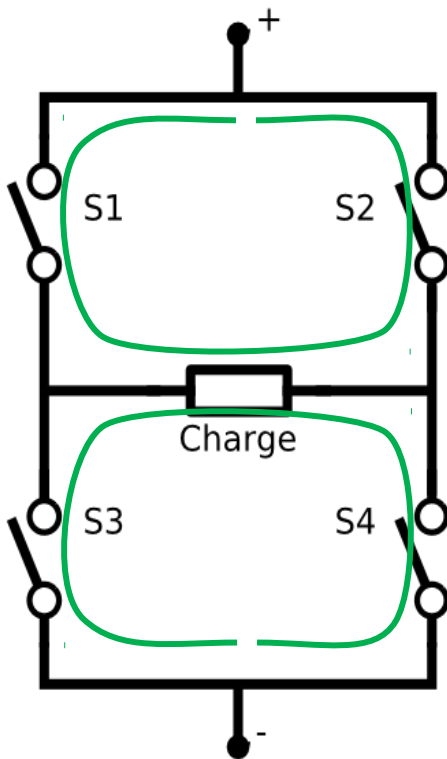
**How does it work?**



| S1 | S2 | S3 | S4 | State |
|----|----|----|----|-------|
| 0 | 0 | 0 | 0 | No current |
| 1 | 0 | 0 | 1 | Positive current |
| 0 | 1 | 1 | 0 | Negative current |
| 1 | 1 | 0 | 0 | Motor short-circuit |
| 0 | 0 | 1 | 1 | Motor short-circuit |
| 1 | 0 | 1 | 0 | !!! Power Supply Short-circuit !!! |
| 0 | 1 | 0 | 1 | !!! Power Supply Short-circuit !!! |

# H-Bridge

**Concept**

H-bridges are a particular combination of 4 MOSFET switches which enable to control the direction and the amount of current which goes through the load.

**How does it work?**

| S1 | S2 | S3 | S4 | State |
|----|----|----|----|-------|
| 0 | 0 | 0 | 0 | No current |
| 1 | 0 | 0 | 1 | Positive current |
| 0 | 1 | 1 | 0 | Negative current |
| 1 | 1 | 0 | 0 | Motor short-circuit |
| 0 | 0 | 1 | 1 | Motor short-circuit |
| 1 | 0 | 1 | 0 | !!! Power Supply Short-circuit !!! |
| 0 | 1 | 0 | 1 | !!! Power Supply Short-circuit !!! |

14

# Brushed DC motors

**Concept**

The brushed DC motor is the simplest DC motor. It is made of a magnetic stator, a rotating coil and a rotor.

**How does it work?**

Current is delivered to the coil trough brushes touching a contact ring. The current in the coil creates a magnetic field that has to align with the one of the stator. This creates a torque that makes the coil rotate. At some point, the rotor turned so much that the contacts on the ring have changed the brush they are touching. This causes the current to reverse in the coil and an inversion of the magnetic field, that then has to realign with the one of the stator which keeps the rotor in motion.



15
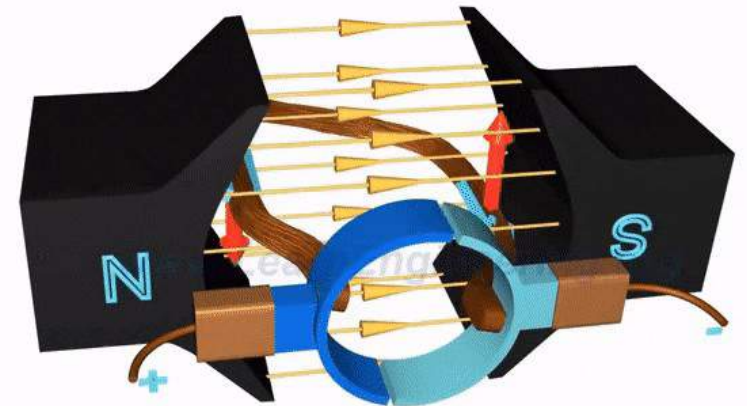
# Brushed DC motors driving

Driving a DC motor is relatively easy :

• To change torque and speed, change the applied current

• To reverse the direction, reverse current direction (i.e. inverse voltage polarity)

• To stop it, turn the power supply off

**Low-side drive circuit example**
• Use a MOSFET as a switch to pulse width modulate the supplied voltage

• D1: protection against voltage spikes that could harm the MOSFET

• R1: pull-down resistor to be sure that the MOSFET is off when no voltage is applied

• MD : MOSFET driver
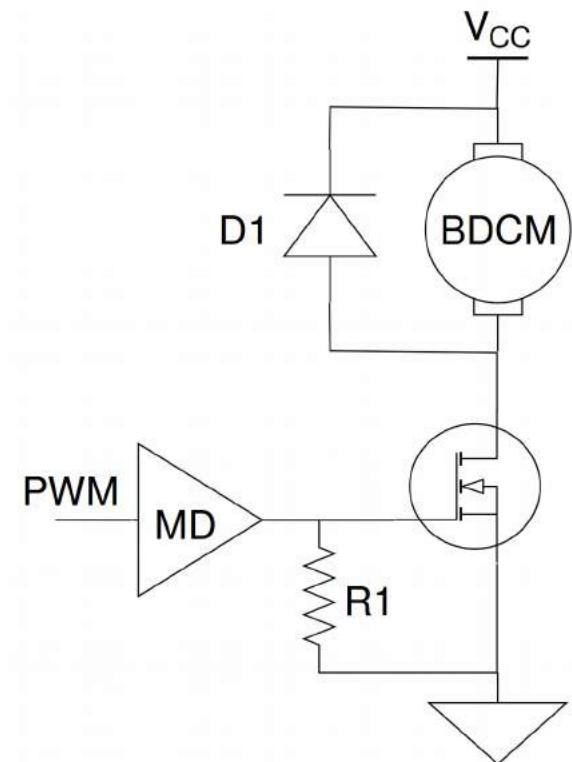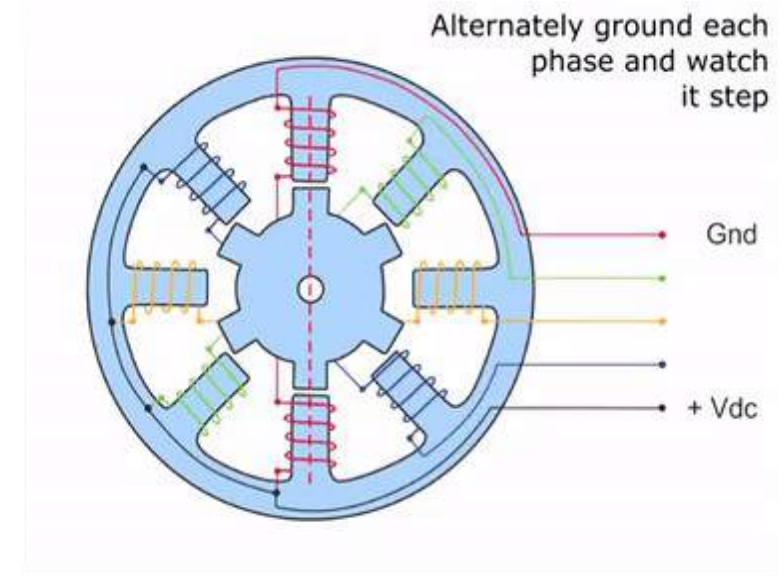
# Stepper motors

**Concept**

The stepper motors are made of a soft iron rotor with square teeth on its side and of several pair of coils for the stator

**How does it work?**

Making current flow in two opposite coils will create a magnetic field. This magnetic field will go trough the rotor. The system will tend to reduce its reluctance. The only degree of freedom available to do this is the rotor rotation. Reduction of the reluctance is achieved by aligning a pair of teeth with the coil. The rotor will therefore move to get this alignment. By successively powering pairs of coil, it is possible to make the rotor do continuous rotations.



Alternately ground each phase and watch it step

Gnd

+ Vdc

# Stepper motors driving

**Generalities**

Two-phase stepper motors are the most common. They can be unipolar or bipolar.

