SOURCE

SOURCE:
a quarterly periodical that takes a look into interesting opensource projects and applications. In this issue we have a focus on opensource hardware approaches. With tutorials on how to design a PCB in KiCad, a beginners FreeCAD project, a look at how to certify projects as opensource with OSHWA and a little Inkscape.

Welcome to SOURCE

Hello, I'm Jo, AKA Concretedog, and first of all, thank you for taking interest in SOURCE.

If you've looked through issue 1 and 2, you'll know by now that SOURCE is my place for writing content that doesn't fit elsewhere in my paid writing work. It tends to be a little more in depth in it's tutorials whilst promoting opensource. It's pay as you feel and I'd love to see some donations come in for my time as it's a pretty big undertaking getting this all together! That said, times are hard and if you aren't in a position to donate, I really hope you enjoy reading this, tell your friends about SOURCE, and much solidarity to you.

In this issue I've written to the theme of opensource hardware. For me the two big applications for hardware are KiCad for all your electronics design needs and FreeCAD for your mechanical CAD and more. I've also looked at how to approach certifying opensource hardware with the Open Source Hard Ware Association (OSHWA), and whilst there is no rule that says you have to certify there, it's a marker that your project is well licensed and configured in a way that makes it clear to people what they can do with it. I really hope you enjoy this issue, as ever do share this with people, you can host it, compress it and email people, put it in your USB dead drop, or just share links with people, let's spread opensource far and wide.

Contents

FreeCAD, A Beginners project, PCB Magnet Clamps

I spend a lot of my life writing about FreeCAD. In case you don't know FreeCAD is a free and opensource CAD environment that can be used for a huge variety of use cases. In Issue 1 of SOURCE we did a deep dive into using FreeCAD with a complex yet accessible set of computational fluid dynamics tools to perform sophisticated analysis. In this article we are going to focus on a useful first steps project with FreeCAD to make some useful hardware.
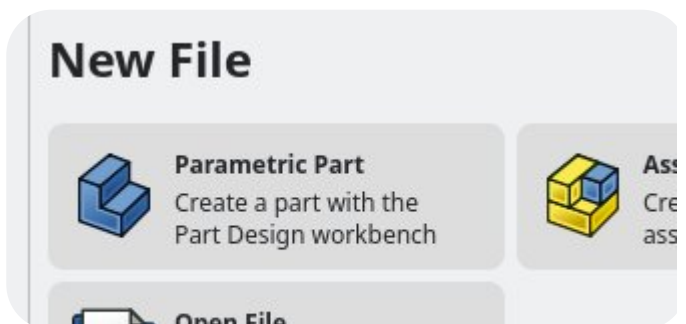
We're going to walk through designing these PCB tool clamps. If you see this tutorial and are moved to want the PCB clamps then the models are already available on this repository https://github.com/concretedog/magnet_pcb_clamps. However, it's much better to go through the tutorial then you have the opportunity to make changes and adapt them to your own needs.

So, to begin at the beginning, first head over to FreeCAD.org and download FreeCAD for your operating system. There are Linux packages on most Linux platforms and there is a handy Linux app image on the download page. Windows and Mac users have install options also.

Once installed launch the application. You'll be met with a start page which has a few icons that act as shortcuts to slightly different types of new project, and below these some example projects shipped with FreeCAD. Note that when you have made and saved a few FreeCAD projects the most recent projects will appear on this page allowing you to

quickly jump back to projects as required.

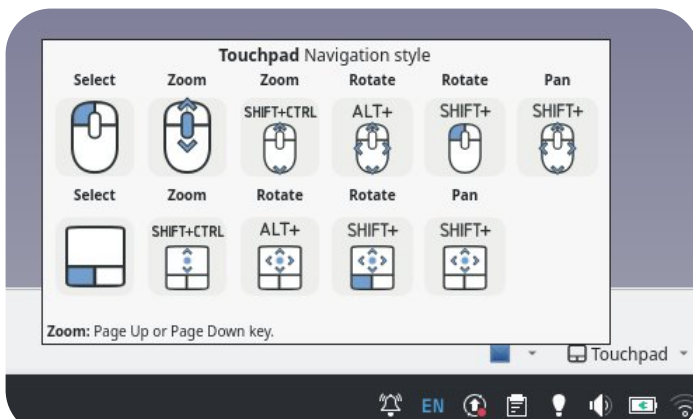Let's left click the "Create a Parametric Part" icon.

