APPLIED CIRCUIT AND PCB DESIGN

WITH KICAD

HANDBOOK

Contents

Controlled collision between 2 (or more) components 2

Net Classes on schematic 4

Set up the board stackup 5

Set up the design rules (constraints) 7

Adjust Board's shape and size 8

Import images as graphs 9

Hierarchical design: represent and replicate schematics blocks 12

Replicate layout blocks 13

Tent vias 15

Power supply layouting 16

Power distribution 17

Use of sheet templates 20

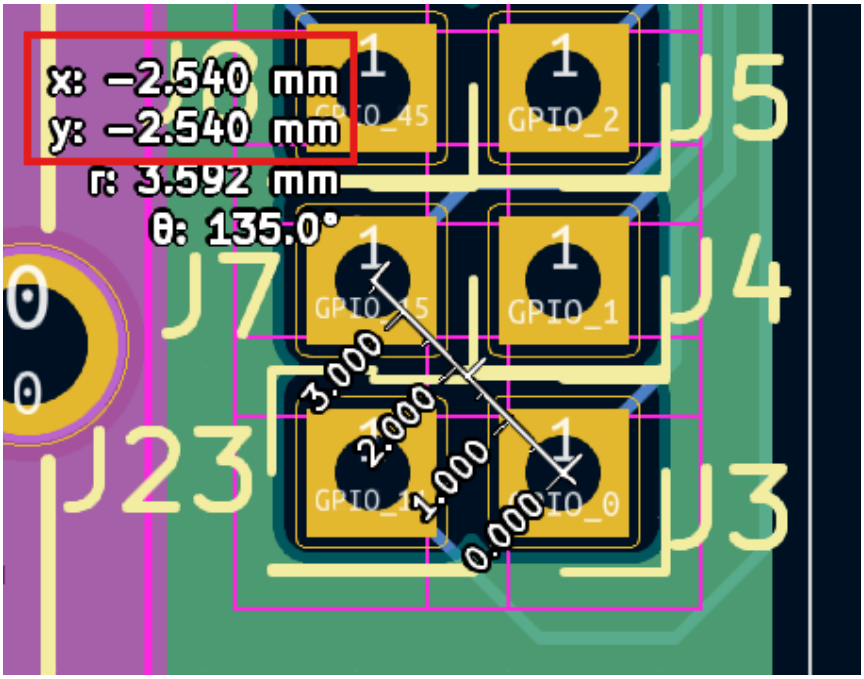
Differential pairs routing 21

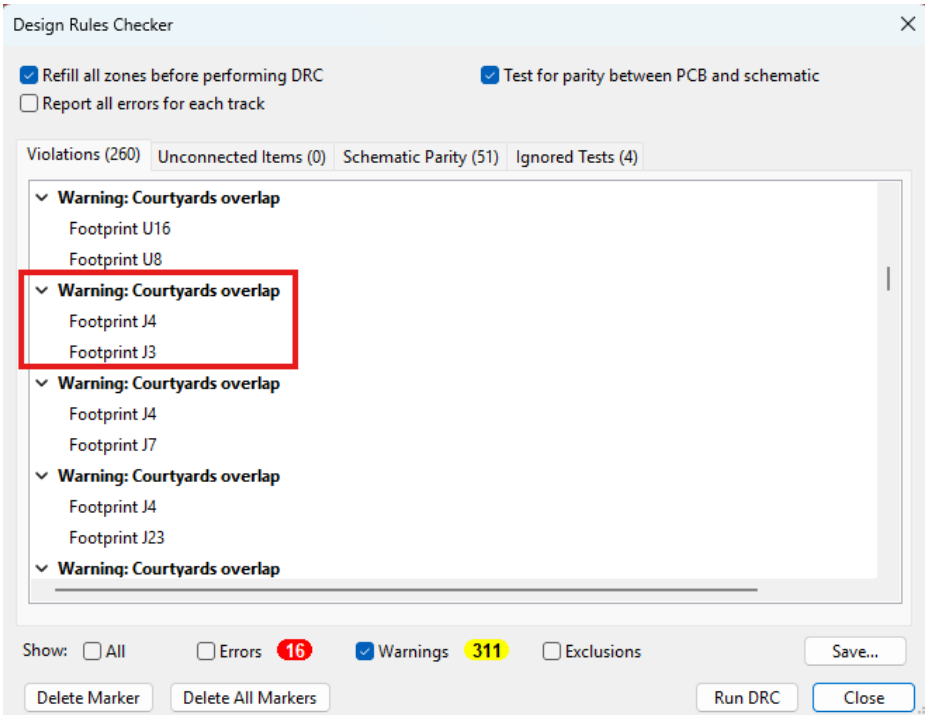
Controlled collision between 2 (or more) components

Problem

In some specific cases, two (or more) components in our design need to be placed in such a way that they collide. This will create a warning during the ***Design Rule Check*** process, but it happens to be a controlled one, since we know that collision is needed.

A clear example of this is the placement of individual pin headers to form a longer pin strip. The pitch between every two pins must be set to 2.54mm (100mils). This leads to a *controlled* violation of the ***Courtyards overlap*** rule:

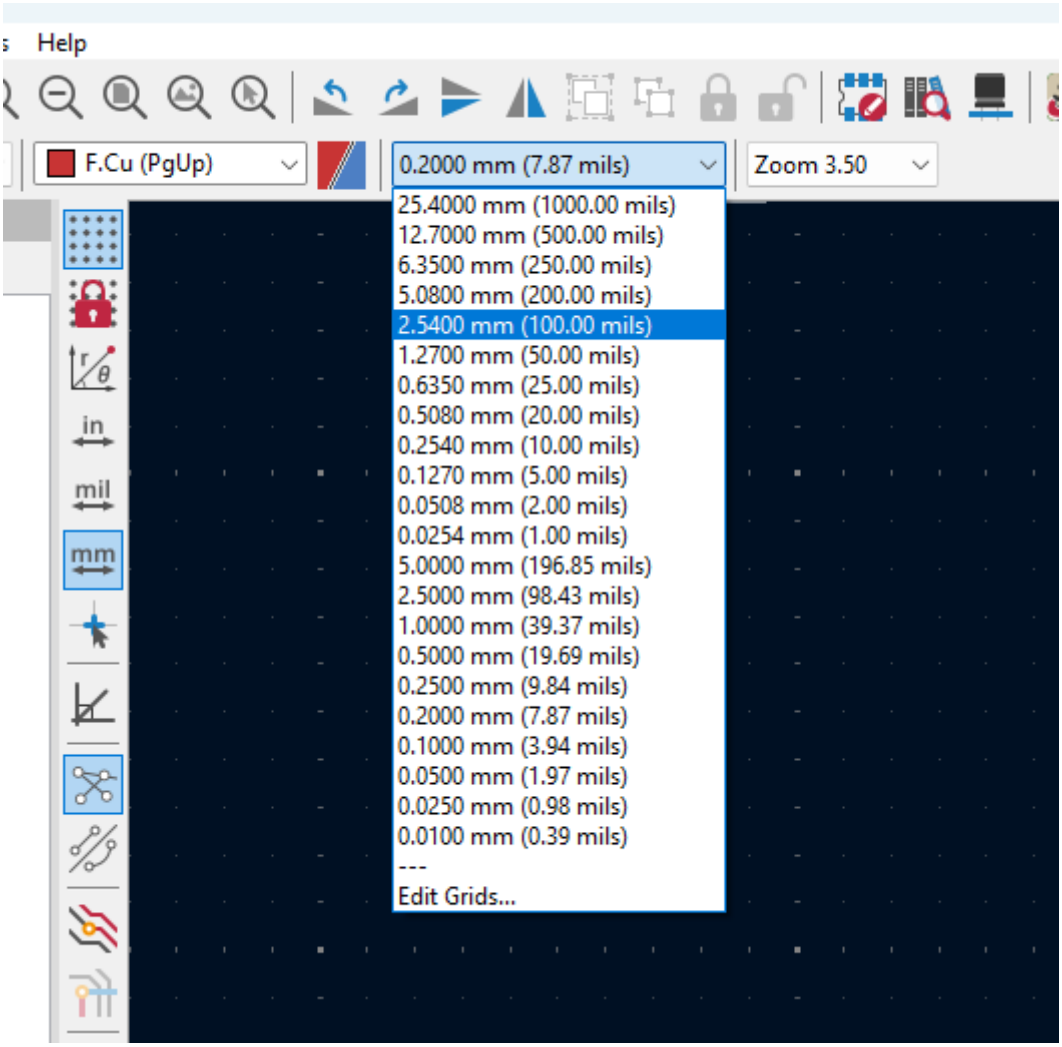




Solution

As a first step, we need to learn how to place any two components at a given distance between them. As said above, we need to align the individual pin headers so that their distance center-to-center is 2.54mm. To get this done, we are going to play with the design grid.

Notice that 2.54 mm equal 100 mils. By changing the design grid to 100 mils we will force our mouse to snap in steps of 100 mils:

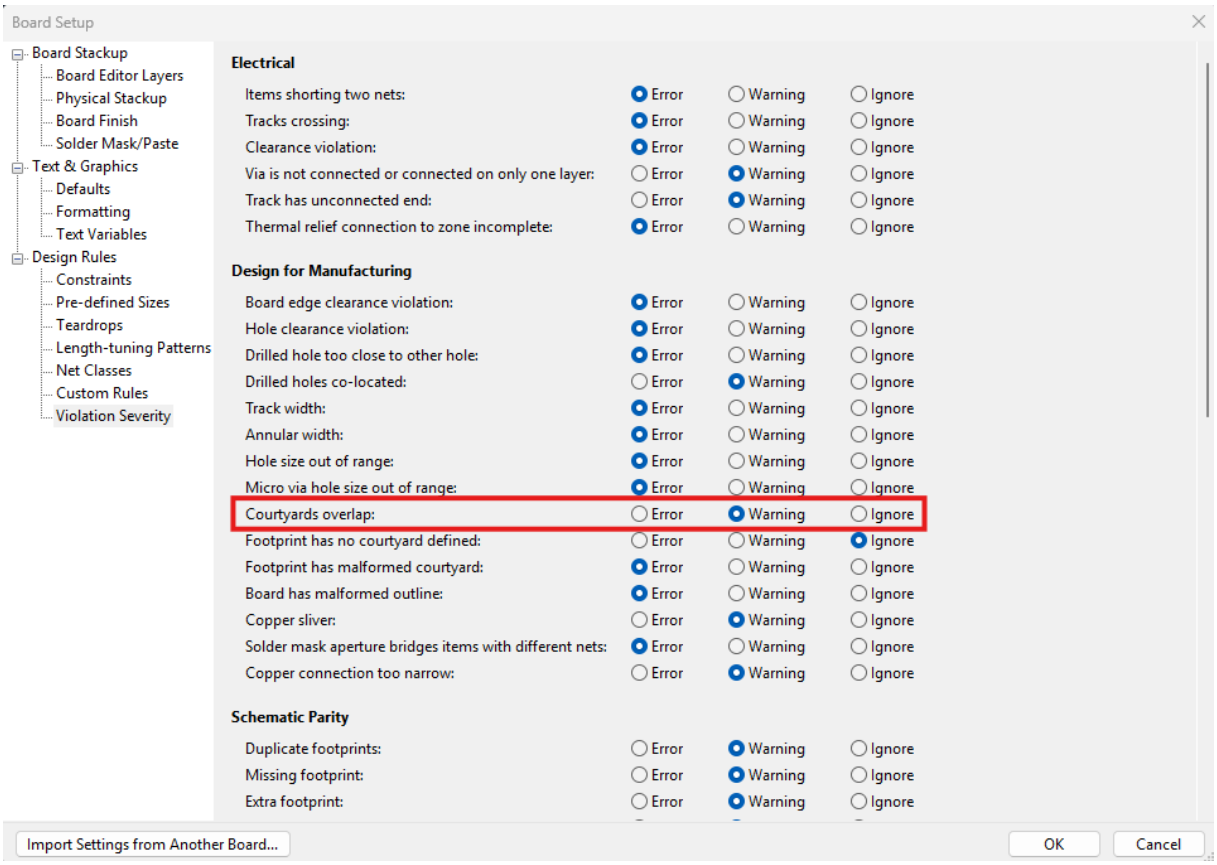


You need now to drag one of the pin headers and choose a location for it. Drag now the next one and place it adjacent to the first one in the next grid point. Repeat these actions as many times as pin headers you need to align.

Check with the ***Measurer Tool*** (***Ctrl+Shift+M***) that the pitch is, as intended, 2.54 mm.

Once you are done, change the grid back to its previous (metric) value.

Remember you can control the ***rules violation severity*** at any moment under ***Board Setup***. See that, by default, the ***Courtyards overlap*** rule is set to ***Warning***:



You would need to change it to ***Ignore*** to get rid of this type of violations.