**Idea**

To make the motor rotate, it is required to send correctly synchronized pulses to each phase. Different driving methods are possible depending on the desired application (more precision vs more torque for example)
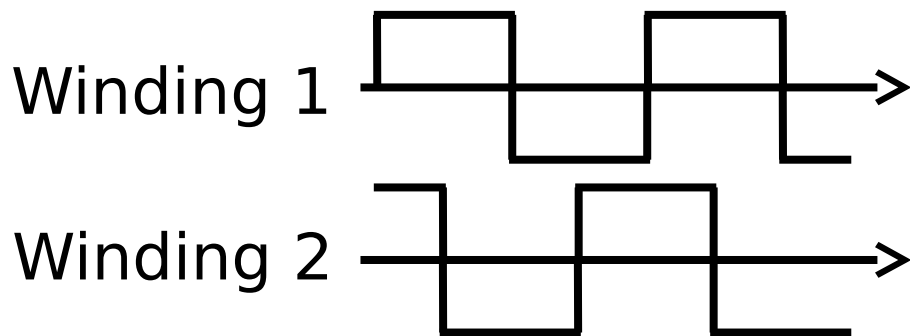


Example of driving : Full step drive (for two-phase bipolar motor)
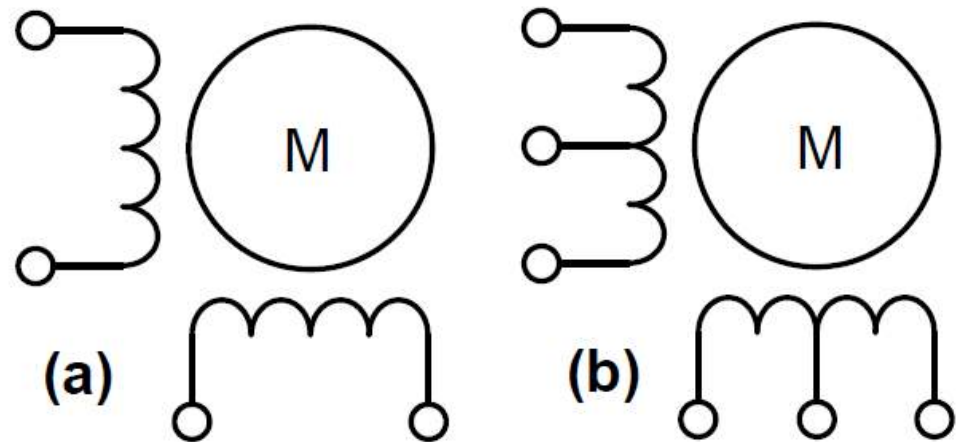
Two phases stepper: (a) Bipolar (b) unipolar

# Servomotors

**Concept**

DC brushed or brushless motors equipped with a sensor and (in general) with an embedded controller that is used to precisely set the angular position of the motor based on an input signal.

**How does it work?**

For most servomotors, the angular position is given by the length of the duty cycle of a PWM signal.



Servo Motor PWM Timing Diagram
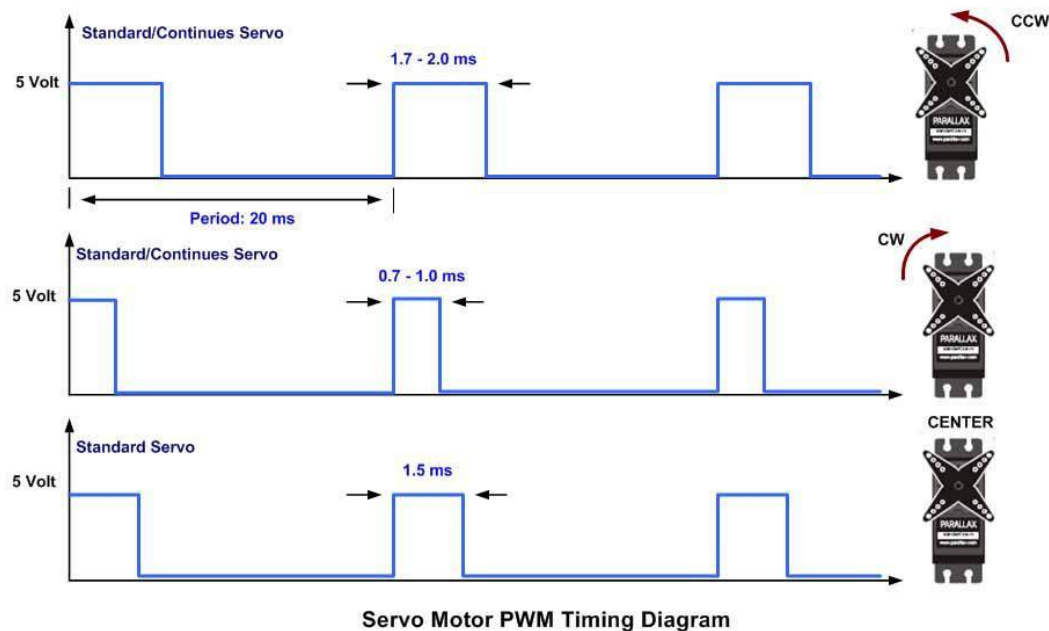
# Electronic circuit design

**Electronic circuit design == rigorous process**

- No place for trial and error
- Need careful planification → purpose of a schematic

An electric circuit schematic is the plan of this circuit. It gives information on all its components and how they interface:

- Connections are represented as simple lines;
- Components are represented by their electrical symbol;
- Each component symbol should be drawn with two additional information:
    - A label which uniquely identifies the component on the schematic
    - A name that gives the component's reference (model), or a value in the case of resistors, capacitors and inductors.

# Electronic circuit design

**Good Practice rules**

- Currents should flow from top to bottom (or equivalently, areas of higher voltage should be located above those of lower voltage);
- Each component MUST be represented by its symbol;
- Connection lines should be as short and as straight as possible;
- Use function blocks (only) if your circuit is complex;
- Connection between several lines is highlighted with a point (when there is no point, the schematic is interpreted as simple lines crossing without connection)

# Electronic circuit design

## Improvable schematic example (find the mistakes)
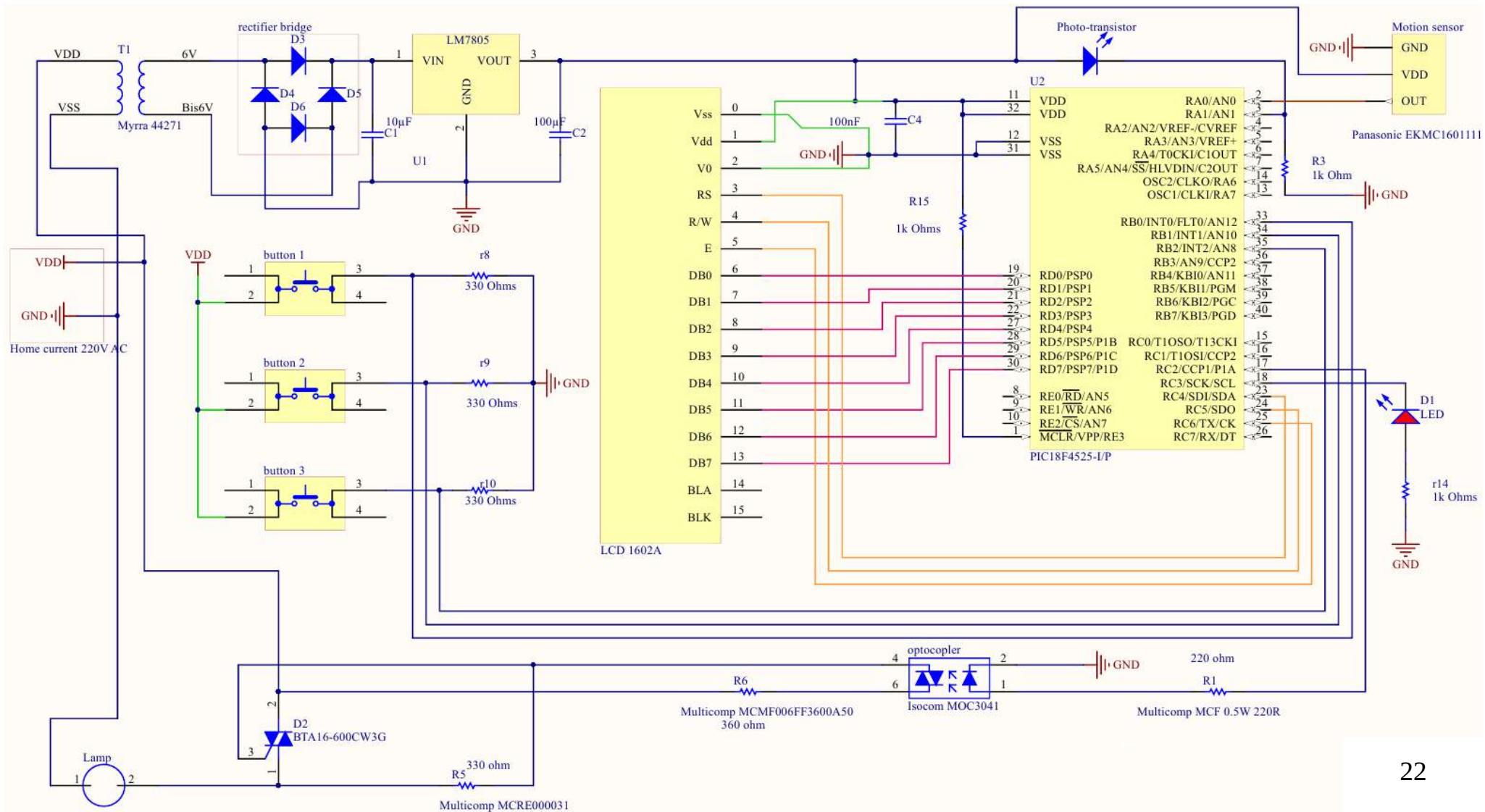
# Circuit implementation in practice