You'll notice a few things change on the screen. One of the most notable changes is we have jumped to a different "workbench", a workbench called Part Design. FreeCAD uses a workbench system where there is a dropdown menu of multiple pre installed workbenches. When you switch between workbenches the different workbenches all have different sets of tools. Don't be overwhelmed if you look at the list of workbenches. You'll find as your FreeCAD knowledge develops that you might only use a small amount of workbenches a lot, but occassionally will dive into specific workbenches for very specific reasons. In this tutorial we are only going to use the Part Design Workbench and the Sketcher workbench, and, you'll see as we go along, that often you will not have to actively change workbenches but rather it will do so automatically for you when you need to.
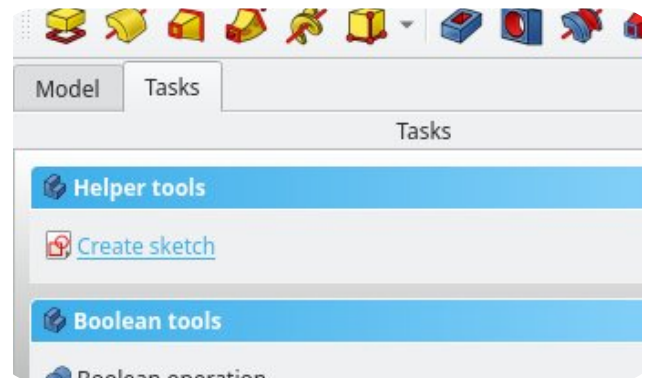
L et's quickly talk about navigation. When you begin learning FreeCAD the navigation can seem difficult, or, when watching a video tutorial, can all just magically happen and you don't know how things have been moved, viewed, rotated etc. I'll try and avoid that, but for now let's make sure that our navigation schemes work in the same way.

FreeCAD has a pile of built in navigation schemes and some of these emulate the navigational approach of other CAD environments. So if you have a big background in Blender, or Revit, then you can set the navigation scheme to match and things like mouse/trackpad zooming and rotation should work in the way you are used to. For now though, or if you are totally new to FreeCAD let's make sure our navigational style is set to "Touchpad". You do this by clicking the pop up Navigation Styles menu on the lower toolbar and selecting "Touchpad". Note if you hover over this menu on the toolbar you get a preview image which describes how the selected navigation style works and provides a key for common navigation tasks.
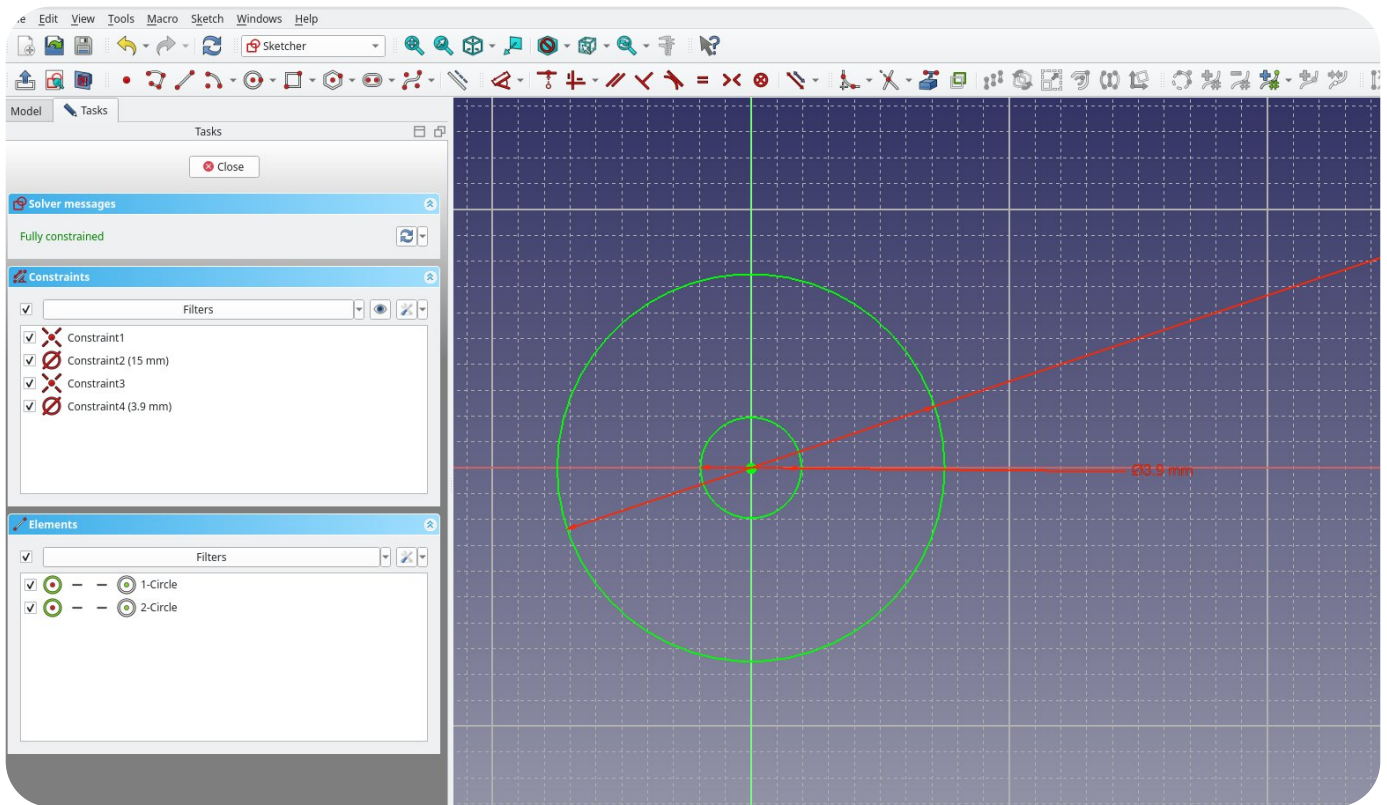