Net Classes on schematic

Problem

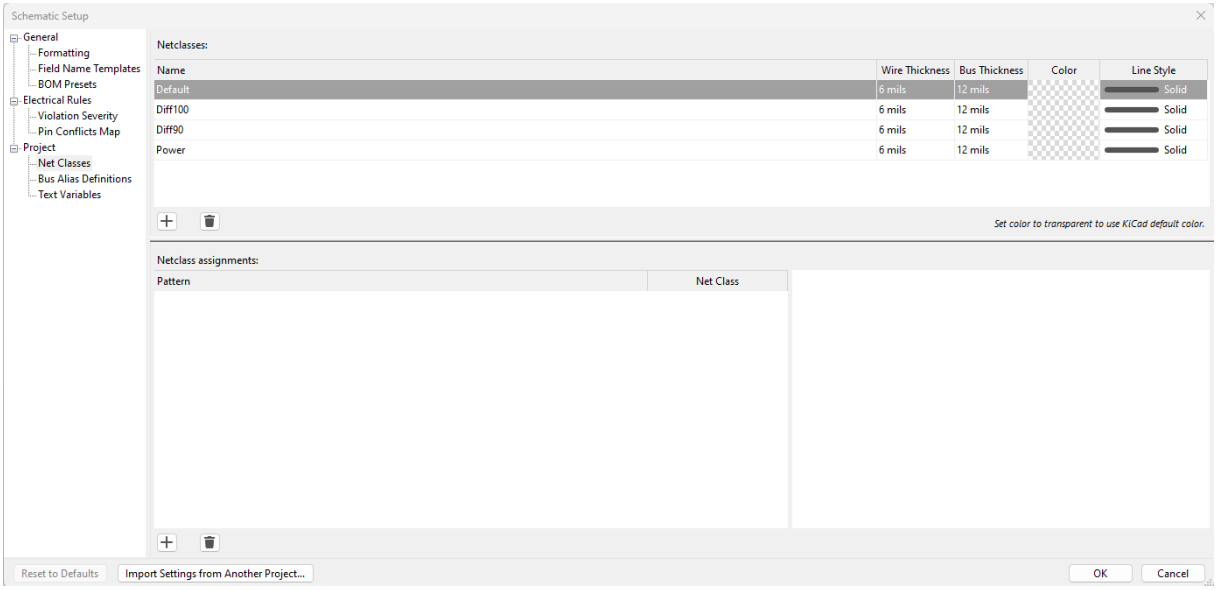
Often, it is desirable to group together signals with similar characteristics so that we can add design rules to them.

One example could be the minimum width for a given net: power nets carrying high currents need to keep a minimum width.

Solution

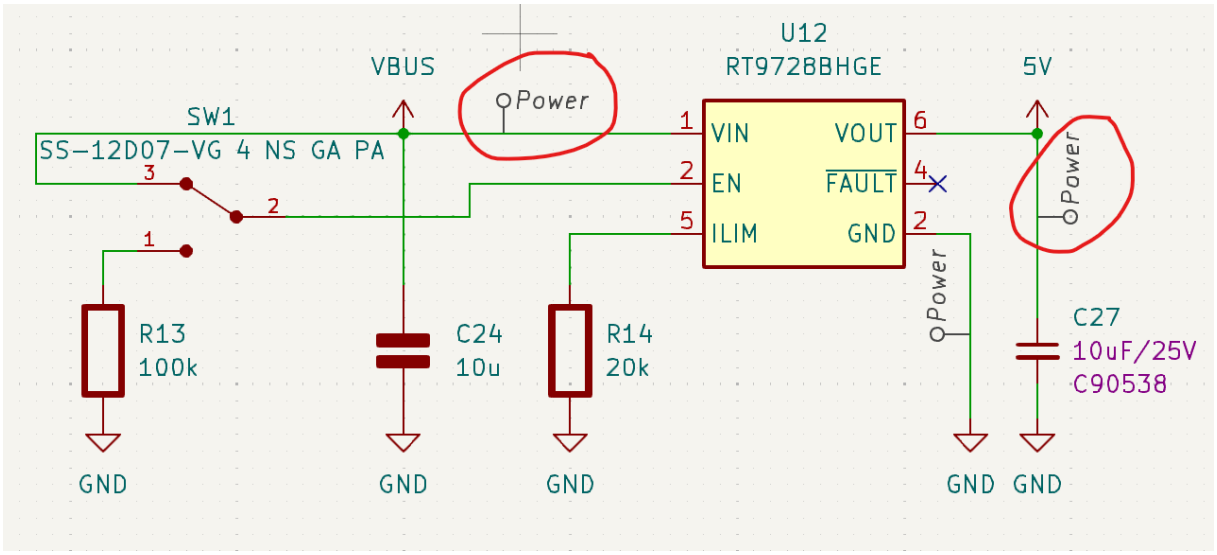
While on the schematic editor, go to ***File→Schematic Setup***. Then, select under the menu ***Project***, ***Net Classes***. Observe that the only ***Net Class*** is the ***Default*** one, which comprises all the signals not assigned to any other ***Net Class***.

Let us now add 3 net classes more. Use for that the ***+*** button until you get this:



Press ***OK*** once done.

We need now to assign the desired nets to the newly created net classes. For that, we will use the command ***Place→Add Net Class Directive***. In the opened window, type under ***Value*** the desired ***Net Class*** name (***Power, Diff100 or Diff90***). Press ***OK*** and attach now the directive to the required wire. An example would be this:



Remember you can rotate the directive with the shortcut ***R***.

Place the directives on all the signals as per the master schematic.

Set up the board stackup

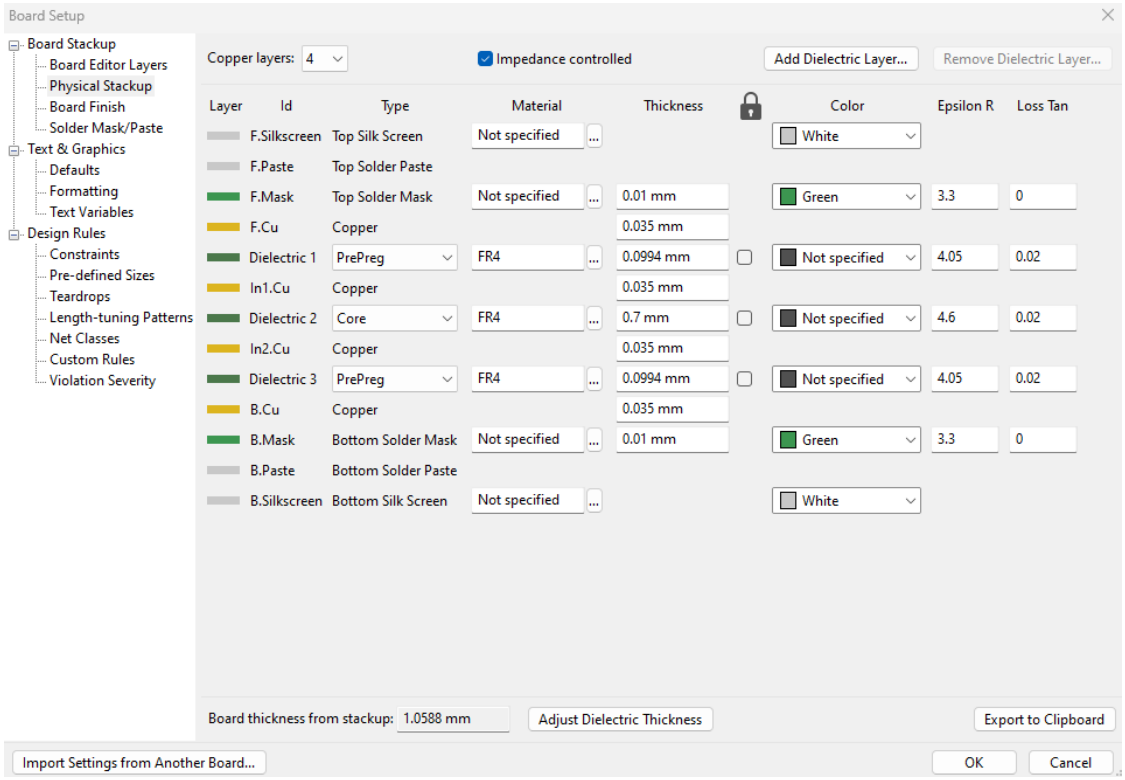
Problem

One of the most crucial steps in the whole design process is the choice of an appropriate combination of materials that will make up the PCB. Its impact on the board performance, cost, and easiness of routing is paramount.

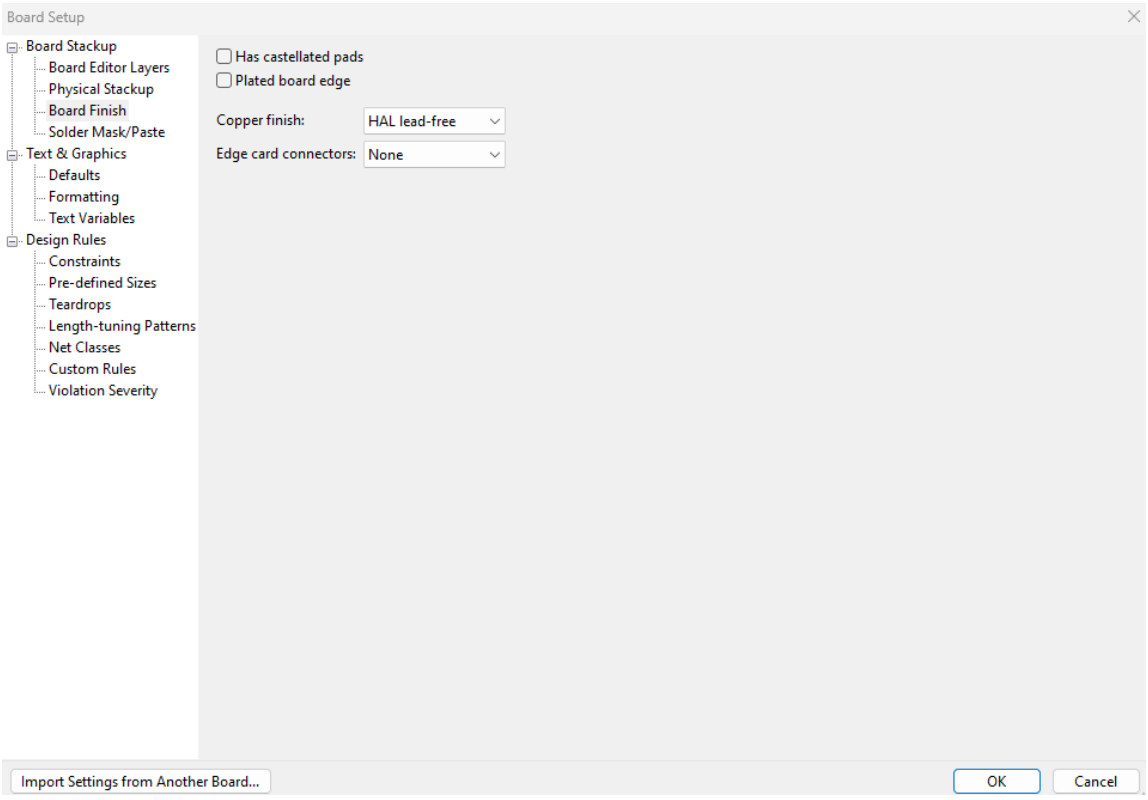
Solution

The stackup selection for this project is given to you as a design premise.