**The circuit schematic is only the first steps towards a working prototype!**

Circuit designers have to answer the following questions next:

- Exact specifications of the component that will be bought? (ideally already answered at the previous step)
- Packaging of each component?
- How does the component interface with the rest of the circuit?
- Which support is he going to use to implement its circuit?
- How will the layout is his circuit be organized?

All those questions are critical and must be answered accordingly.

# Choosing a component

**What you have to take care about in general**

- Input and output ratings
- Packaging (physical shape and type of connections)
- Maximum allowed power
- Bandwidth if you need fast switching

**Note**: A same "meta-component" (e.g. LM7805) can have several variants that differ by their absolute maximum ratings (such as output power for example)

# Common components example

**Voltage regulators**

- LM7805 : Input 10→35V; Ouput 5V
- MCA05D15D : Input 5V; Output -15V and 15V

**Optocoupler :** HCPL2211

**MOSFET switch :** heavily depends on your needs (see the power management and semi-conductors categories on Farnell)

**H-Bridges :** MC33887, L293NDE, L298

# Packaging types

**Through-hole technology**

Dual In-line
Package (**DIP**)
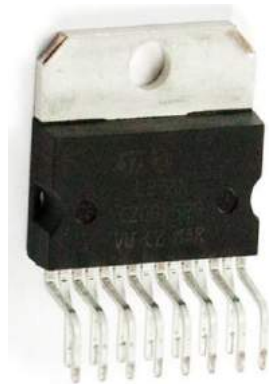
Transistor Outline
(**TO-220**)

**Surface-mount technology
(can only be mounted on PCBs!)**
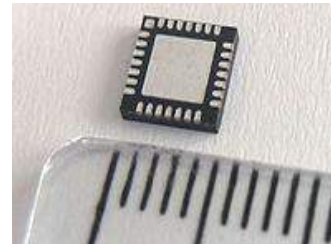
Dual/Quad-Flat
No-leads (**DFN**/**QFN**)

Quad Flat
Package (**QFP**)

Small Outline
Integrated Circuit
(**SOIC**)

Surface-mount
components (**SMC**)

# Component interfaces

**Headers/pins/tulip ports**

- Direct connection with the board.
- Used to keep your important components
  easily removable (only for THT).



**Connectors**

- Interface between components output and the board
- May have to be assembled by hand!
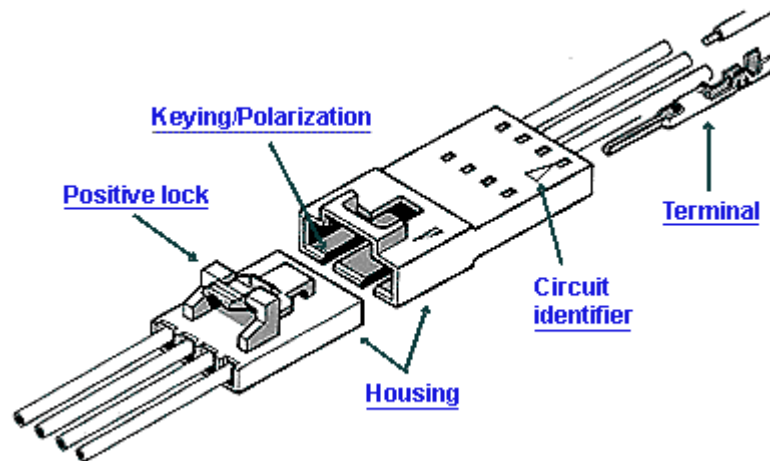
# Circuit Support I

**Breadboard**

- Can host only trough-hole components
- Provides an internal circuitry
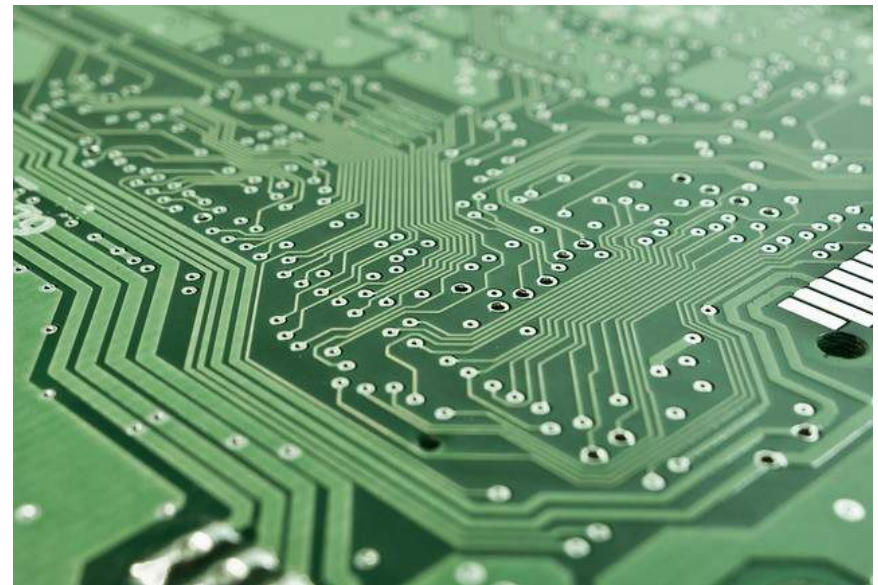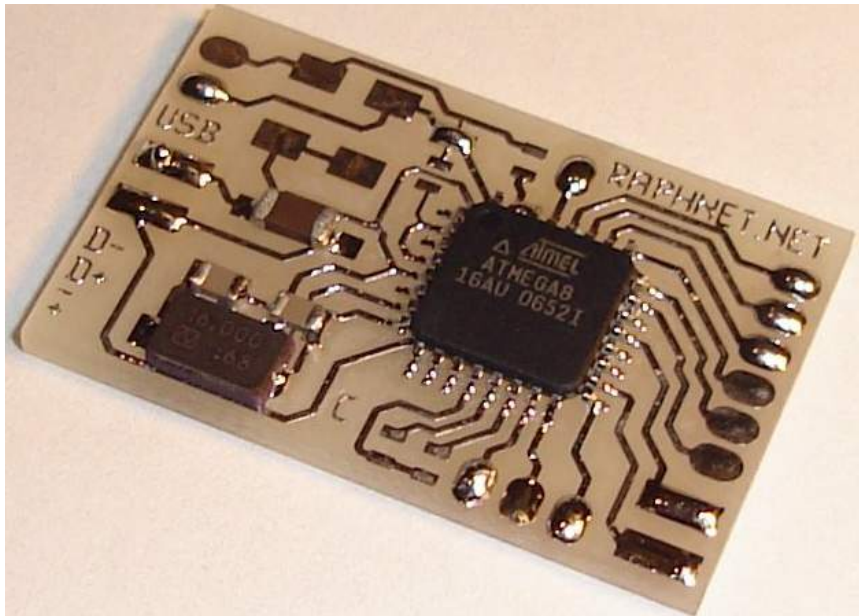- Extremely useful for fast experimental prototyping

**How does it work?**

# Circuit Support II

**PCB**

The Printed Circuit Board is a a board with some copper tracks which:

- Holds mechanically all the components
- Performs all the electrical connections between the components
- May perform an active role (antenna, ground plane, …)
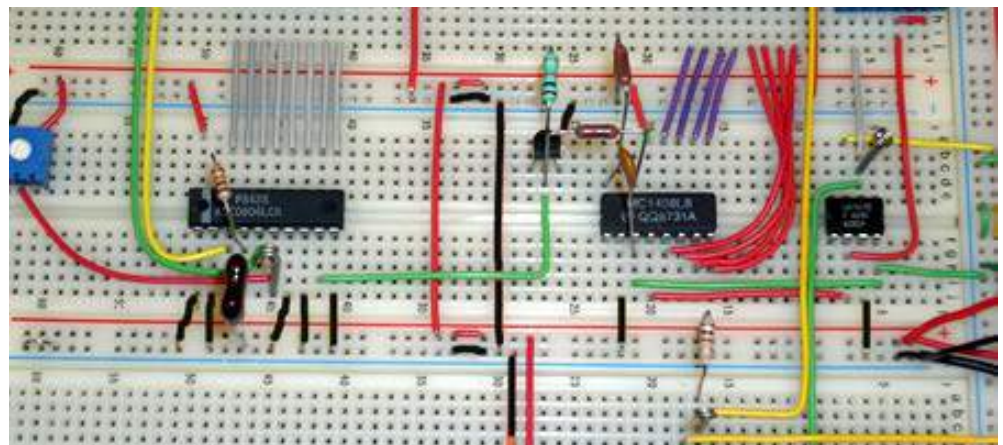- May be composed of several layers

# Circuit Layout | Breadboards

- Circuit layout strongly constrained on breadboards
- Can quickly become messy
    - ➔ Need to keep things clean for easier debugging
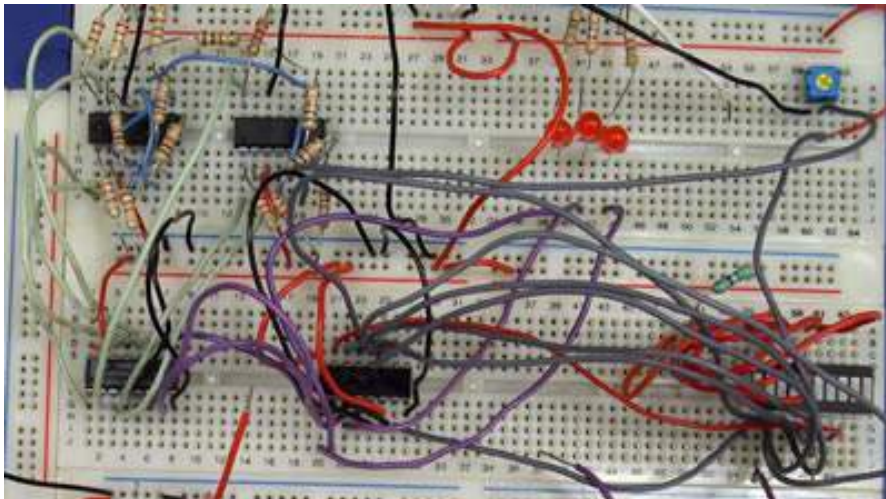
**Avoid this**                                        **Prefer this**

# Circuit Layout | PCBs

Thinking the layout of a PCB is much more complex as the degree of freedom is much wider than for breadboards.

Challenges of a PCB design:
- Needs to be done specifically for one given circuit
- Get a support on which components can be soldered
- Make correct connections between components
- Choose an efficient layout

Example of CAD softwares : KiCad, Geda, Eagle,... Some are free others not.

# PCBs and projects

Montefiore has a PCB printer that can be used for your projects. Designing and printing a PCB is however NOT a mandatory scope of the project. The choice of making one or not is left to you.

The next slides are dedicated to PCB design. They are there to help you in the design process. Feel free to read them and ask questions on them whenever you want during the year.

# PCB | Design procedure

**Aim**

Create a bunch of files that can be interpreted by
a PCB printer and that tell:

- Where to remove copper for each layer
- The location of the holes to drill
- The location of the holes to anodize
- Where to put varnish
  - → An example of such file is given on the website



**Procedure**

- Design your circuit (draw your schematic)
- Choose the model (and therefore the footprint) of your components
- Organize the positioning of your components on the board depending on their footprint
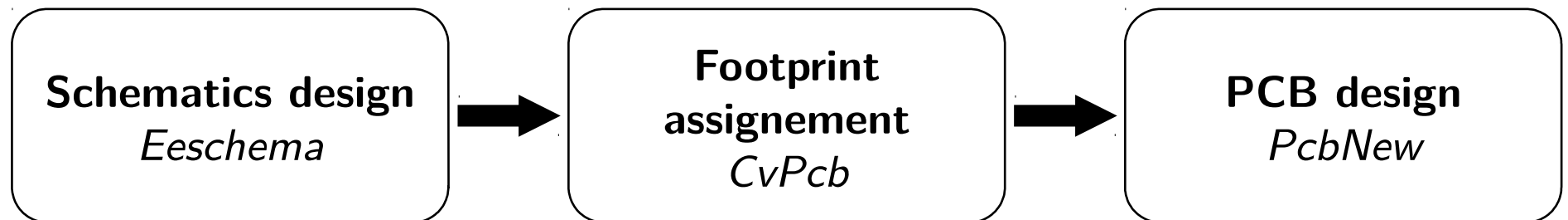- Draw the tracks to make the links between the components
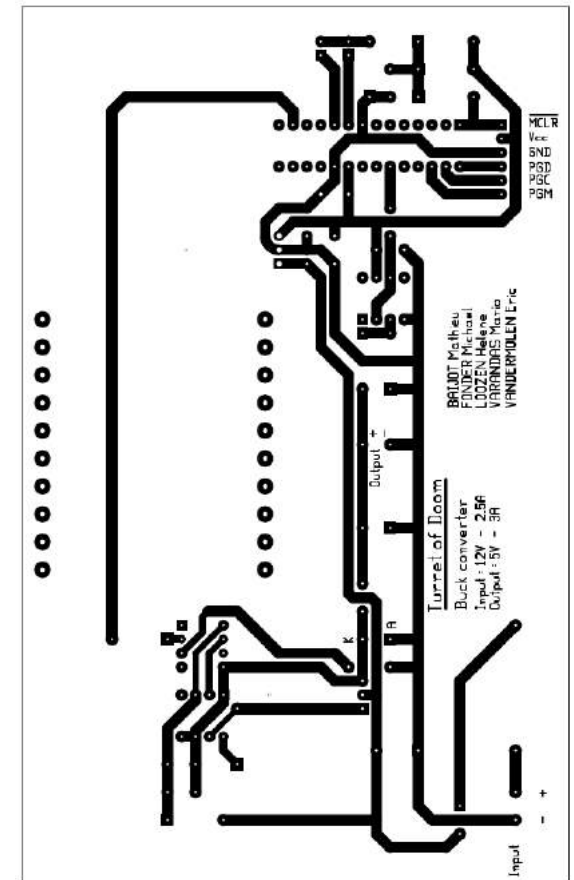
# KiCad

**Strengths**

- Accessible learning curve
- Good libraries
- Good community with good tutorials
- Free and cross-platform

  → For download and tutorials link, see the exercise session website

**Worflow**

| **Schematics design** *Eeschema* | → | **Footprint assignement** *CvPcb* | → | **PCB design** *PcbNew* |
|---|---|---|---|---|

# KiCad | Eeschema

**Concept**

- Eeschema is a tool designed to draw electronic circuits schematics.
- The circuit schematic file purpose is to give a clear understanding of the circuit. It :
    - Displays all the elements
    - Displays all the connections between elements
    - Does not display the physical location of the components
    - Can be divided in functional blocs

**How does it work?**

- Add one symbol for each component
    - One from the library
    - Or your own symbol (there is a tool to create your own symbols)
- Draw the links between the components
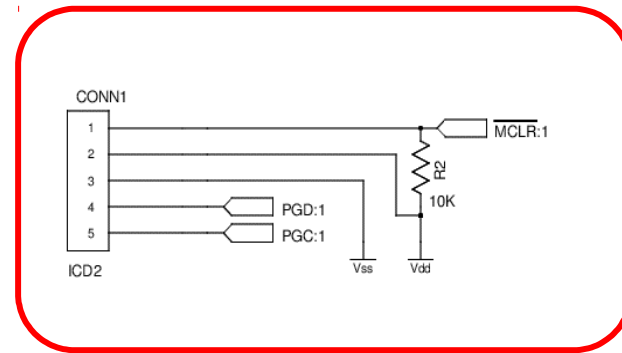- **Be sure that the pinout of your symbols correspond to the one given in the datasheet!**