As we clicked to "Create a Parametric Part" icon FreeCAD moved us to the Part Design workbench and created an item in the file tree view called "Body". The Part Design Workbench creates parts by sketching geometries and then performing operations on the sketches, like creating a pad of a sketch to pull a sketch into a 3D object. The important thing to know is that the term "Body" refers to a single continuous part, so it you try to sketch or create something that has 2 completely separate parts in Part Design, you'll have problems! A way to think of it is to imagine a nut and a bolt. Each item is a Body, the nut is a different item to the bolt and both the Body items are completely

separate. When you screw the nut onto the bolt they are still two separate bodies, but we would call this an "Assembly". So even though in this tutorial we aren't going to make an assembly, it shows us that if we were to model a nut and a bolt, each part should be in a separate unique body.

Let's begin by making the simplest part of our PCB clamp design, the upper section that is essentially a disc with a hole in it to form the upper clamp part. Notice in the file tree area we have two tabs, one is "model" and one is "tasks", click on the tasks tab and you should see "Create a Sketch" as an option. In FreeCAD, much the same as all software, there are often multiple ways to achieve the same outcome. We are next going to left click on the "Create a Sketch" entry on the "tasks" tab, but you can also note that in the main tool area of the Part Design workbench there is also a tool icon for "Create a Sketch".

Throughout this tutorial we will name tools using the rollover text tool tip that appears when you hover your pointer over an icon. This means you have to explore and read the tool descriptions to find the tool, but this in turn helps you learn the software!
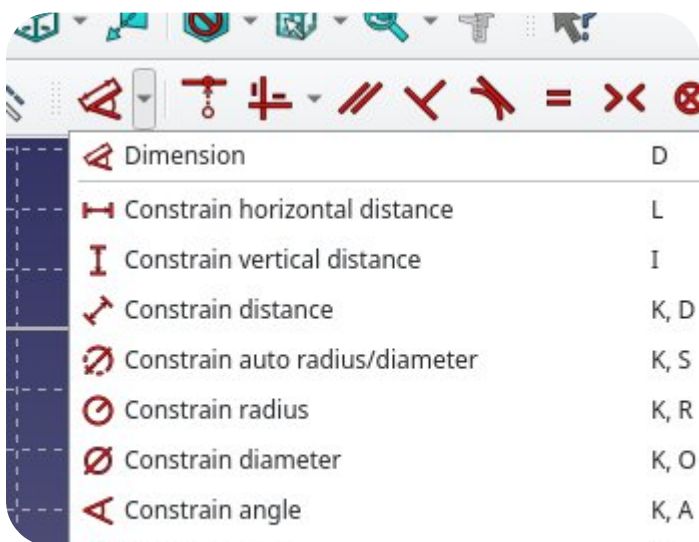
Having clicked the create a sketch option we should then be asked which plane we want to attach the sketch to, XY, YZ or XZ. As we aim to 3D print these items it can make sense to imagine we are working looking down on the bed of our 3D printer, our printer bed would then be the XY axis, so click XY axis.

> When you have used a drawing tool in the sketcher workbench you can right click to "drop" the tool and return to the regular pointer.

At this point FreeCAD jumps to another workbench, the Sketcher workbench, where all the tools are changed for tools useful in creating sketches. Notice that in the file view area the "body" object now contains a "sketch" object as well. This first part we are going to make is essentially a round disc with a hole in the centre. We used an M4 bolt as the central column of our clamp, and we wanted this upper part of the clamp to be slightly fixed in place, to achieve this we made the central hole slightly undersized (3.9mm) so that it slightly