Access the ***Board Setup*** menu under ***File***. Then go to the ***Physical Stackup*** under the ***Board Stackup*** menu. We will be using four copper layers in this fashion:



In the same way, select as ***Copper finish HAL lead-free*** (under ***Board Finish***):



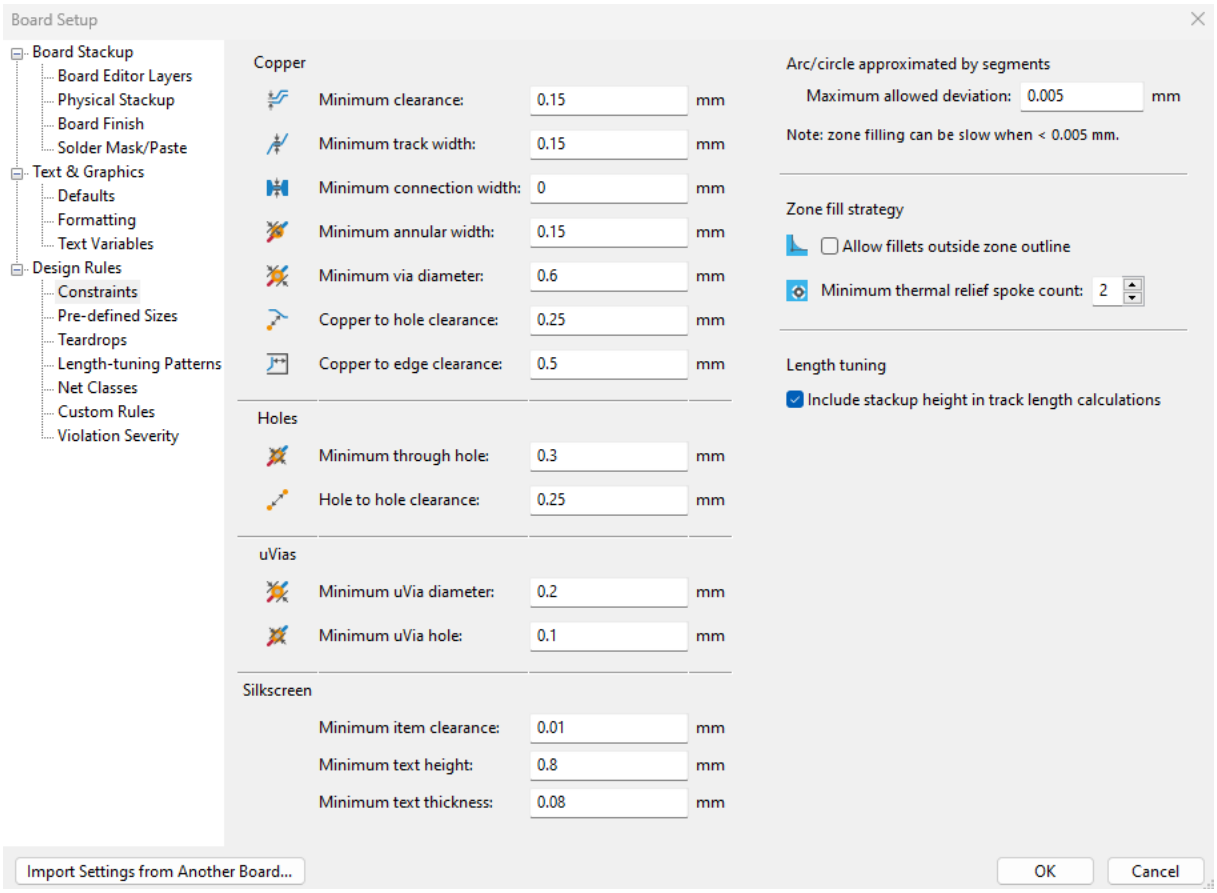
Set up the design rules (constraints)

Problem

Manufacturers have technology limits. They cannot just manufacture any design. We need therefore to stick to some rules for the sake of manufacturability and cost.

Solution

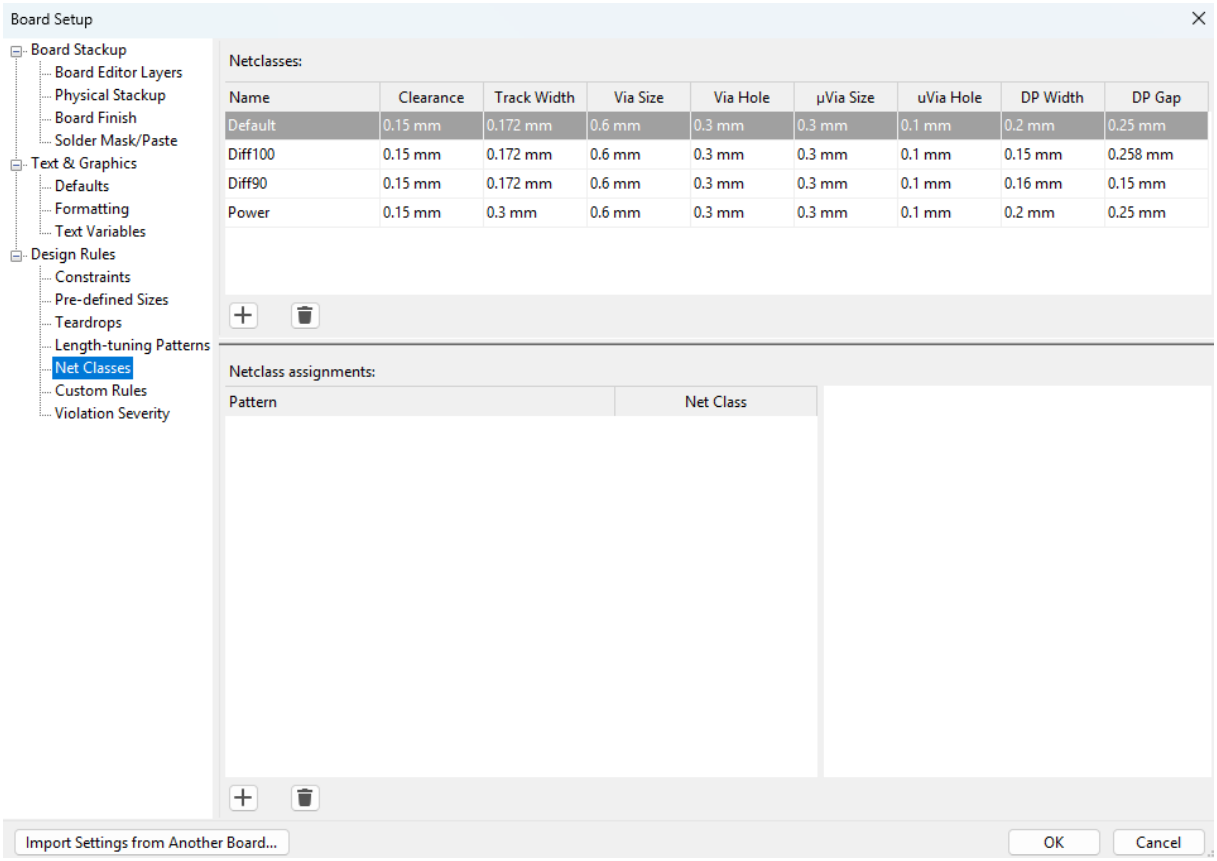
While on the PCB Editor, go to ***File→Board Setup***. Access now the menu ***Constraints*** under ***Design Rules***. Fill up the fields so that they look like this:



We need now to set the rule values for each specific net class. In the same ***Board Setup*** window, access now the ***Net Classes*** menu.

Note that the net classes that we created in the schema have been transferred to the layout.

Adjust the values as follows:



The width and gap values are chosen so that we get 50 ohms single-ended, 90 ohms differential or 100 ohms differential respectively.

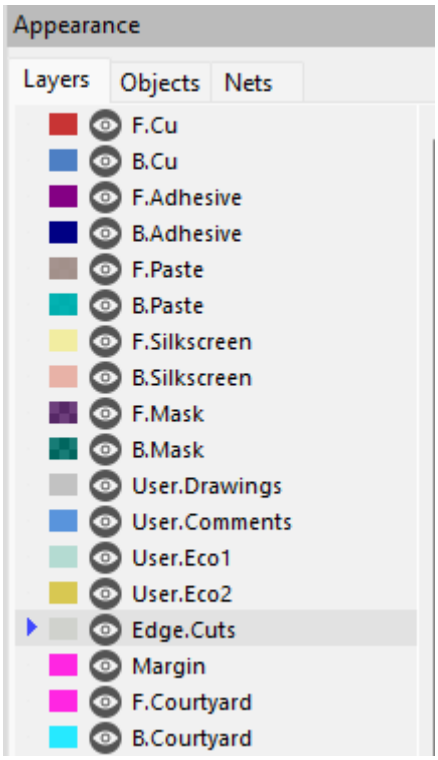
Adjust Board's shape and size

Problem

After finishing the schematics entry, we need to define a PCB board. The maximum size you are allowed to use is 90 cm x 70 cm.

Solution

While on the PCB Editor, click on the right-hand panel on the layer ***Edge.Cuts***. The board shape is defined on that layer in KiCAD.



Go to Place→Draw Rectangle. Click one time to set the starting point and move the mouse away and click a second time. Then ***right-click*** and ***Cancel***. The rectangle is now drawn, but with wrong dimensions. Let us adjust them.

Click on the rectangle and then press the shortcut ***E***. That opens the object's properties. Adjust the ***Height*** and ***Width*** to 70 and 90 mm respectively and press OK.

The rectangle (our board) has now the right dimensions.

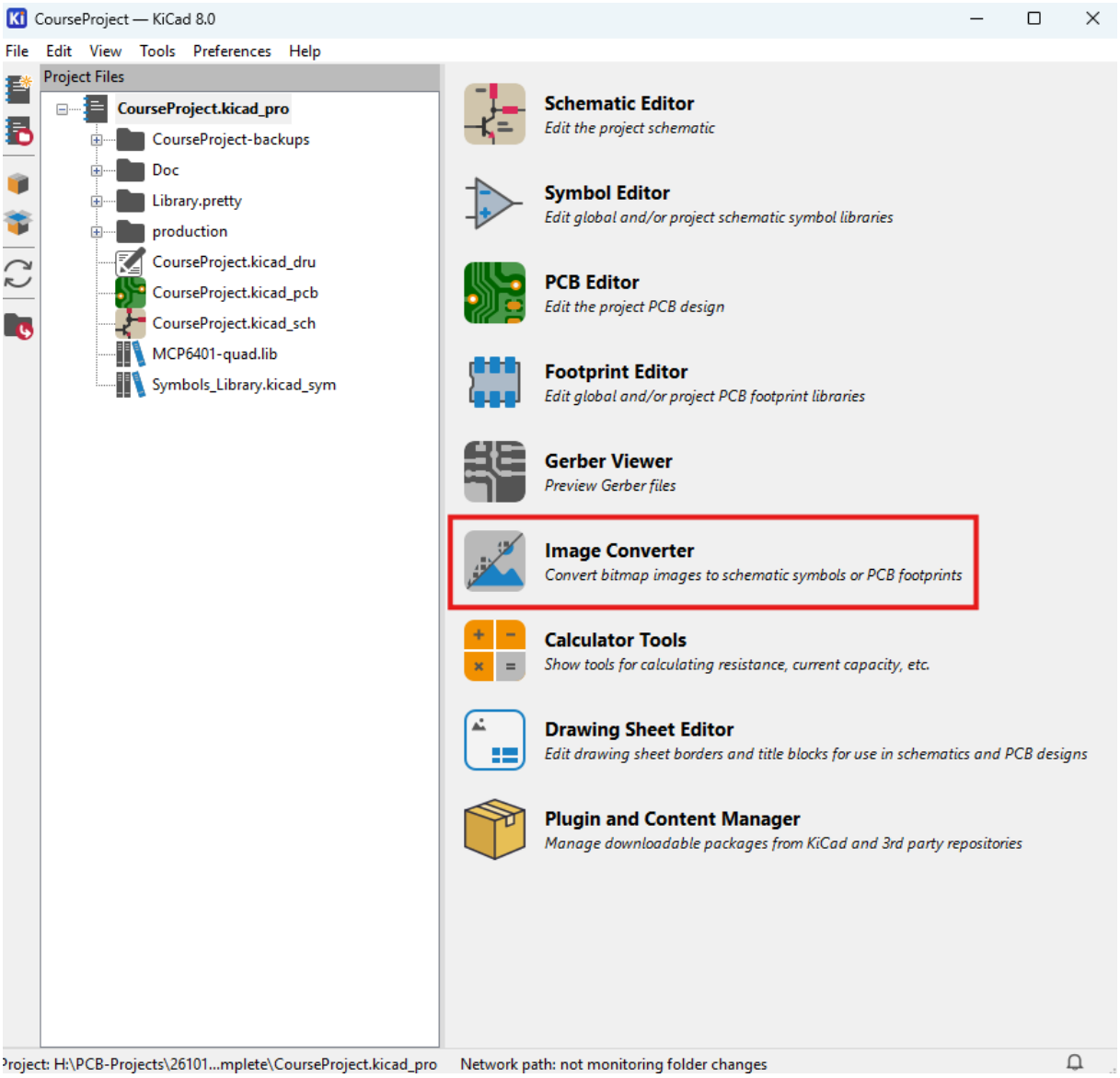
Import images as graphs

Problem

Graphic elements make designs look more professional and corporative, besides giving you free space for your creativity.