# KiCad | Eeschema : Schematic example

- Example: Blinky



**Programming**

**Oscillator**

**Power supply**

# KiCad | CvPcb
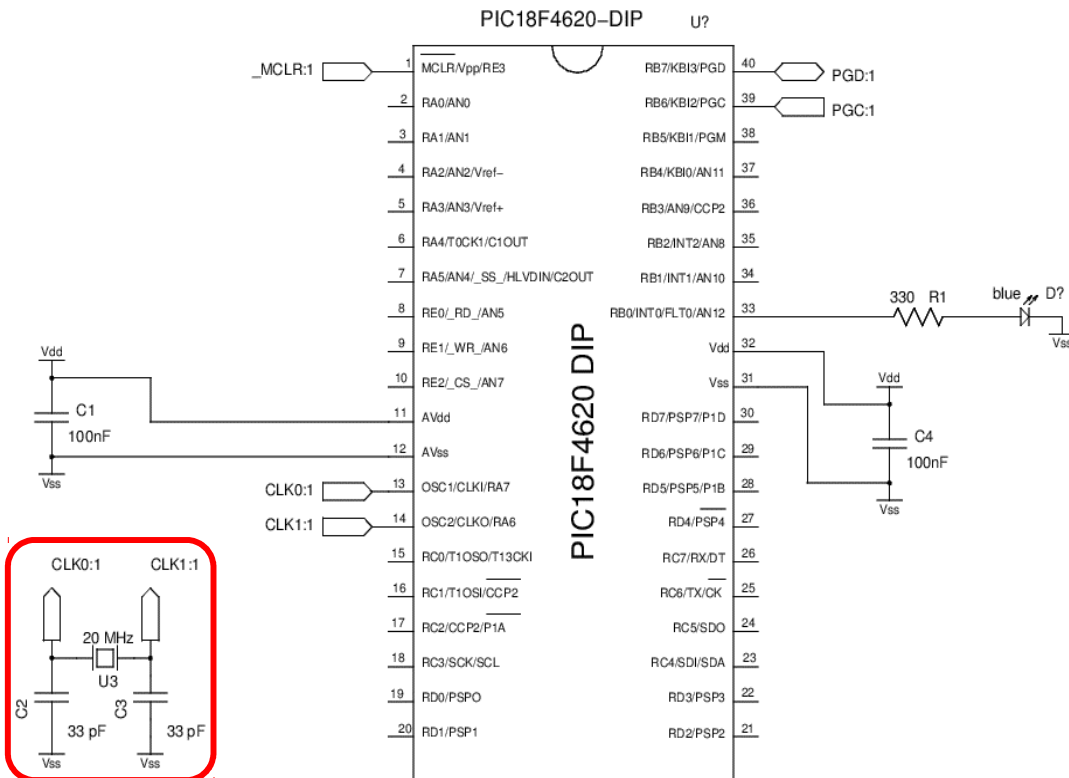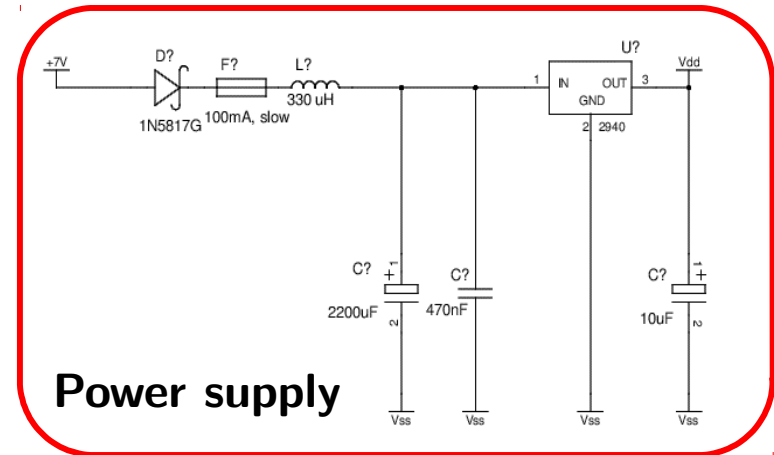
**Concept**

- The schematic doesn't tell anything about the components physical size and properties

  → Need to define the packaging (component size and pin locations) for each component

**How does it work?**

- Link one footprint to each component
    - One from the library
    - Or your own footprint (there is a tool to create your own one)
    - When editing or creating a footprint, refer to the datasheet for the specifications and verify your result with a 1:1 paper print
    - Assign the correct number to the pins/pads
    - Use the grid as a tool to correctly place the different parts of the footprint

# KiCad | PcbNew

**Concept**

- Assign the physical location of the components and tracks
- Commonly used unit : **mil**

$$1 \text{ mil} = 1/1000 \text{ inch} = 0.0254 \text{ mm}$$

**Best practise**

- Always write PCB & author name on the board
- Leave a 3mm spacing between circuit and board outline
- Hole in every corner (i.e. place the PCB on screws)
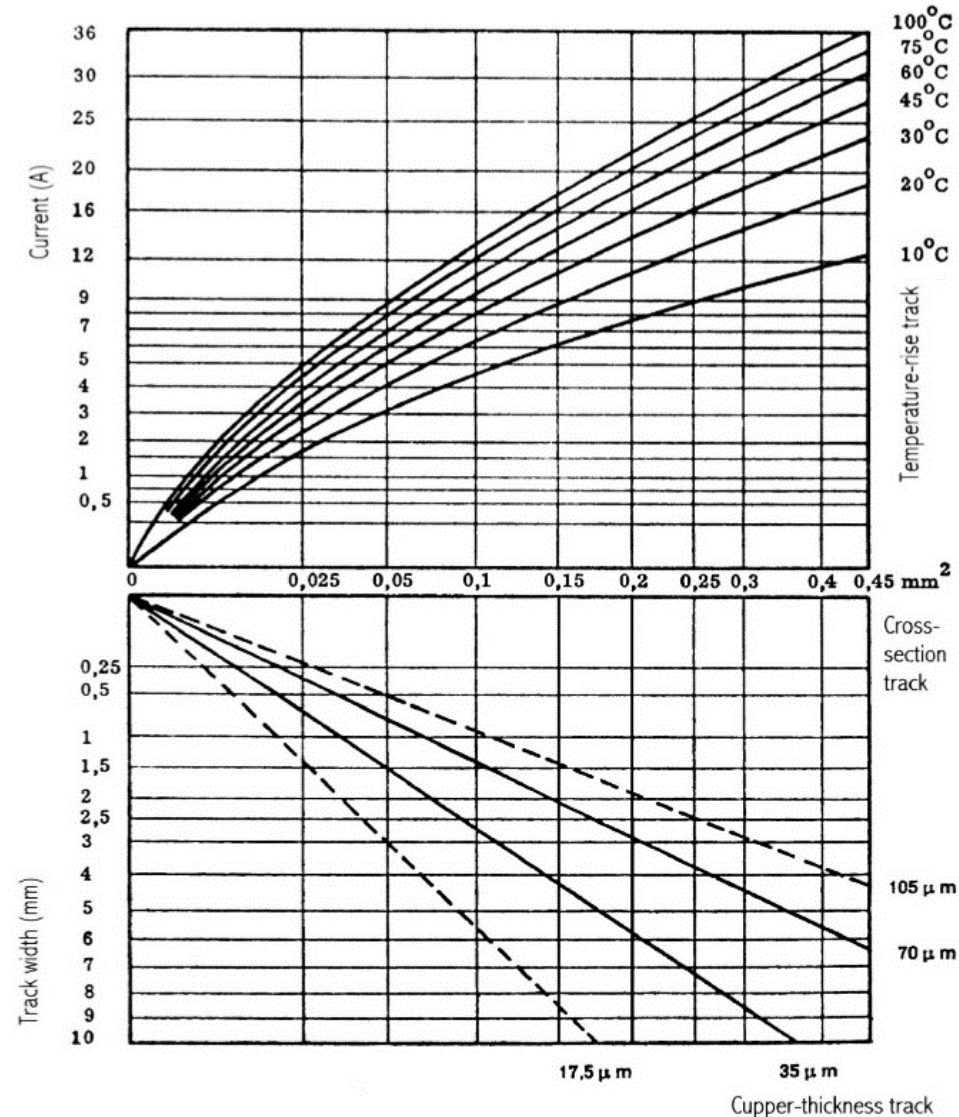- Decoupling capacitors have to be as close as possible to the component they protect