threaded onto the bolt. Left click on the "Create a circle" tool icon. Now in the preview window left click on the very centre point of the two axis. This constrains the circle we are about to draw so that the centre of the circle, no matter the diameter, is always set on that point. Dragging away from the centre point we can see a circle. You should also see an "On View Parameter" input box where you can type in a value for the diameter of your circle in here to finish a sized circle item. However, let's not do that, just so we can see how we can add a diameter constraint to an item previously drawn. Without typing text into the entry box, left click to place a circle object of any dimensions. Your circle is placed, but you might notice that we are still holding the draw a circle tool. If we left click again, we will start drawing another circle. If you accidentally do so you can use control and z to undo the error. Whenever we use a tool in the sketcher workbench we keep hold of that tool until we actively put the tool down, or select a different tool. To put the tool down and return to the normal pointer we simply right click in the main preview window area.
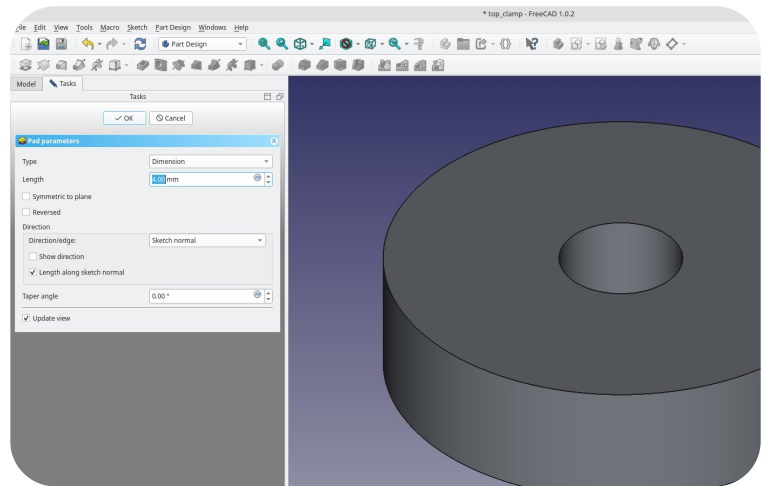
Note that our circle appears as a white line. If we left click and hold on the white line and drag we can resize our circle. This means that the circle isn't fully constrained. It's good practice to try and fully constrain items in sketches as this makes is totally trivial to update a parameter, dimension for example later in the design process and push the changes through the design correctly. A sketch is fully constrained when all the lines in it are green and in the file view menu it reports not "dof" or degrees of freedom. At the moment we have one degree of freedom, and that is the diameter of the circle. We can left click on the "Dimension" tool and then left click on the white circle line. The tool will automatically realise the geometry is a circle and will offer an input box to add a diameter. Let's set this to 15mm. Should you want to define a circle by it's radius, you can click the "Dimension" tool icon drop down menu button and select Constrain radius there.
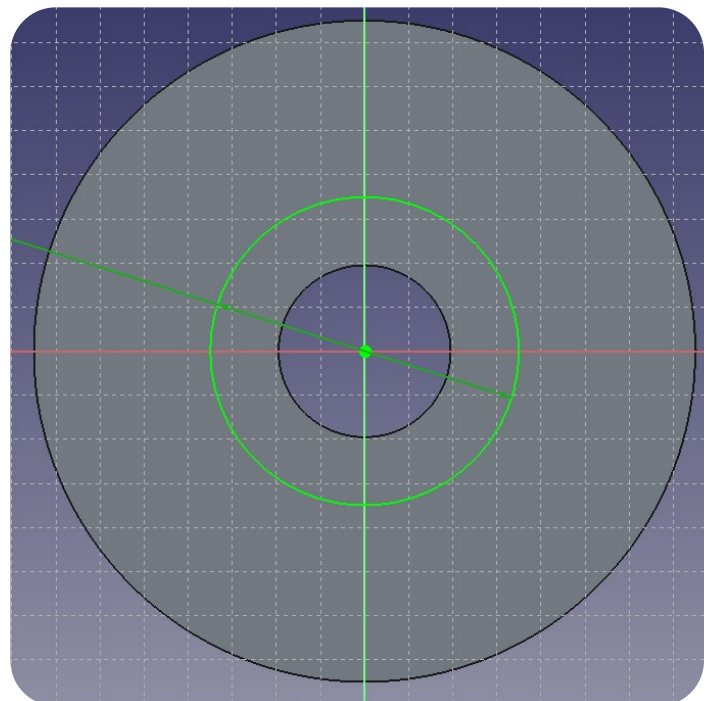

We should see that the sketch has now gone green and there are no degrees of freedom. However we need to add another sketch item into this to create the hole in the centre. Again click the "Create a circle" tool and again click the very centre mark at the origin point and drag out to create a small circle. Either through the input box, or by the method we used for our first circle, constrain the diameter of this circle to 3.9mm.

That's our first sketch complete. Click the "Close" button and the sketch will close and FreeCAD will transport us back to the Part Design workbench.

In the file tree view single left click on the sketch item we just created and then left click on the Pad tool. You should see a Pad dialogue open and our sketch will by default be padded to 10mm height. In the Pad dialogue set the pad height to 4mm. Then left click OK to apply and close the pad.
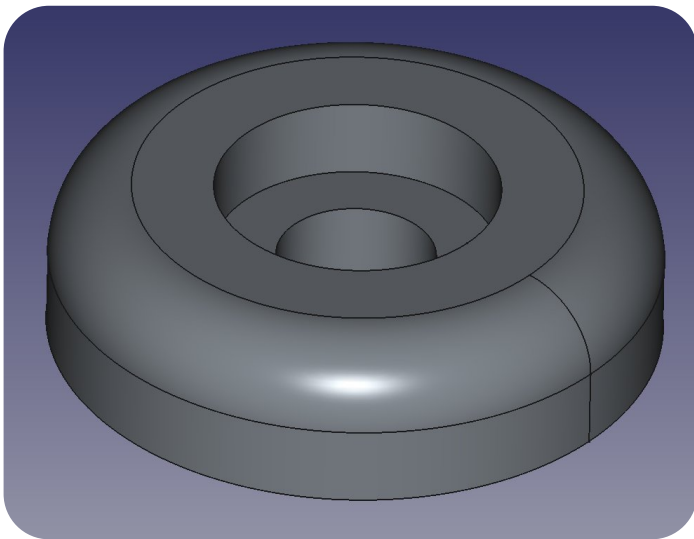