Solution

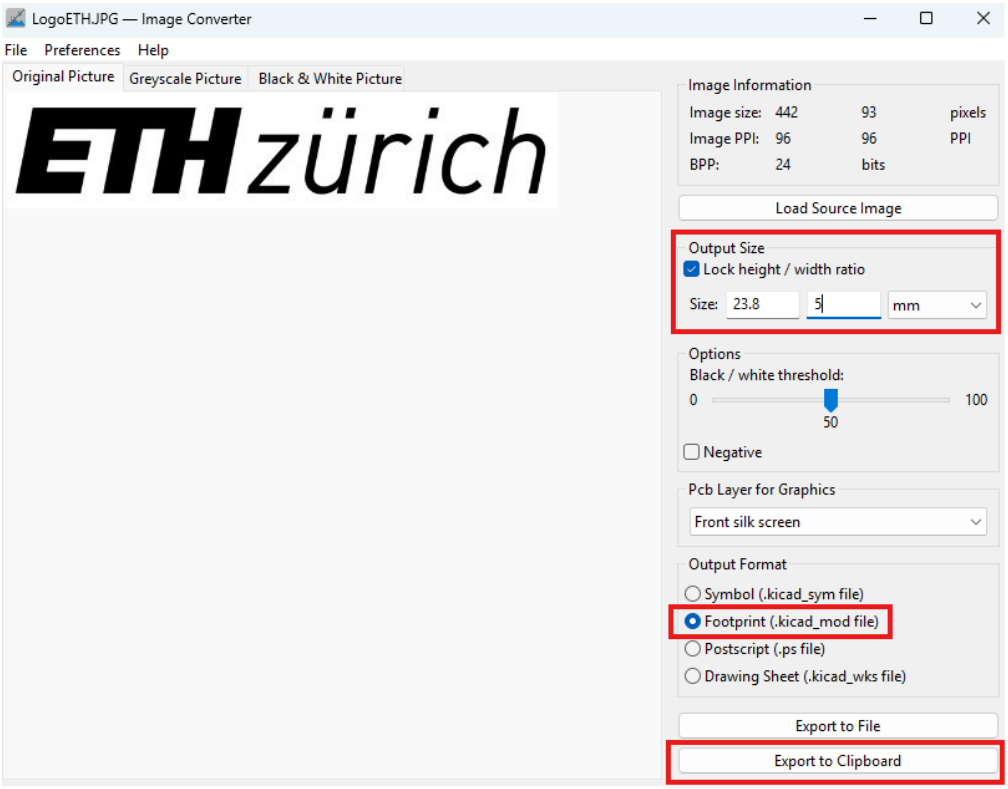
In the initial ***KiCAD*** window, choose ***Image Converter***.



Go to File→Open and select the image file you'd like to bring onto your board.

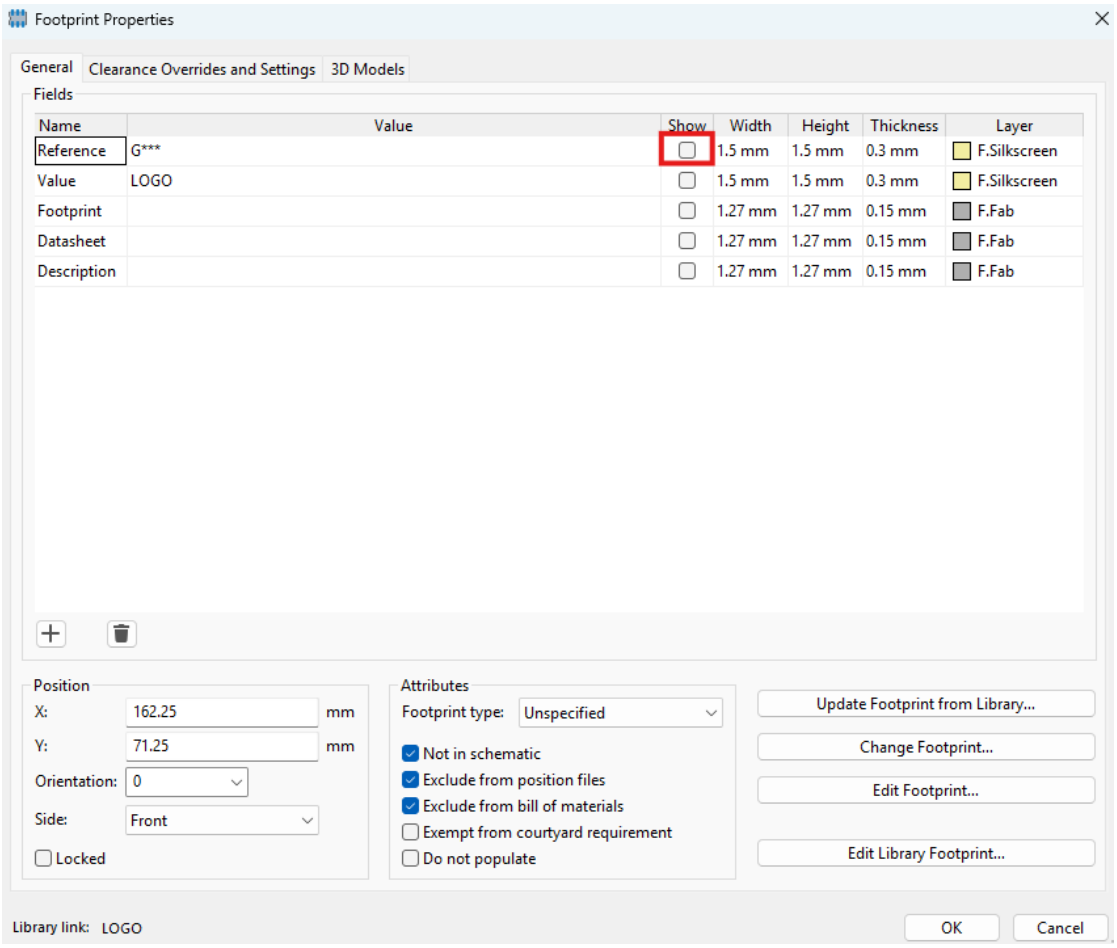
Play around with the settings until you get the image you like. Most important: set the output size to something that fits in the space you have available on your PCB.

Select as ***Output Format Footprint (.kicad_mod file)***. Finally, click on ***Export to Clipboard***.



Open now the ***PCB Editor*** to load your ***PCB*** design. Right-click and choose ***Paste***.

Double-click on the image to edit its properties and hide its designator:



Hierarchical design: represent and replicate schematics blocks

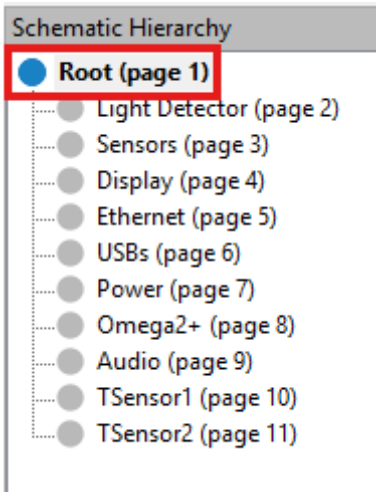
Problem

Presenting the schematics in a hierarchical way helps us to easily understand its functionality. Besides, it allows us to reuse schematic blocks without the need to redraw them. ***KiCAD*** offers us the possibility of getting this done in a comprehensive way.

Solution

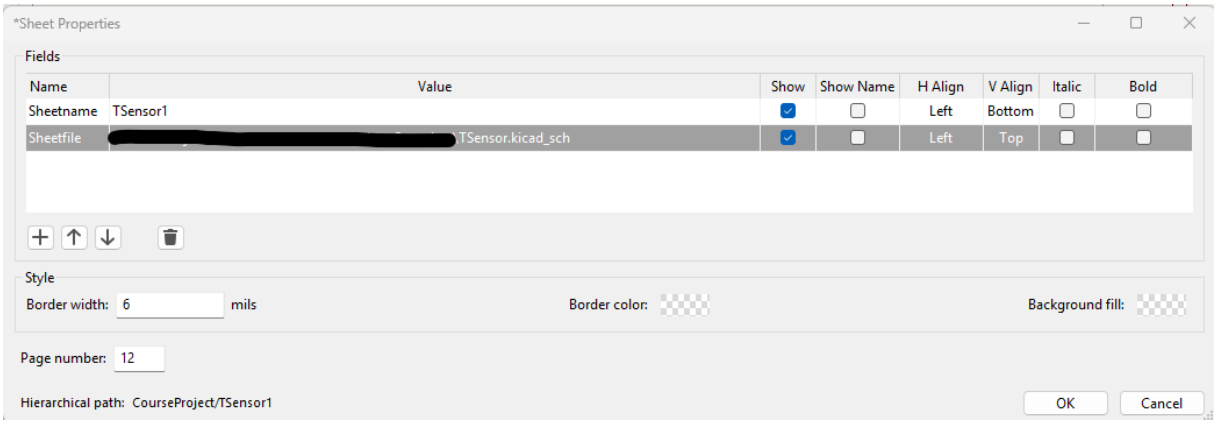
Let's take as example the analog ***Temperature Sensor***. We have two identical temperature sensors. We only need to draw it once though. That would be actually the first step: capture the schematic on a unique schematic sheet.

Now, go to the ***Top Level*** sheet of you design (saved as ***CourseProject.kicad_sch***) and displayed as ***Root***.



There, we have a hierarchical representation of our circuit by means of blocks. We create now a block linked to the ***Temperature Sensor*** sheet. Go to ***Place→Add Sheet*** and click twice in the design area to draw a rectangle.

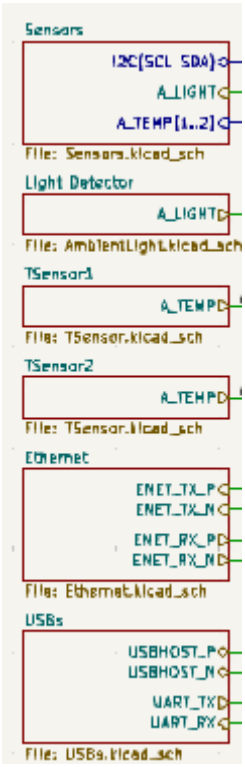
A window will open. Assign a ***Sheetname*** and link it to the schematic sheet you intend to replicate:



Right-click and ***Cancel*** to end the command.

Right-click now on the newly created block and choose ***Import Sheet Pin***. Note that ***KiCAD*** is bringing the ***Hierarchical Labels*** defined in the daughter schematic sheet. Click somewhere to fix them. You can rearrange them if needed.

To replicate any block, repeat the steps above. In the end, you'll get something like this:



You can now add the wires between the blocks to complete the connectivity.