**NO**    **YES**    **YES**



**TO AVOID**

# PCB Tracks Size

**Concept**

- The width of a track **has to be adjusted** depending on the power it will transmit
- The rules depend on the manufacturer
- If printed at Montefiore [mil]:

  - Normal track width:  24 (min 16)

  - Power track width: 30

  - Via:  82x82 (min 70x70)

  - Via hole: 18

  - Clearance: 12

# PCB Design Tips

**Generalities**

- Space elements
- Place several LEDs or buttons for debugging
- Print a 1:1 paper version to check that dimensions are correct
- Do not forget the mirroring effect occuring when using through-hole components
- Beware of and avoid ground loops (I.e. currents going through the ground between two points)
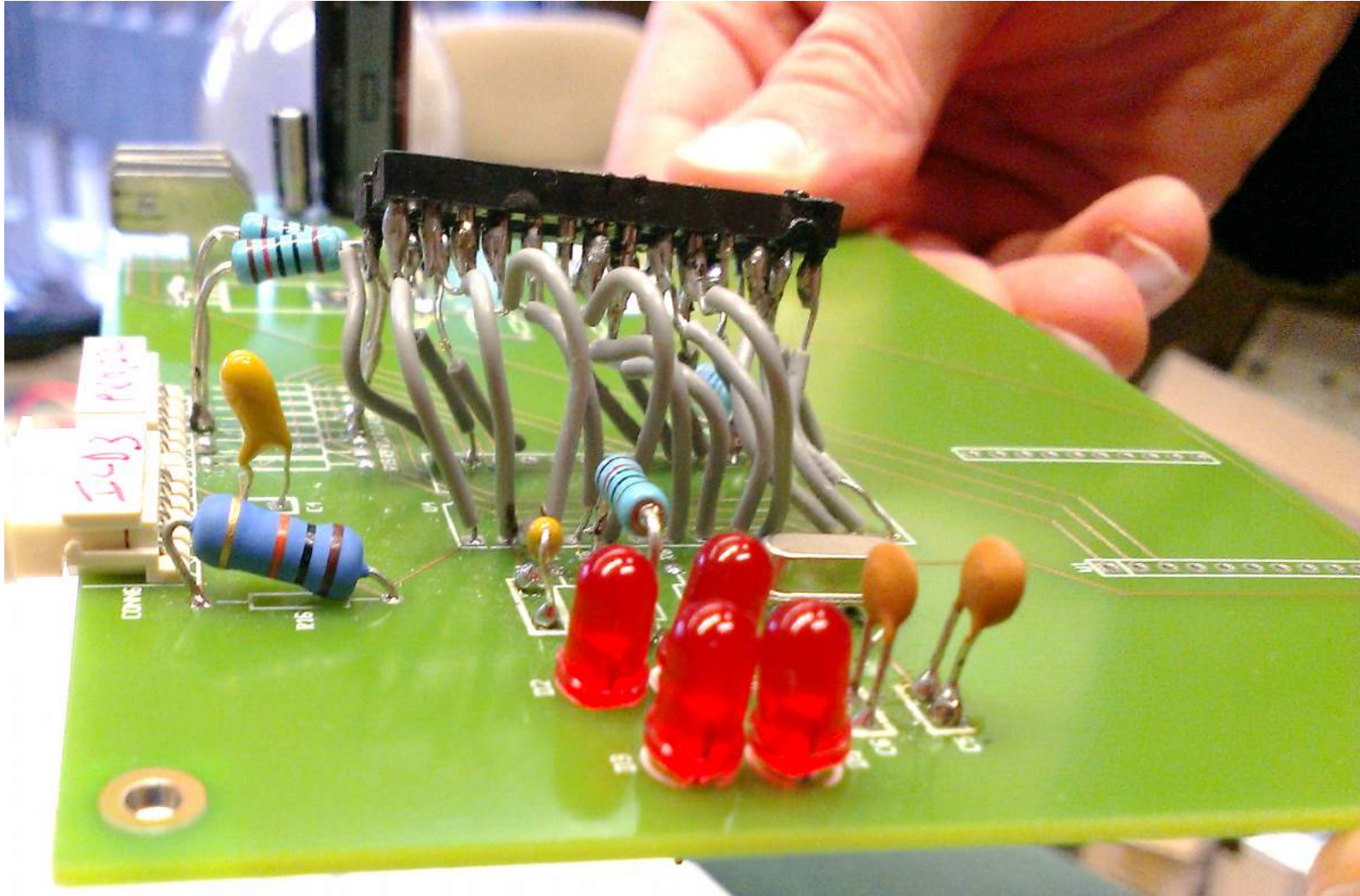- Use the ground star topology
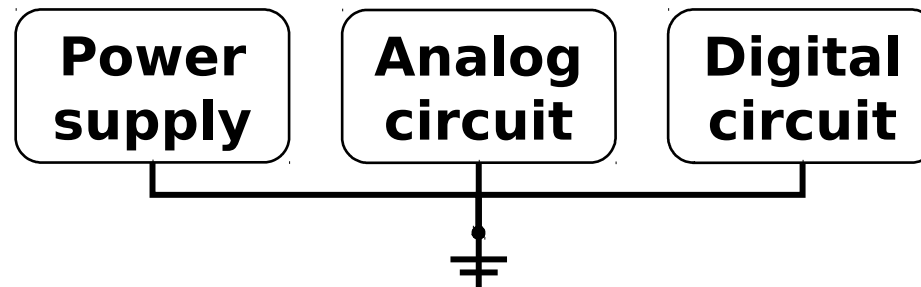