We want to add a small recess into which our machine screw head will sit. The machine screws have a head of 6.5mm diameter and are around 2-3mm deep. To create a hole for these in the preview window left click the upper face of our Pad object. Make sure to only select the upper face (it doesn't really matter which side of the Pad you choose as they are identical) and then click the "Add Sketch" button. Notice that this time when we create the sketch we don't get asked what plane we want the sketch to be in as the sketch is already attached to the face of the Pad we selected.
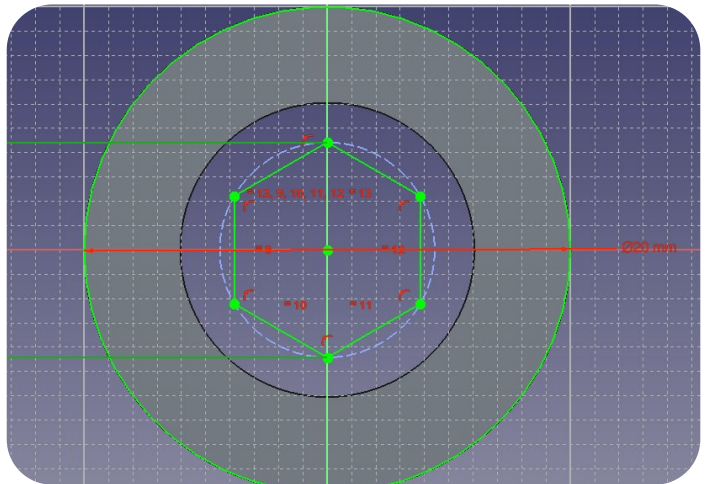
In this sketch draw and constrain a centred circle that is 7mm diameter. Then close the sketch. Then with the sketch highlighted in the file tree view click the "Create a Pocket" tool. The Pocket tool dialogue will open and automatically a pocket will be applied that is 5mm deep, so will travel all the way through our part. Adjust the pocket depth to 2 mm to accommodate the machine screw head and click apply. In the next section we will add a hexagonal pocket to a part, if you decided to use M4 bolts with hexagonal heads you could easily return to this sketch and adapt it. Finally to finish this part in the preview window click to select the outer most edge of the upper side of the padded object. Then left click on the "Fillet" tool icon. In the fillet dialogue increase the radius to 2mm and apply the fillet to create a nice upper edge on our design. Note that it's a good idea to add operations like fillets and chamfers at the end of a part making process as they can be a common cause for creating topological naming issues etc. Finally save this part as a project using the "File > Save As" menu.
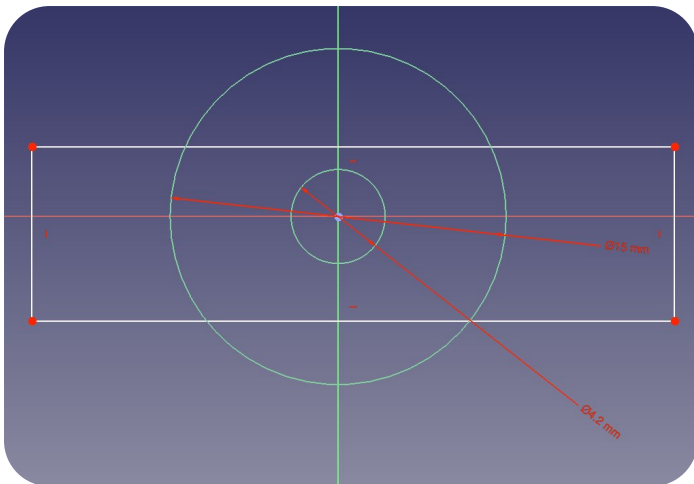

Next let's make the base part of our PCB clamp. We begin this by, in the active body, clicking to create a sketch and connecting the sketch to the XY plane. In this sketch, again, we are going to create a circle for the outside of the base object, let's make it slightly larger than the top cap part we made and set the diameter to 20mm. Measuring our magnets they are 12mm diameter and 3mm thick. So let's create an inner circle in our sketch with the diameter set to 12.1mm. You may need to experiment with this value but on the 3D printer used to print the ones in the pictures we found 12.1mm resulted in a pleasing press fit for the magnet which we reinforced with a drop

of superglue. With our two circles created, close the sketch, and then use the Pad tool once again to pad the sketch to a height of 3mm.
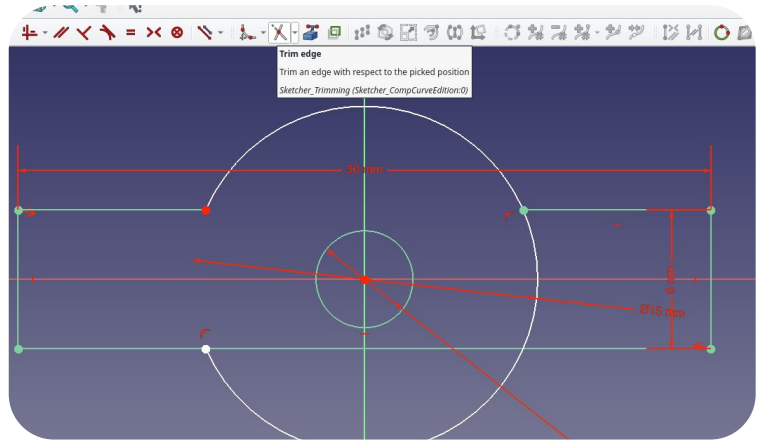