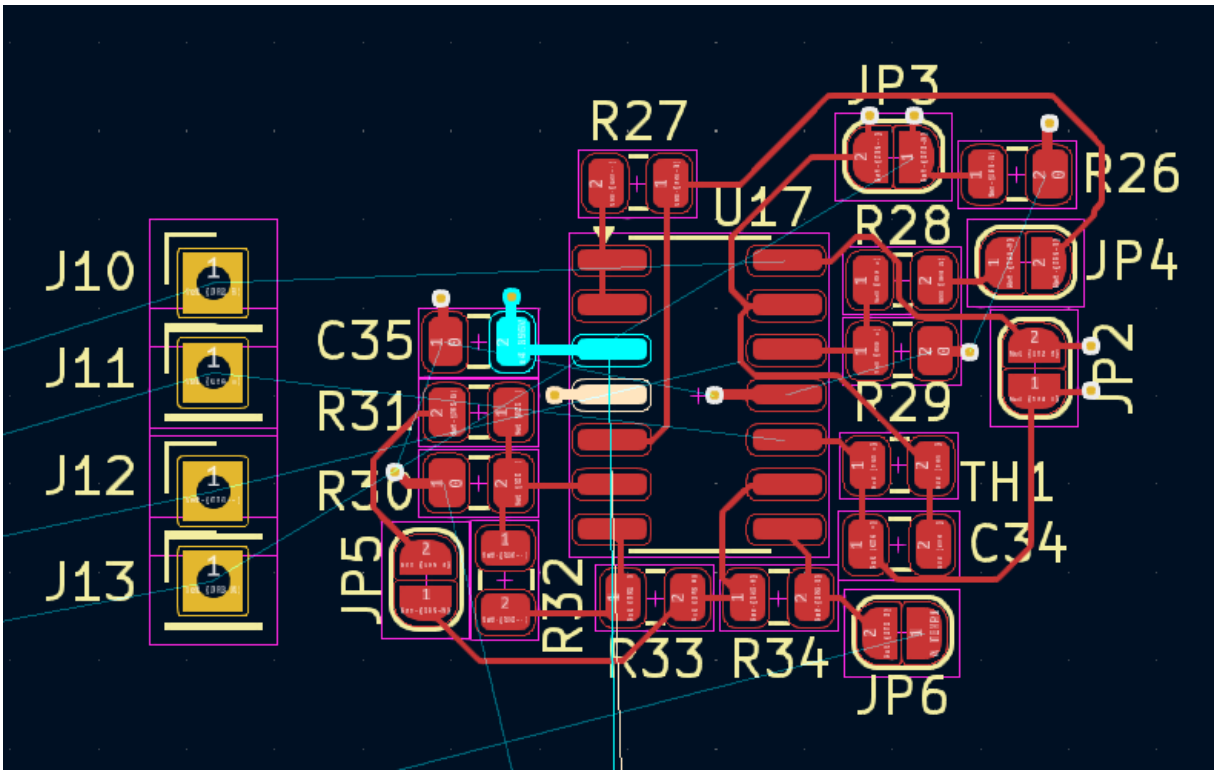
Replicate layout blocks

Problem

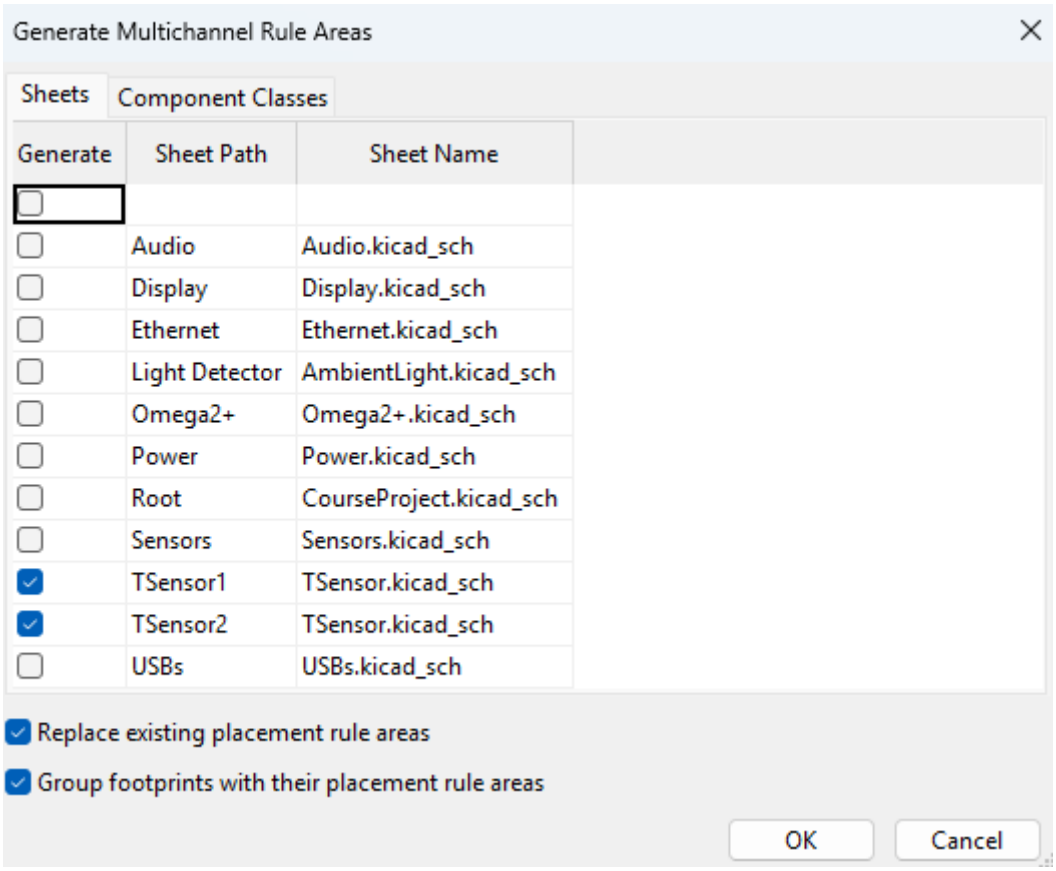
As a consequence derived from the previous point, we can somehow infer that there should also be a way to replicate the layout blocks linked to replicated schematics blocks.

Solution

In order to copy one sub-channel layout to another one, we need, as a first step, to place and route that first sub-channel.

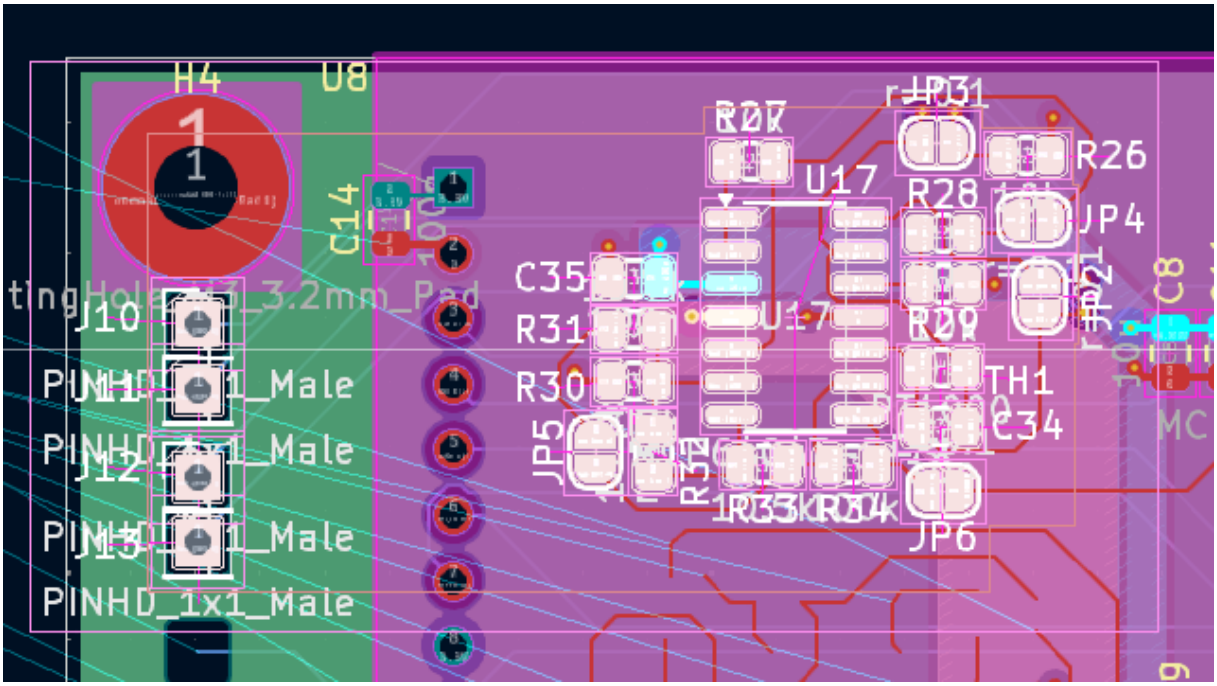


To transfer that same layout to the second (or third or n) channel, select one of the footprints part of the sub-channel and then go to ***Tools→Multi-Channel→Generate Placement Rule Areas…***. Choose now what elements you want to replicate:



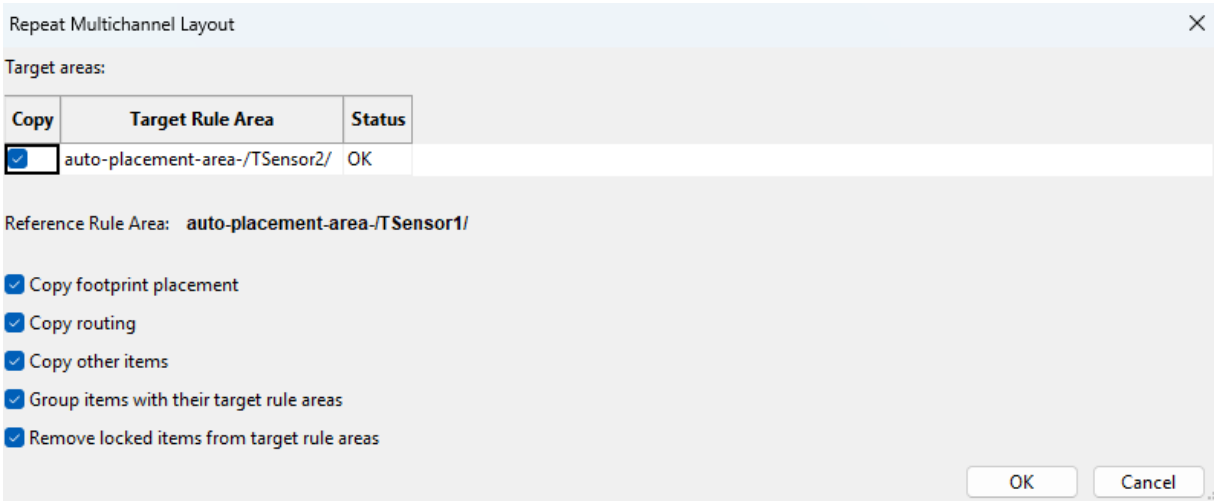
Click ***OK***.

Observe how both ***TSensor*** components are now enclosed by a ***Zone***. Click to select this newly created zone in the channel you have already placed and routed:



Go now to ***Tools→Multi-Channel→Repeat Layout…***.

Select the second channel (to which you are coping the first one):



See how the second channel takes the same aspect as the first one. You can now move them into the chosen location within the board limits.

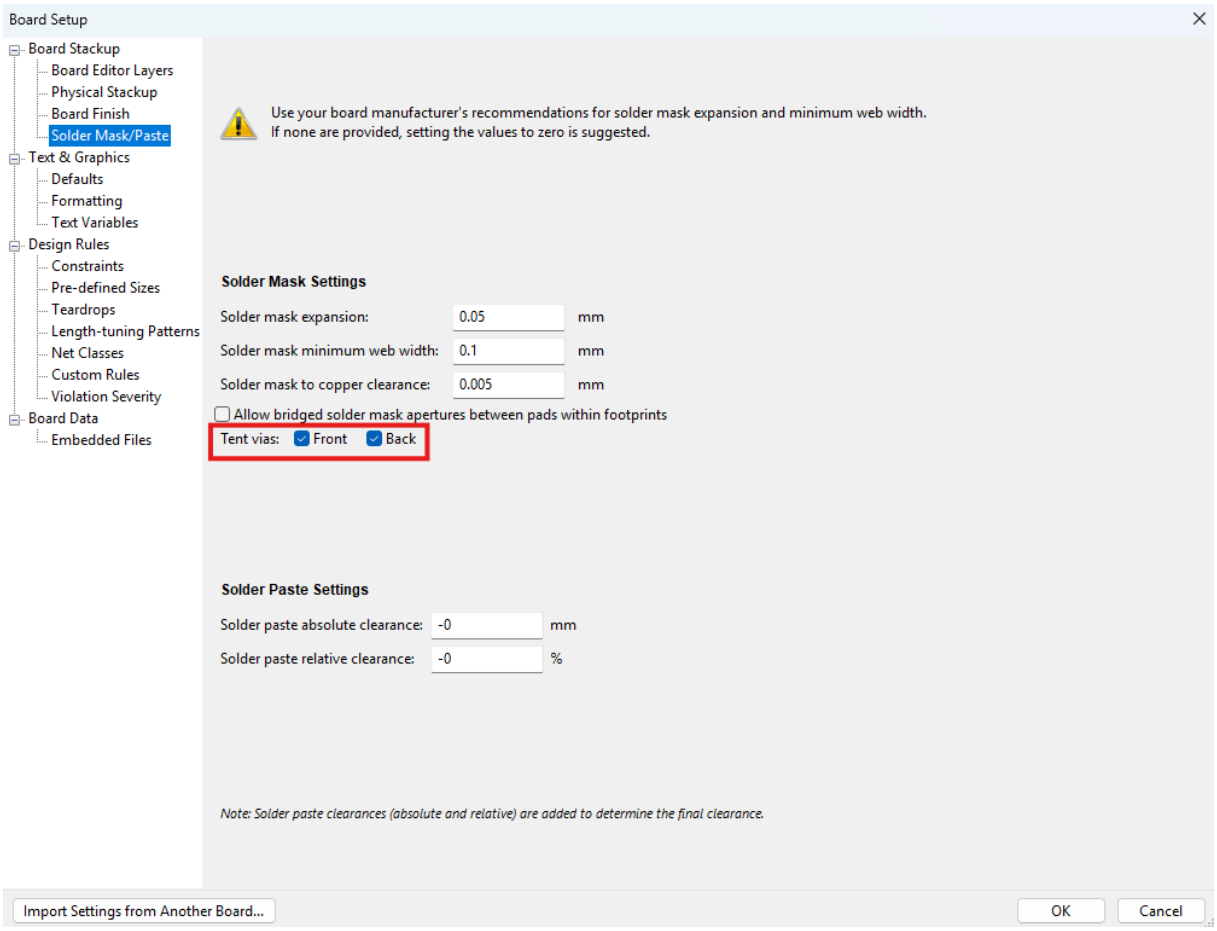
Tent vias

Problem

By default, ***KiCAD*** doesn't cover the vias with solder mask. This might actually be very useful and ease up the soldering process, since we will be avoiding any possible short-circuit to any via.

Solution

Access ***File→Board Setup*** and select the menu ***Solder Mask/Paste*** under ***Board Stackup***. Make sure you check the option ***Tent vias***, both on ***Front*** and ***Back*** sides.



All your vias are now tented.

Power supply layouting

Problem

Especial care must be taken when layouting power supplies. There is a mix of noisy signals and sensitive ones and therefore, some guidelines need to be followed.

Solution

The best reference on how to do a good layout for a power supply is usually provided by the chip manufacturer on the IC's datasheet. To illustrate with an example, we focus on the chip ***RT8097CHGB*** that we use to generate our 3.3V power rail. The following image is taken from the datasheet and is self-explanatory:

12 Layout

12.1 Layout Guidelines

The PCB layout is an important step to maintain the high performance of the TLV62565 devices.

- The input/output capacitors and the inductor should be placed as close as possible to the IC. This keeps the traces short. Routing these traces direct and wide results in low trace resistance and low parasitic inductance.
- A common power GND should be used.
- The low side of the input and output capacitors must be connected properly to the power GND to avoid a GND potential shift.
- The sense traces connected to FB are signal traces. Special care should be taken to avoid noise being induced. Keep these traces away from SW nodes.
- GND layers might be used for shielding.