Ground star topology

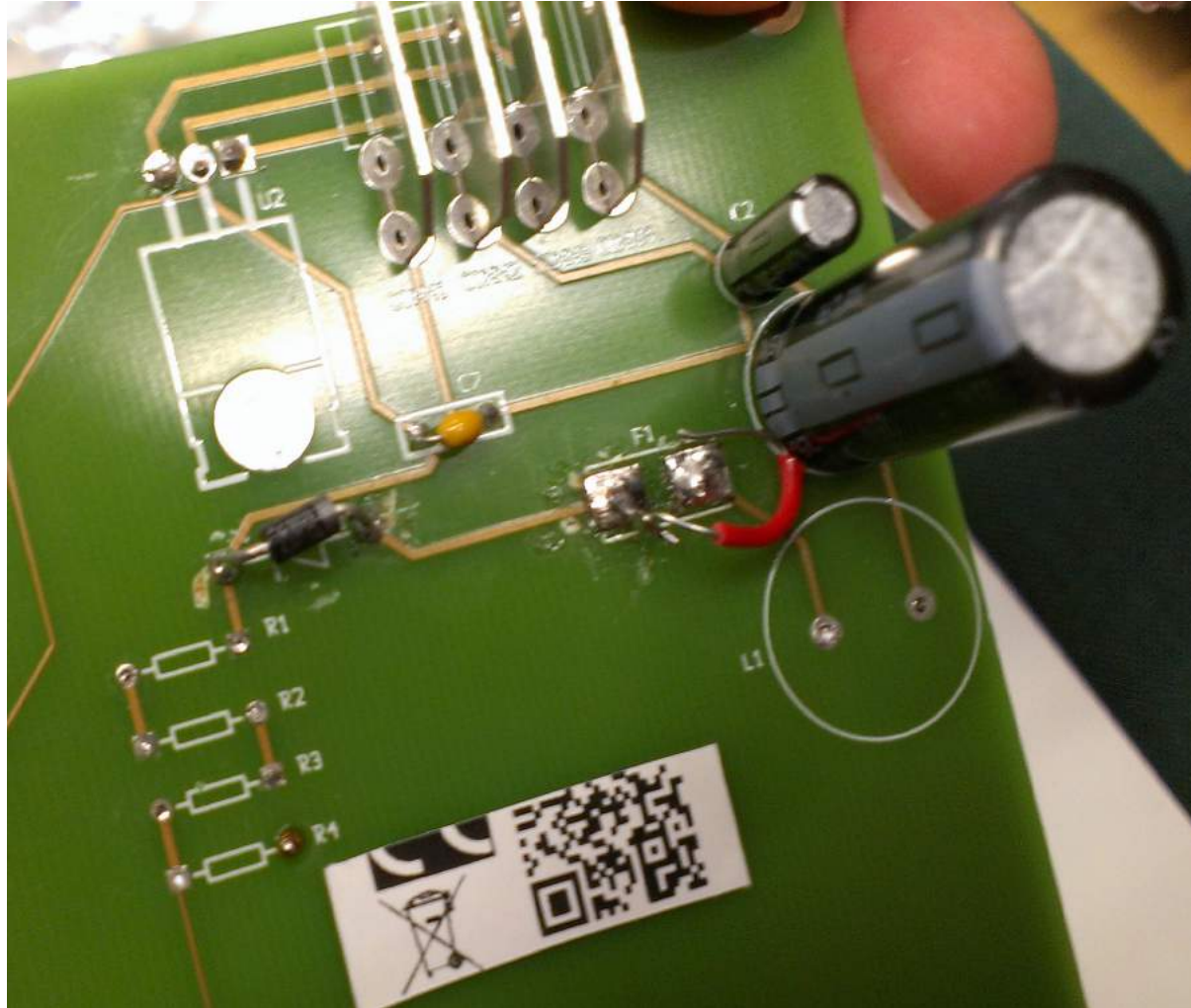**Do not forget : 1 error == a completely useless PCB**
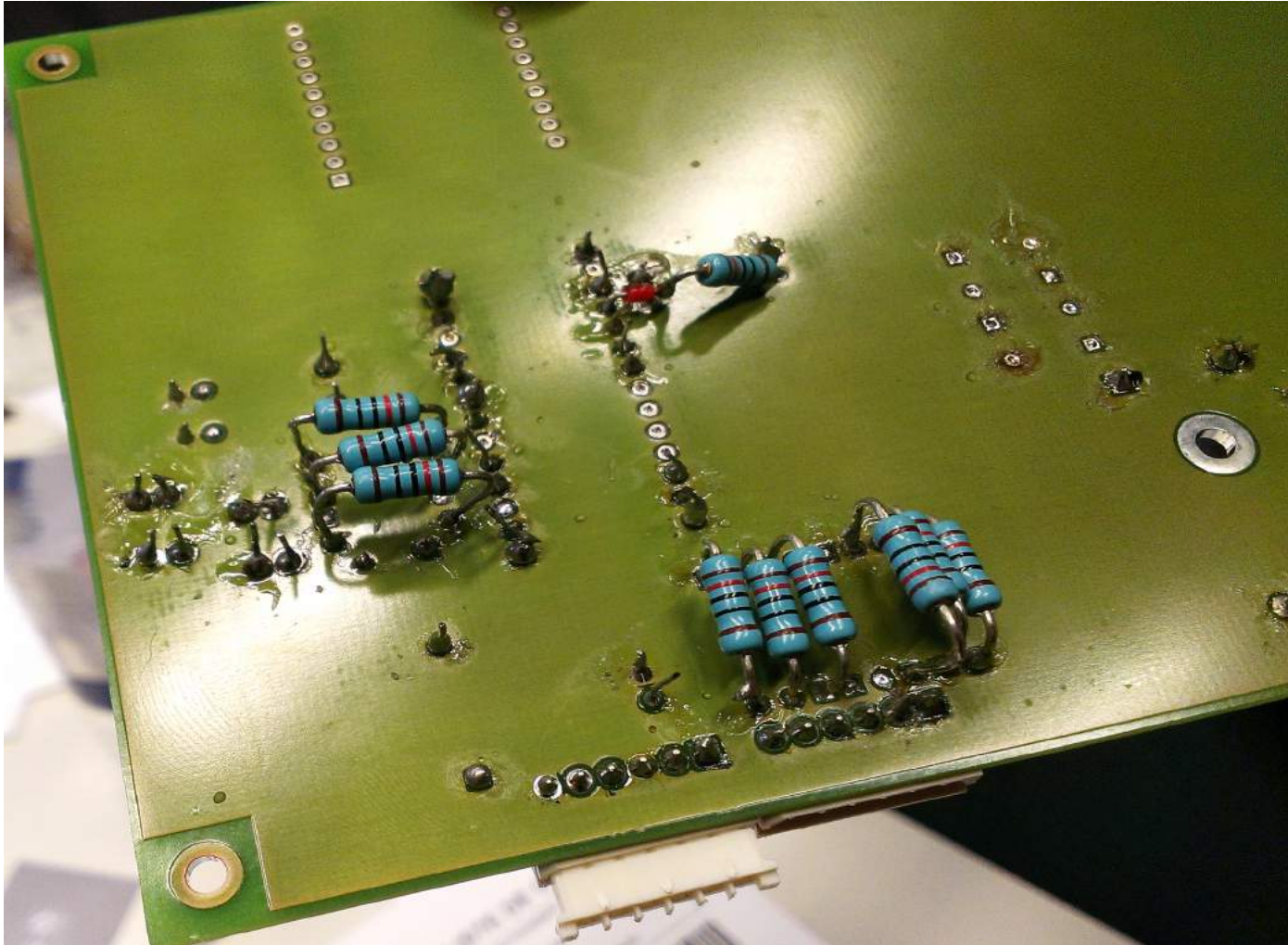
# Example of what to not do
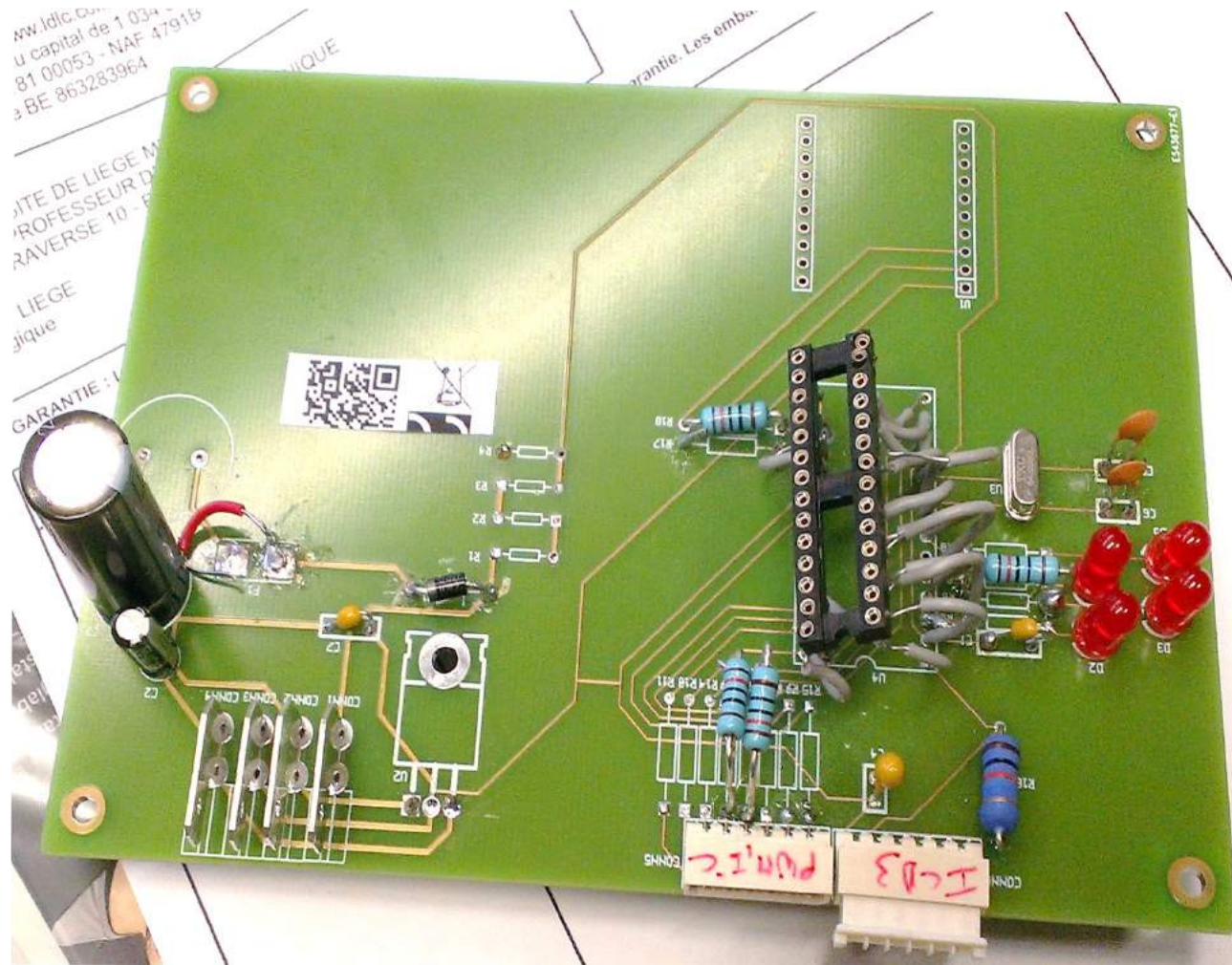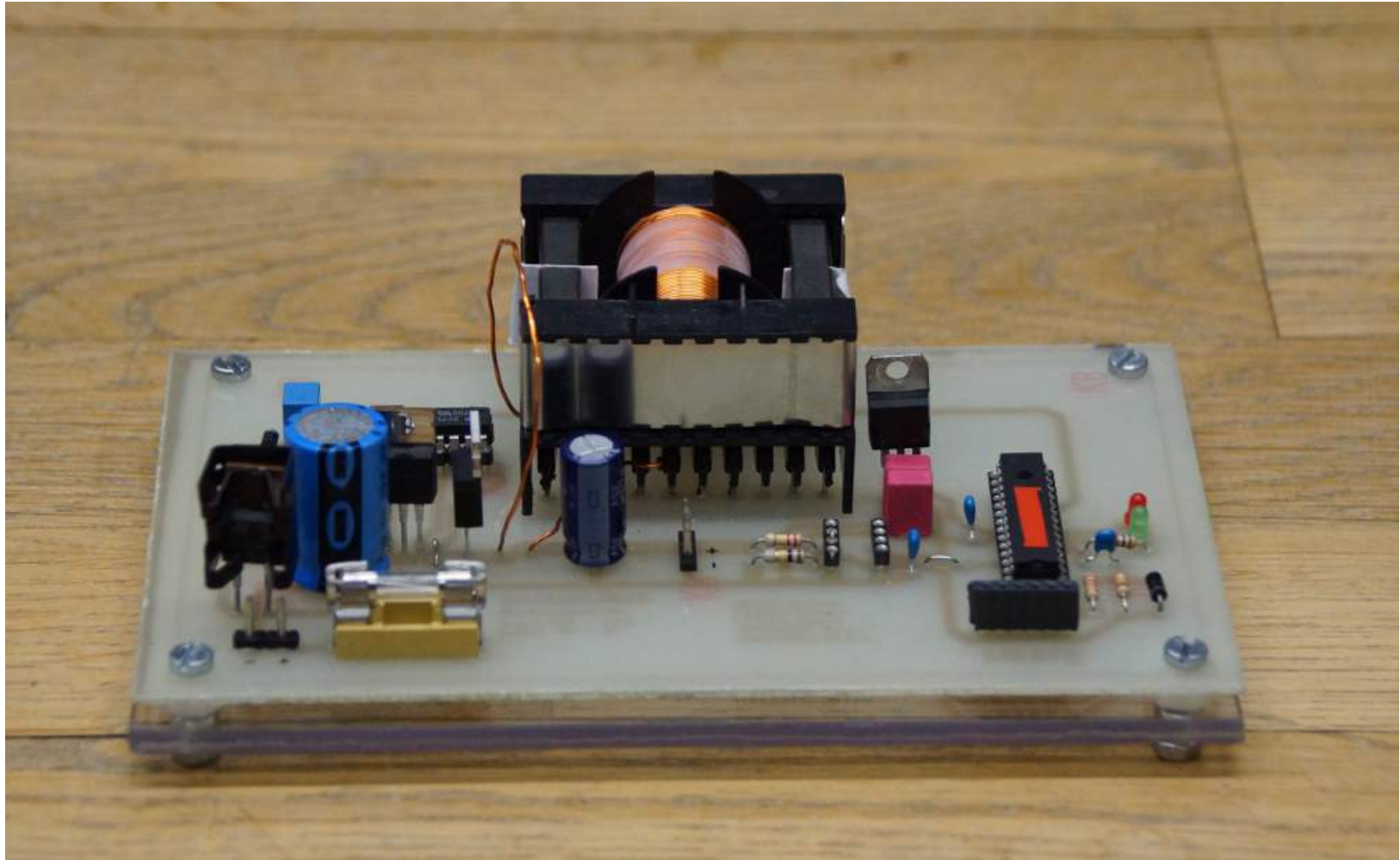
# Example of what to not do
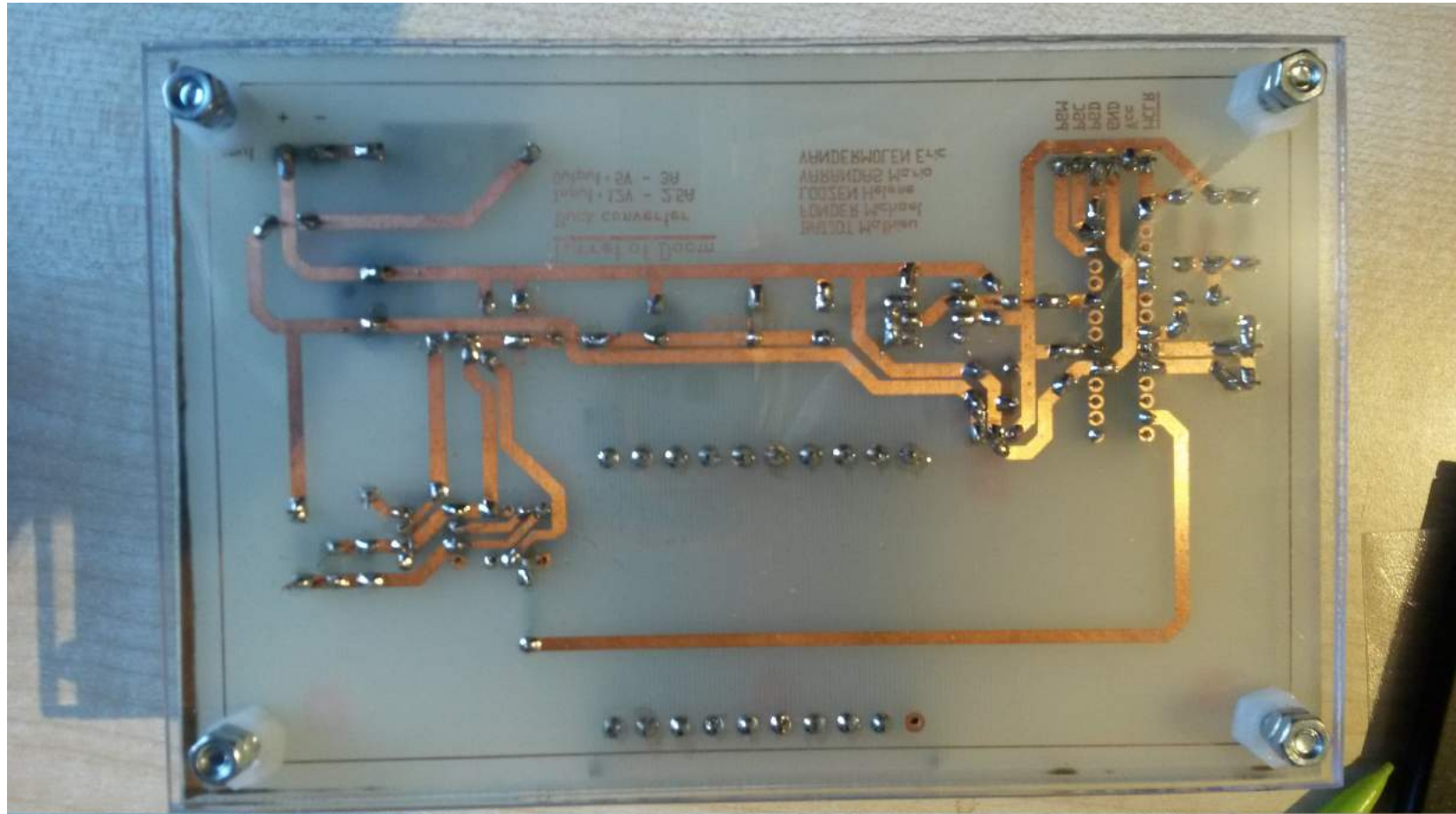
# Example of what to not do

# Example of what to not do

# Example of ok design

# Example of ok design

# Debugging procedure

**Trust no one!**

- Keep calm
- Use the oscilloscope
- Check **elements connections & pinout**
- Check soldering
- Check the power voltage
- Check pin voltage
- Process step-by-step and repeat this procedure