In turn, similar to the cap part we made, in the live preview window click the upper face of the Pad item and add a sketch. In this sketch we create the matching 20mm diameter outer circle but instead of an inner circle we will create an inner hexagon. Left click on the "Create a regular polygon" tool, it should by default be set to draw a hexagon, but if it isn't you can use the drop down menu on the tool icon to select the hexagon option. Similar to the create a circle tool, left click the zero origin point to begin to draw a hexagon constrained to the centre of the design. Moving the mouse or trackpad outwards you can then pick an outer node/corner of the hexagon and left click to place it on the y axis line. This will constrain the hexagon so that it has an upper and lower point/corner on the line. To size the hexagon we can select any two opposite points or nodes, we've opted to dimension the hexagon with the uppermost and lowermost nodes. Carefully left click each of those nodes and then click the Dimension tool drop down menu and select the "Constrain Vertical Distance tool". Our (very cheap!) nuts were quite varied in tolerances so we ended up with a hexagon constrained to 8.85mm on this axis which resulted in a loose fit but with room for plenty of glue to hold the nut captive. Closing the sketch we then padded this section to a height of 3mm to accommodate our captive nut. We repeated a similar process of adding a sketch to the upper face and added an outer circle and an inner circle constrained to 4.1mm through which our bolt would travel and padded this to 2mm. Finally, similar to the top cap we then added a chamfer to the upper outside edge of the base part.

All we have left is to make the floating clamp section which has the small rectangular parts which make it easy to press down. For this we will use some other tools in the sketcher workbench. We began in a similar way to the upper cap and the base parts by drawing two circles, one a 15mm diameter circle which matches the upper cap outer diameter. The inner circle is set at 4.2mm diameter so this floating section will travel freely up and down the bolt. We then left clicked the "Create Rectangle" tool and left clicked and dragged to create a rectangle over the top of our larger circle.
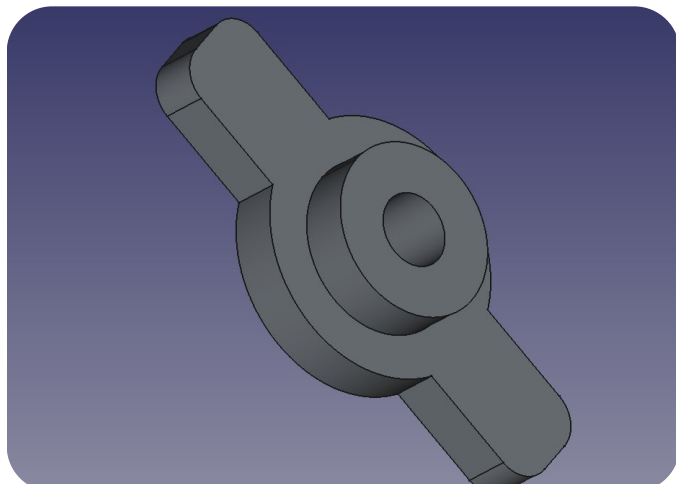

The plan is to position this correctly to form the two tabs and then delete the lines we don't need. To get started we left clicked to select the upper right and lower left corner nodes, then in turn, left click to click the centre zero point node, pay attention that you do this in that exact order or the next part won't work! With those three nodes selected in the correct order next left click the "Constrain Symmetric" tool. This tool makes geometry behave symmetrically with respect to the final node. If you now left click to grab any part of the rectangle and drag it to change dimensions you'll see that it is always symmetric to the zero origin point.