12.2 Layout Example

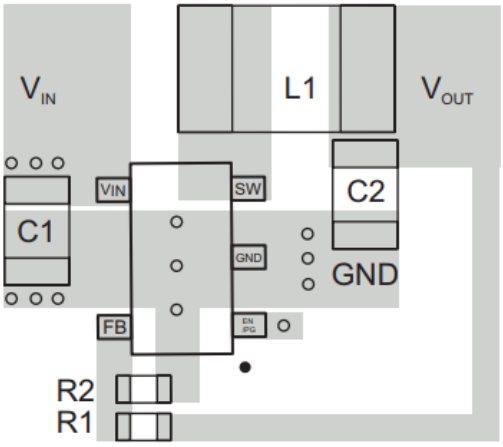


Figure 22. TLV62565/6 Layout

Power distribution

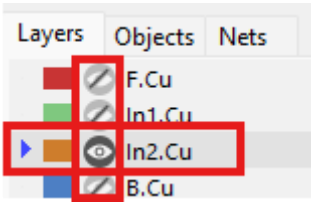
Problem

The two internal layers (***In1.Cu*** and ***In2.Cu***) are used exclusively to distribute power signals. This is a design decision. It is a good practice to route power signals (or high current signals) using areas (***Zones*** in ***KiCAD***) instead of tracks, in such a way that we provide a larger surface for the current to flow. This is the purpose of these internal layers.

Solution

We need at this point to share or split the whole surface of the layer into smaller sub-surfaces (zones), each of them assigned to a power supply signal. Keep in mind that the areas must be exclusive: they must contain all the elements belonging to a power supply net, and only to that power supply net. Otherwise said, one zone can't contain other power supply nets than the one it is connecting.

As a first step, we need to select the layer in question (that has been previously defined in the stack-up as ***In2.Cu***). It also helps making invisible the rest of the layers:

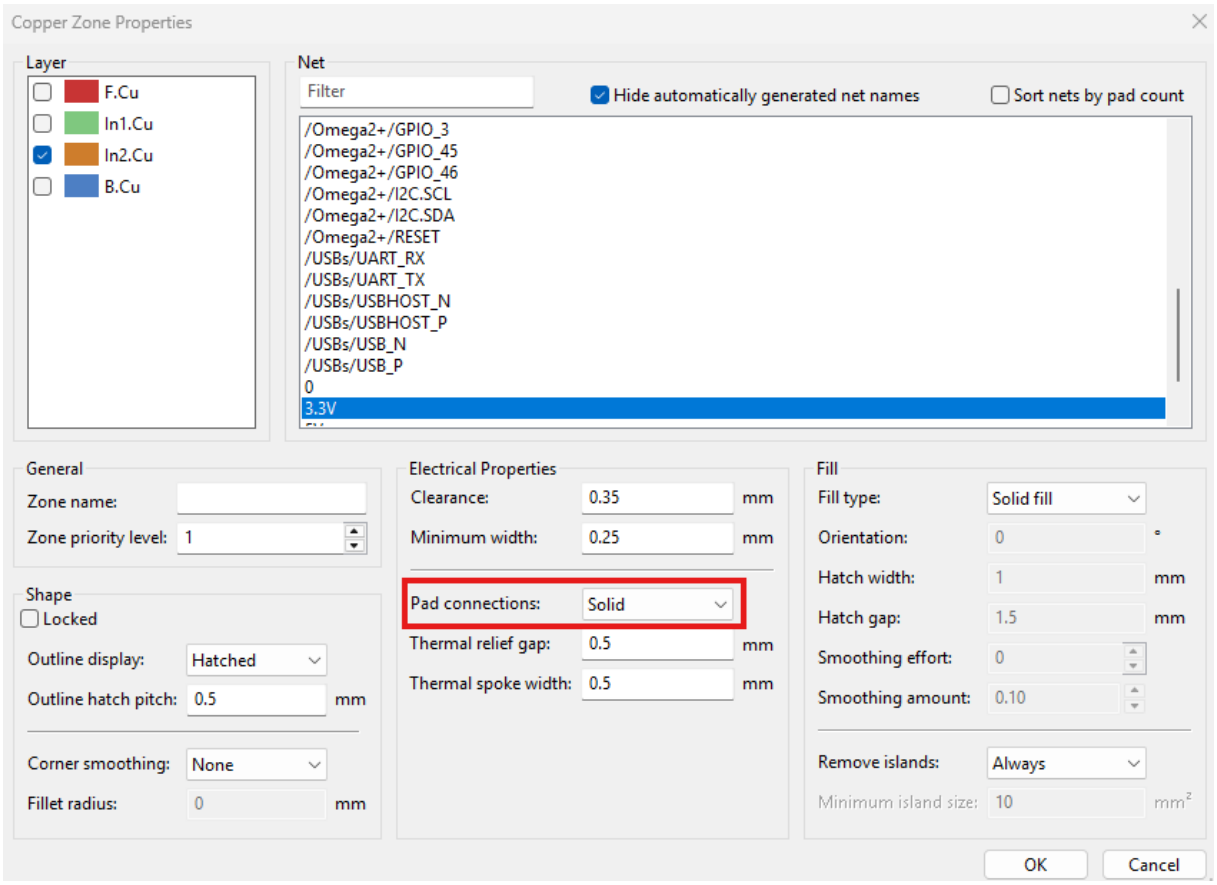


Don't forget that we are working on an internal layer. This means there are no pads or components on it. Only vias or through-hole pads have access to it.

We intend to create a copper area that contains all the vias and TH pads part of the same power net leaving out of it any via or TH pad belonging to a different power net.

At this point, it comes in handy the hotkey `. Place the mouse pointer on any element and hit `. See how the whole net is highlighted. This way you can identify all the elements belonging to a given net.

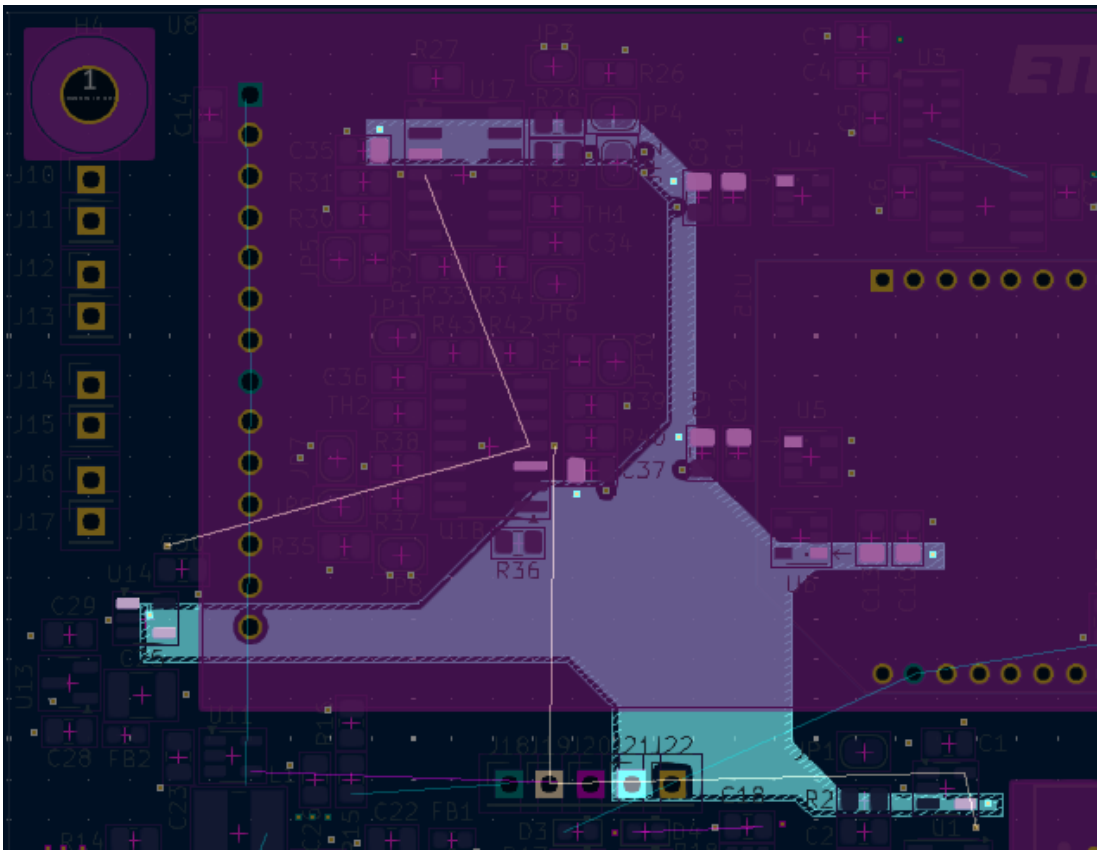
Let's start now with the zone creation: go to ***Place → Add Filed Zone***. Click on the design area and choose in the upcoming window the net you are routing:



Don't forget to set the ***Pad connections*** to ***Solid***.

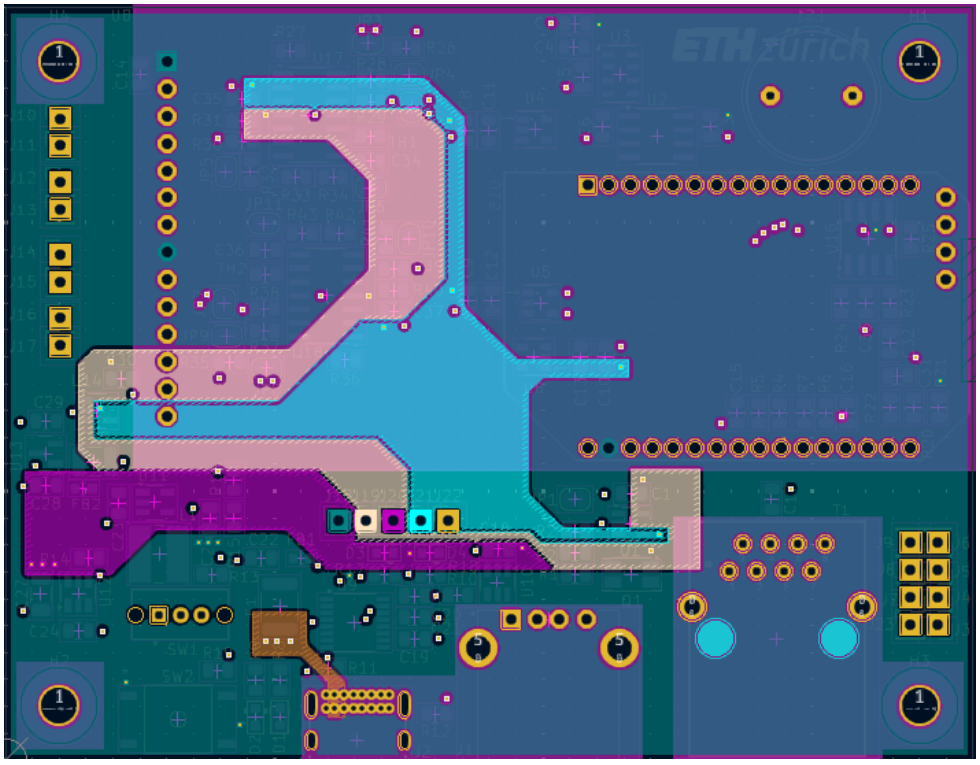
Zones with higher priority levels will be brought to the front whereas the lowest priority levels will be left in the background.

Now, we can start drawing the zone. Just define it by clicking correspondingly on the points that will define its perimeter. Make sure the plane is a closed surface:



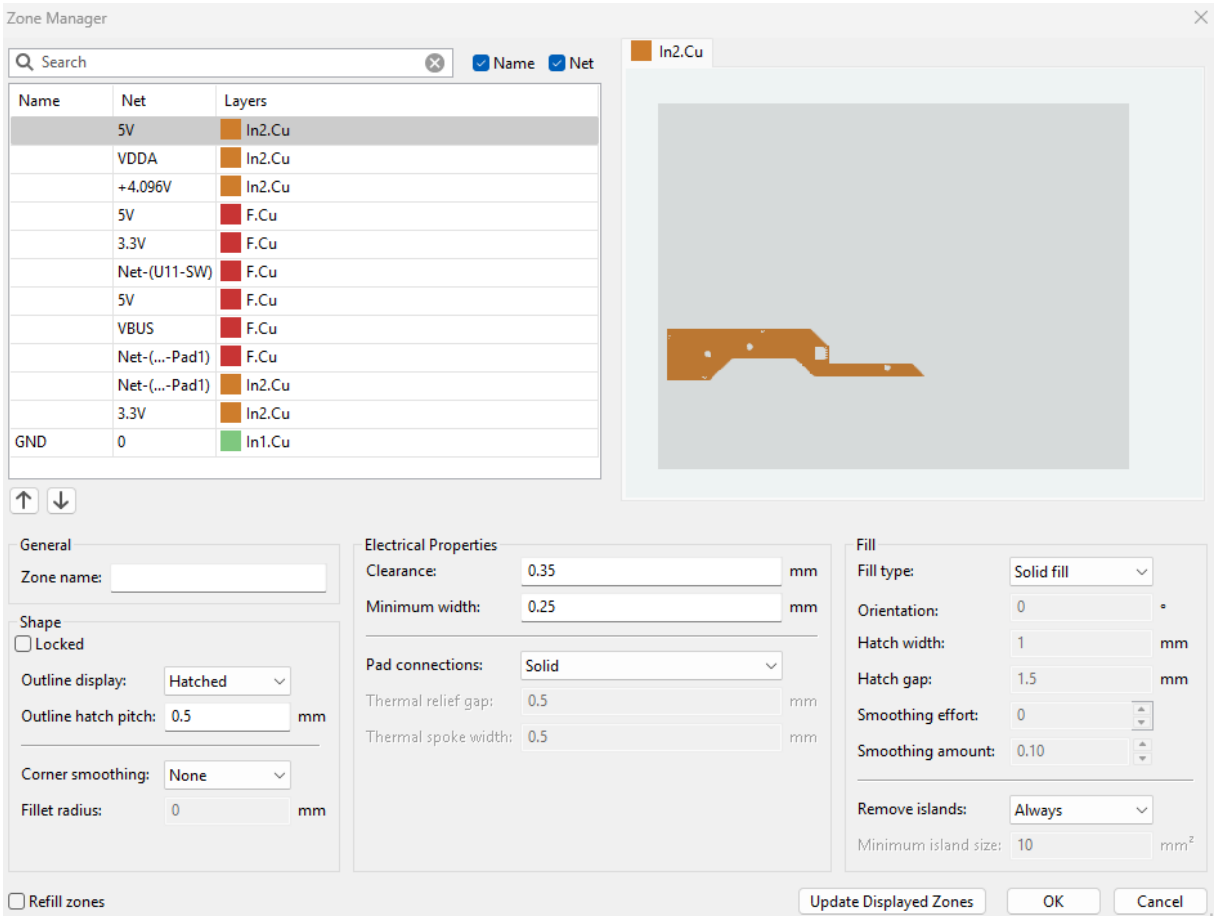
Iterate the process for each of the nets you want to connect on this layer. Start from the innermost nets and continue towards the outermost ones. Remember that you can create any kind of shape. Try to make them as large as possible and make sure they are a continuous surface. They also must not contain any element of any net that is also being routed on that layer.

A possible result of the process might look like this:



The remaining internal layer (***In1.Cu***) will be dedicated entirely to ground. One single ***Zone*** will take the entire surface on that layer.

You can manage all the zones features (including their priorities) in the ***Zone Manager***, accessible through ***Tools→Zone Manager***:



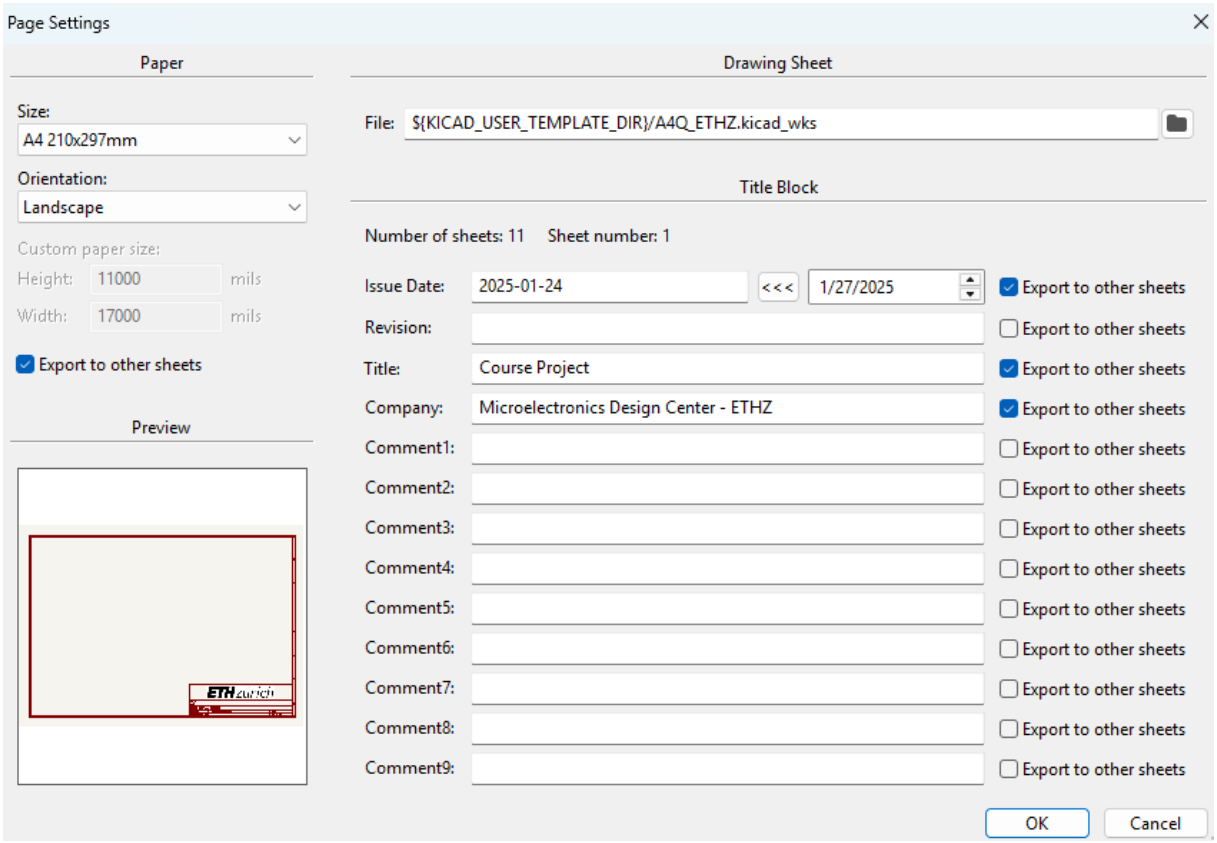
Use of sheet templates

Problem

In order to work in a corporative professional way, special care must be taken with the schematics presentation. The proper use of schematics templates contributes to a better professional outlook.

Solution

The very first step after creating our schematic sheet would be to set the sheet size and the drawing parameters. Access ***File→Page Settings***. Set the paper size to A4 and fill in the parameters accordingly. An example could be this:



Make sure all the pages are correctly numbered.

Differential pairs routing

Problem

By definition, some signals must be routed in differential mode: the signal travels from source to sink in two different copper tracks that run parallel to each other: one of them carries the signal with positive polarity and the other the signal with negative polarity. This type of signal transmission is much more efficient in terms of signal integrity and allows for higher speeds. Classic examples are ***Ethernet*** and ***USB*** data signals.

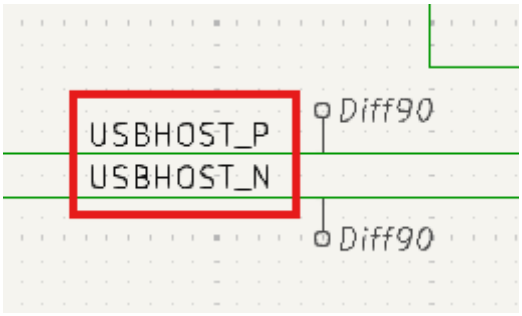
Solution

Schematic level

The signals that are to be routed in differential mode need to be named, at schematic level, as follows:

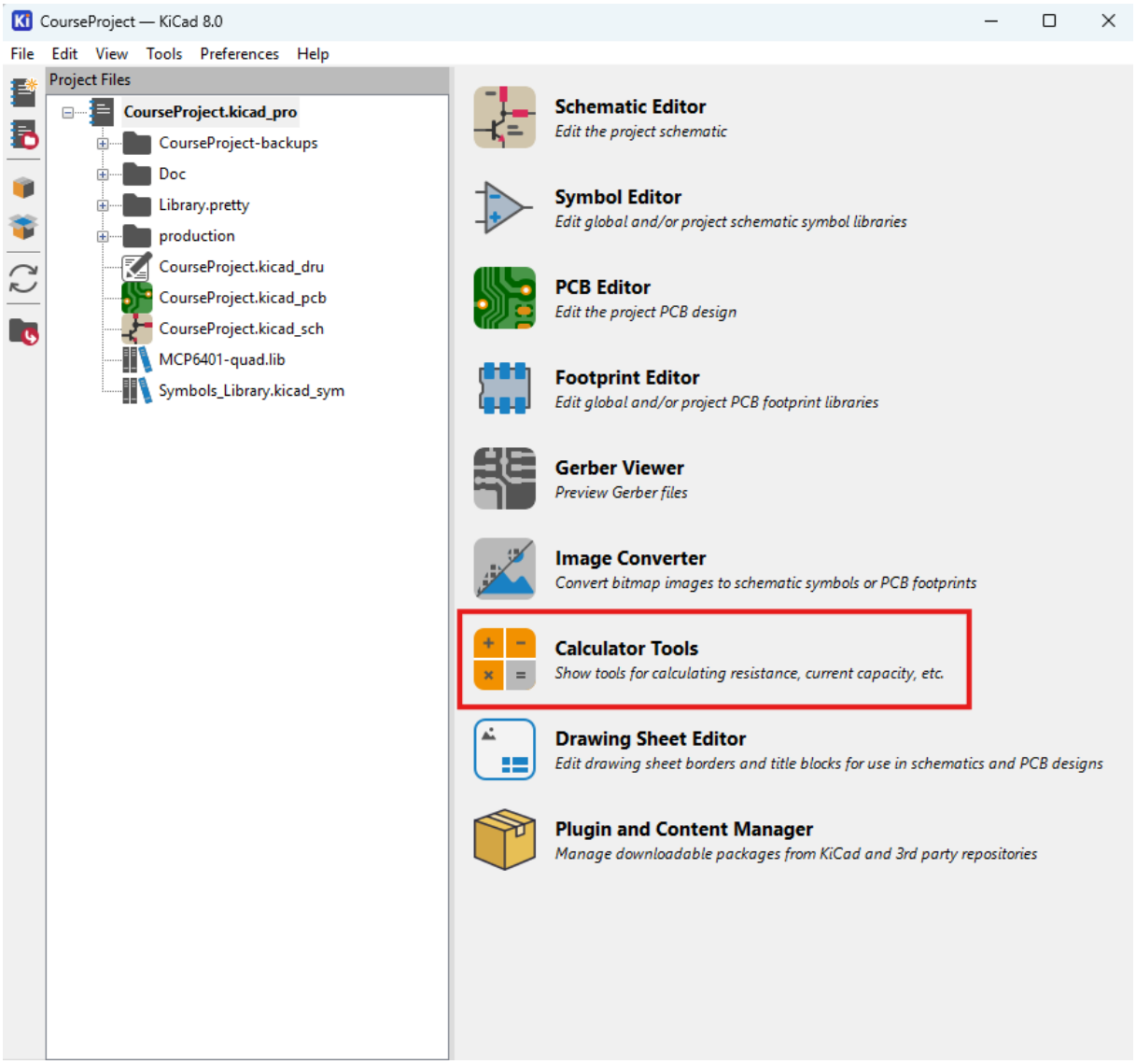
- ***SignalName_P*** for the positive signal of the pair.
- ***SignalName_N*** for the negative signal of the pair.