Selecting the upper horizontal line of the rectangle we can use the Dimension tool to constrain the horizontal length of the rectangle to 30mm and likewise a vertical line to 6mm. At this moment your sketch will be fully constrained, but during the next series of operations, we will break the constraints as we delete parts of the sketch.
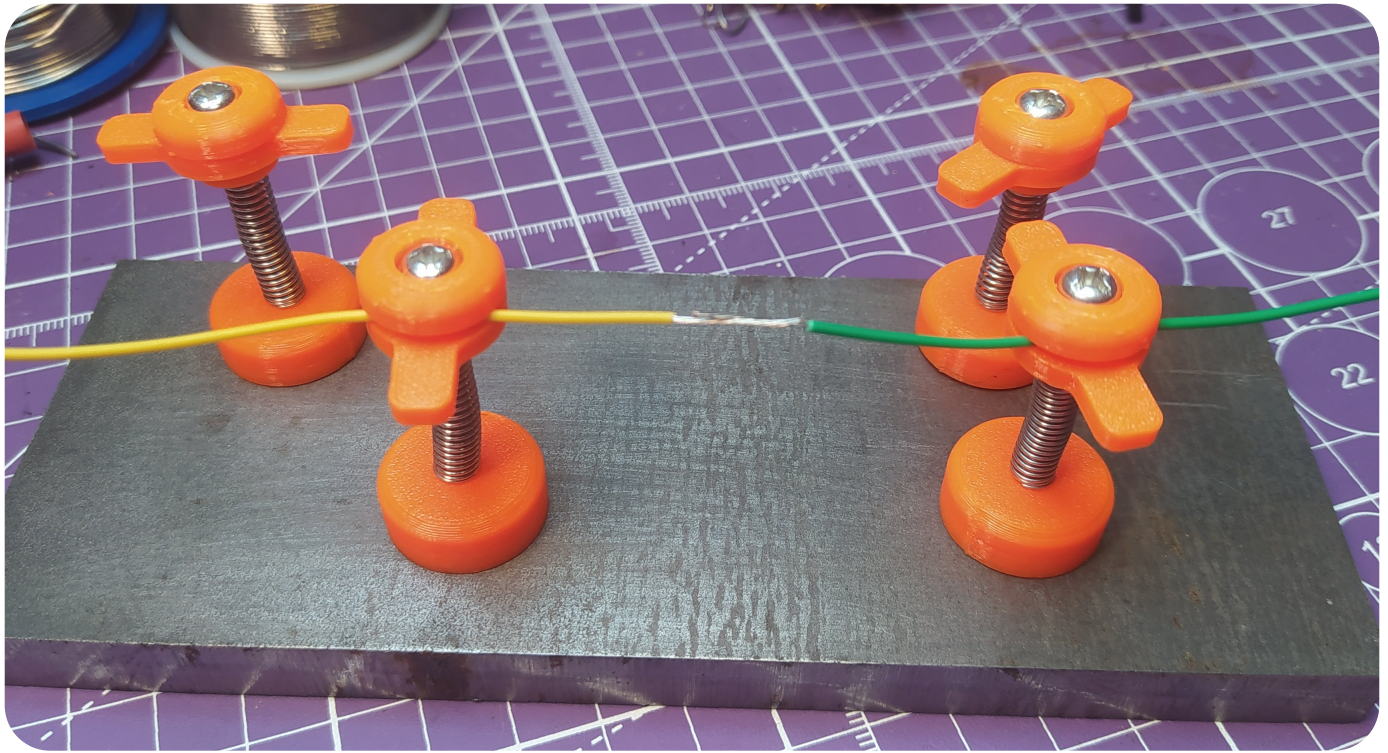
For this we are going to use the "Trim edge" tool. Selecting this tool we can then click the line sections that we don't need in the sketch. One thing of note is that if you left click sections and they are removed you can sometimes end up with a line section that isn't attached to anything else. As the "Trim edge" tool is designed to trim away line sections connected to other lines it cannot delete an unattached line. However, you can easily right click to drop the "Trim edge" tool and then with the standard selection tool, select the unattached line and click delete. After you have removed the unwanted interior lines from our sketch you will notice it is now partially unconstrained, best practice is to have the sketch fully constrained but in this instance it's OK to leave it and close the sketch if you prefer.

Closing the sketch we return to the Part design workbench and we can now pad the floating clamp sketch we just made to a height of 3mm. Finally we can add a sketch to a face of the pad object create a circle sketch and then pad this additionally to 3mm to provide a little preload compression to our spring. If you find the springs you have are too weak, or a little to firm, you can adjust this item to compress or decompress the springs based on your requirements.

FreeCAD has a Mesh workbench where you can perform all manner of mesh alterations, and create meshes with custom settings, but often for quick 3D printing jobs the standard mesh settings can be applied and you can export an stl or 3mf file from any workbench using "File > Export" and selecting the desired file type. One thing of note is that you have to select the final item in the file tree to select the object to export. So for example if you want to export the base item, the last operation we performed was a "Fillet" so the last item in the file tree is "Fillet" and we need to highlight that to export our 3d object file.

With all 3 parts exported, load them up in your favoured slicer, slice them and print them! To assemble each clamp in addition to the three 3D printed parts you'll need a circa 35mm long 5mm diameter 0.5mm wire diameter compression spring, a 50mm M4 machine screw and M4 nut, and a 12 x 2mm circular Neodymium magnet. Press the nut into the hexagonal base hole and secure it with a couple of drops of superglue. Wait for this to dry before you insert the magnet or else the magnet may pull out the nut! When ready, insert the magnet, again with a small drop of glue to secure it in position. Let the base assembly dry and then screw in the machine screw with the spring and upper clamp parts already threaded on to the screw. Finally, repeat for as many clamps as you want to make!