This is done by adding a ***Net Label*** (hotkey ***L***) to each of the two signals:

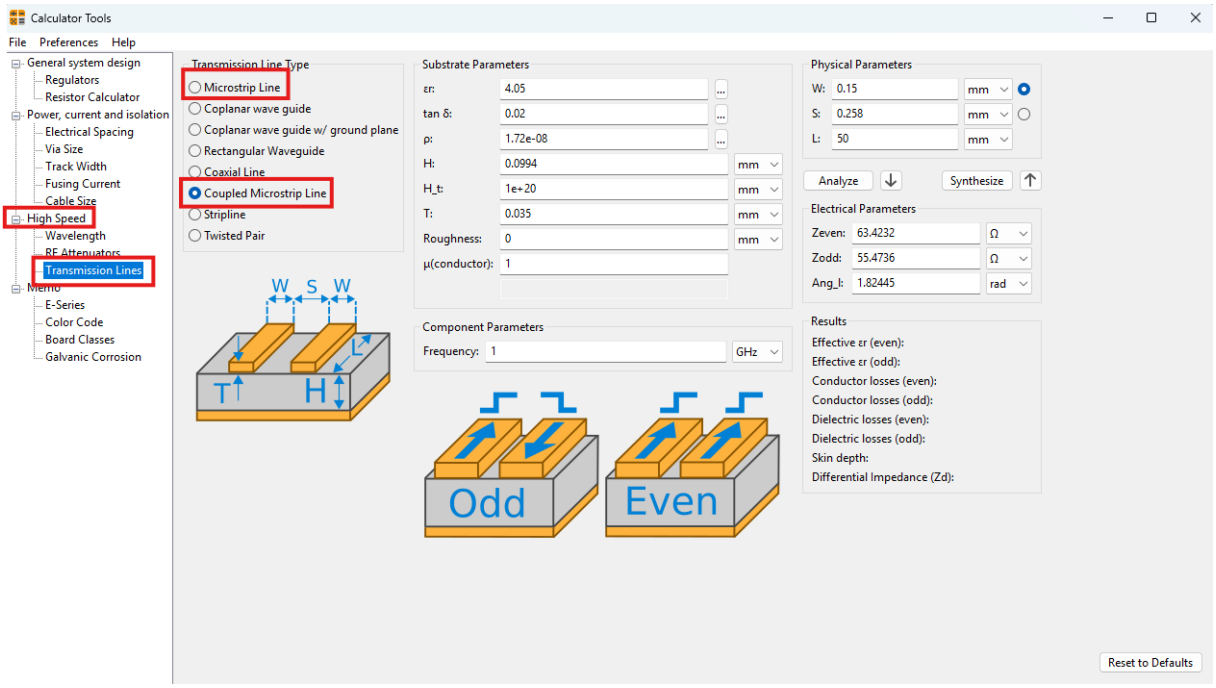


Layout level

The differential signals (as the single-ended ones) are defined through an impedance value. That value depends on the geometry of the routing: ***tracks width, gap between them and distance to the ground plane***. We can calculate those values using the ***Calculator Tools***:

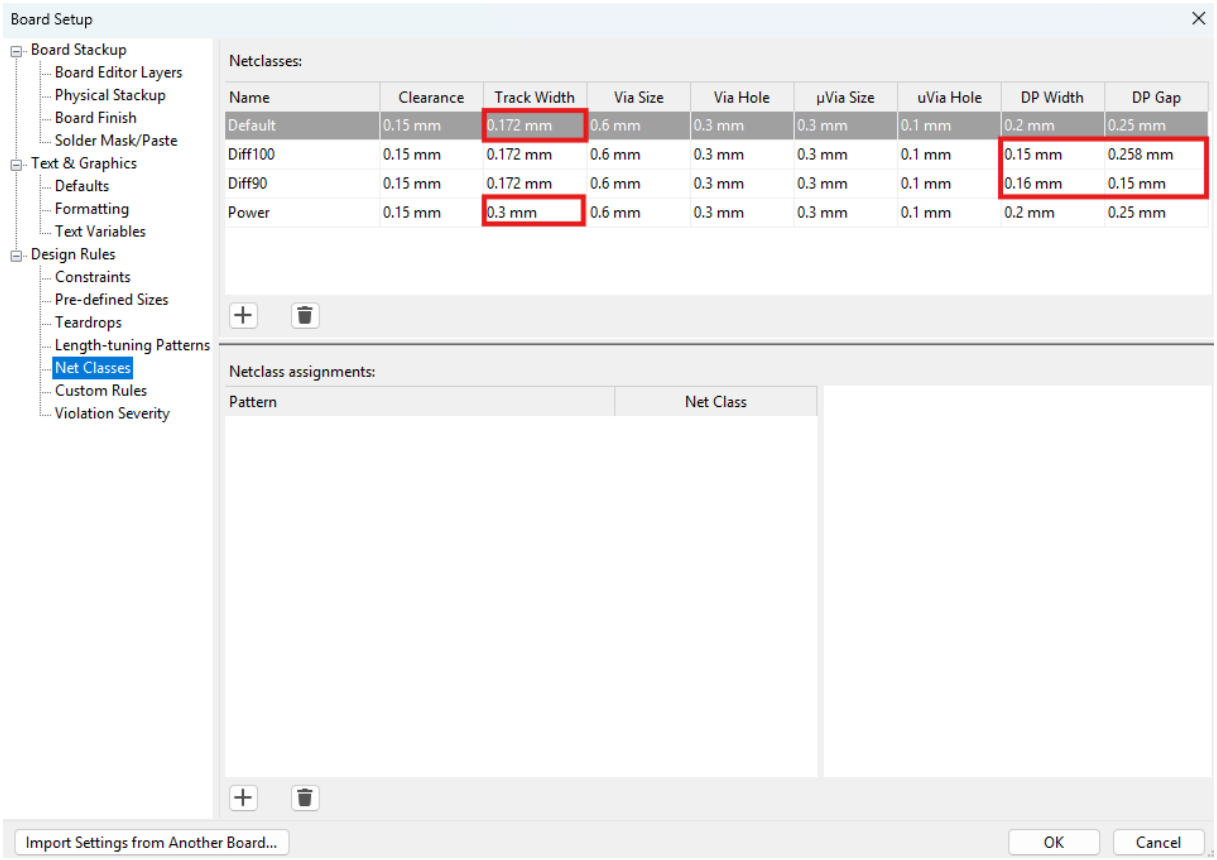


Under ***High Speed*** you'll find ***Transmission Lines***.

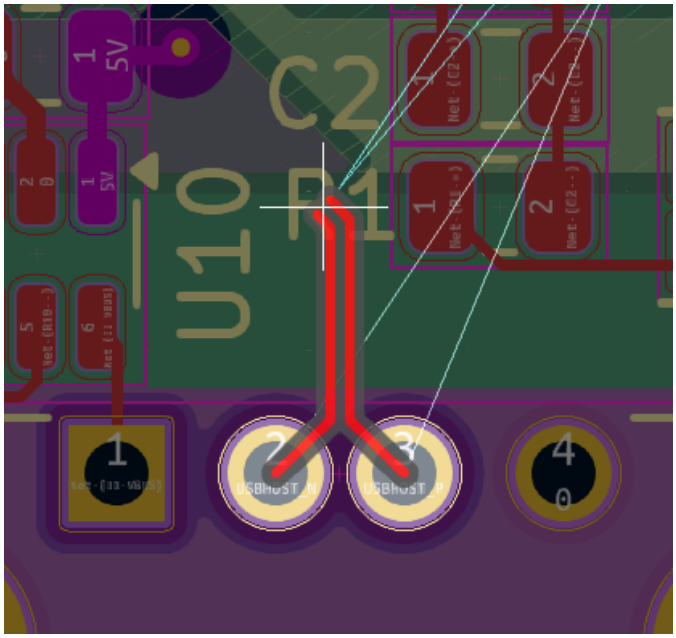


The two types of transmission lines we use in our design are microstrip lines (for single-ended transmission) and coupled microstrip lines (for differential transmission). Feel free to experiment with the parameters to obtain the physical dimensions.

For the course project, these dimensions are previously calculated for you. They are already captured in the ***Board Setup*** – ***Net Classes***.



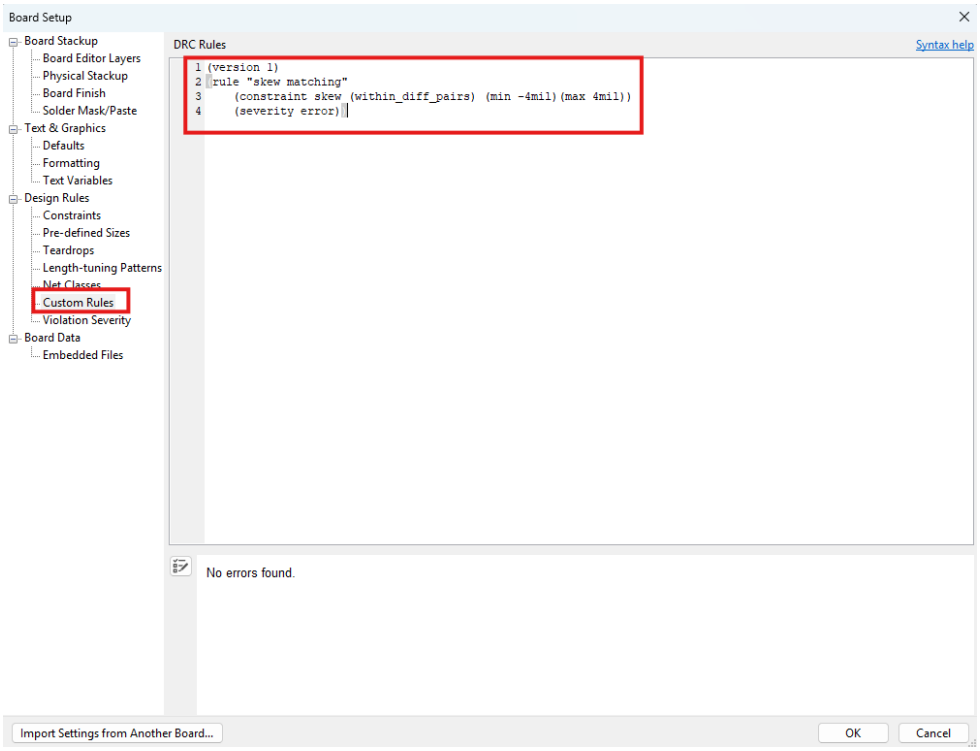
Once the rule is defined, we can start routing. ***KiCAD*** offers a special routing mode for differential signals, in such a way that both signals are routed simultaneously according to the rule. To run that command, go to ***Route→Route Differential Pair*** (hotkey ***6***). Click then on one of the pads of the differential pair.



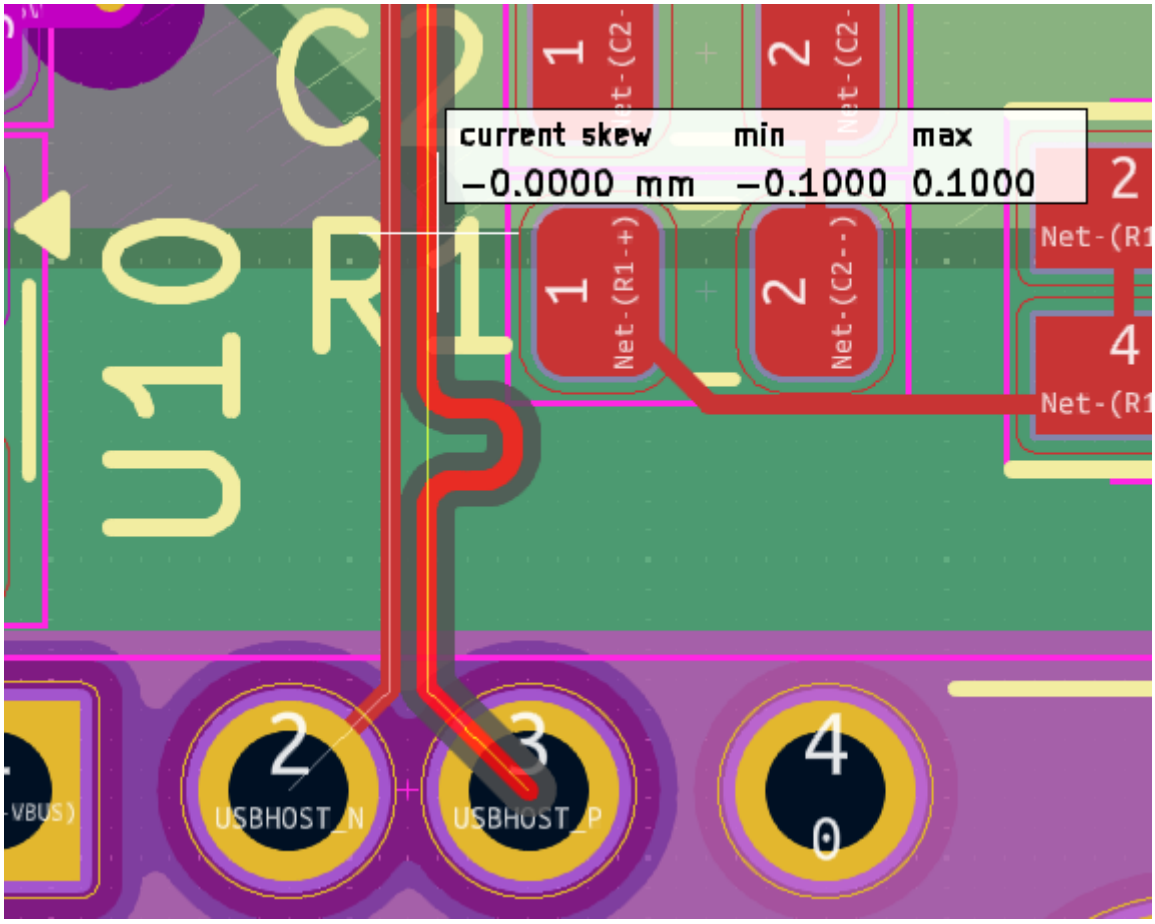
Observe how both signals of the differential pair are laid out together. We can route now in the same way as a single-ended signal.

It is also important to keep the length of both tracks within the differential pair as equal as possible.

There is no default rule for this purpose in ***KiCAD***, but we can create a custom one to get it checked. Access ***File→Board Setup*** and select the option ***Custom Rules***. Then, type in the following text:



Let's now tune the length of the shorter track of the differential pair. Go to ***Route→Tune Skew of a Differential Pair*** (hotkey ***9***) and click somewhere on the shorter of the two traces. See how ***KiCAD*** adds serpentines to the track as you hover the mouse:



Click to fix the accordion. By default, the matching tolerance is ±0.1mm, which is more than enough for our purpose.