These clamps, as designed are available for download on this repository, and we, elsewhere in this issue of SOURCE have used them as an example of how you might get an opensource hardware project certified by the Open Source Hard Ware Association.
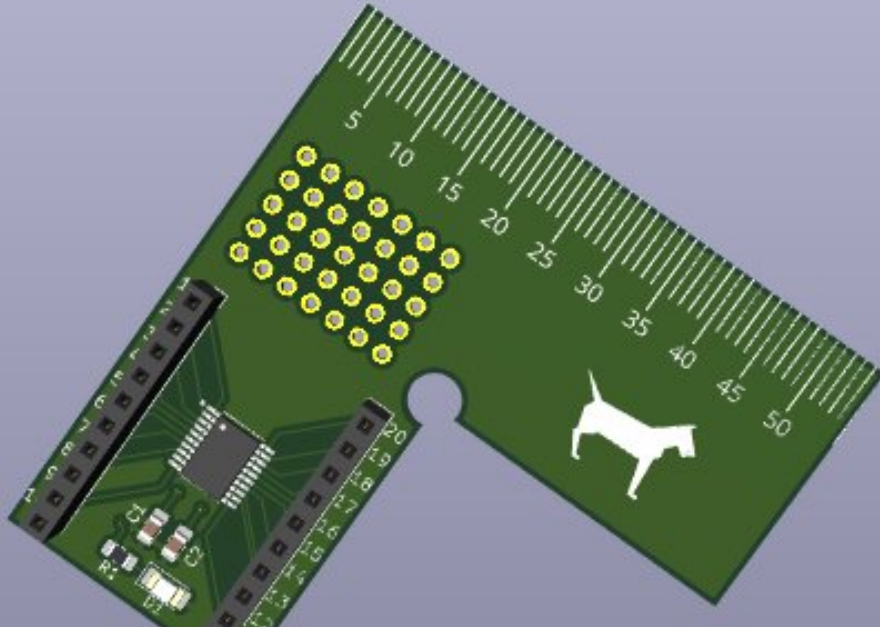
Pay what you feel via KoFi

Support me on Ko-fi

Pay what you feel via Paypal here
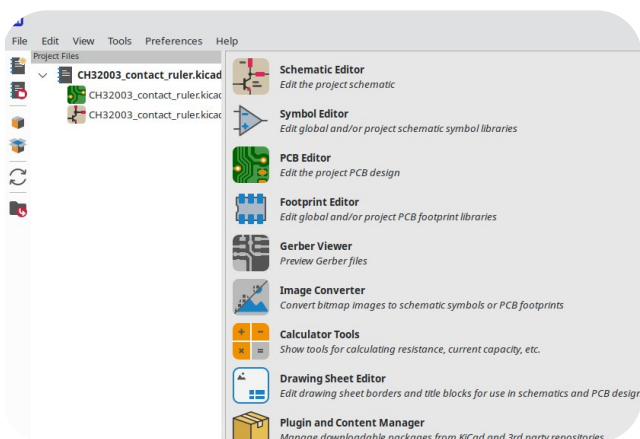
PayPal

Beginners Guide to KiCad

Opensource hardware comes in all manner of forms and made from all manner of materials, but perhaps one of the most commonly associated areas is Printed Circuit Board (PCB) design. When it come to creating opensource PCB's KiCad is the absolute go to tool. Let's dive in and make a PCB design and send it for manufacturing.

The first port of call is to install KiCad on your machine. For windows and mac users this might be easiest by going to the KiCad download section but for Linux users you might use your package manager to install. Either way you'll find information about installing for your system over on the KiCad website. One note with installing is that you'll need to also install the standard KiCad schematic symbol libraries, footprint libraries, and optionally but recommended the 3d model libraries.
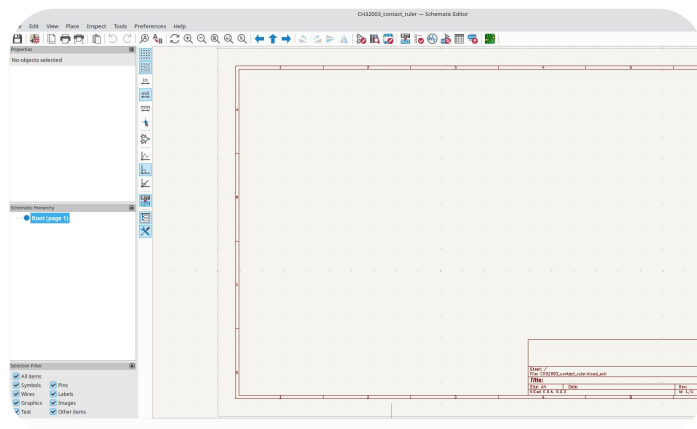

With KiCad installed open it! You'll see the main project page with 9 different icons. Each of these is kind of a separate application that deals with a seperate part of the PCB process, in fact originally some parts of KiCad were completely independent software projects.

Like many complex CAD projects there are lots of different approaches, a standard approach with KiCad is you create a project, create a schematic with symbols representing the different parts of your circuit and the connections, then assign specific components in specific packages you want to use by selecting "Footprints". These contain all the geometry, pads, holes, for that component. Footprints and their associated schematic symbols and connections are added to a "netlist" file. The netlist is then pulled into the PCB editor where you can lay out the components and the symbolic connections, shown as a "ratsnest" of white lines, are replaced with copper traces that you lay out.
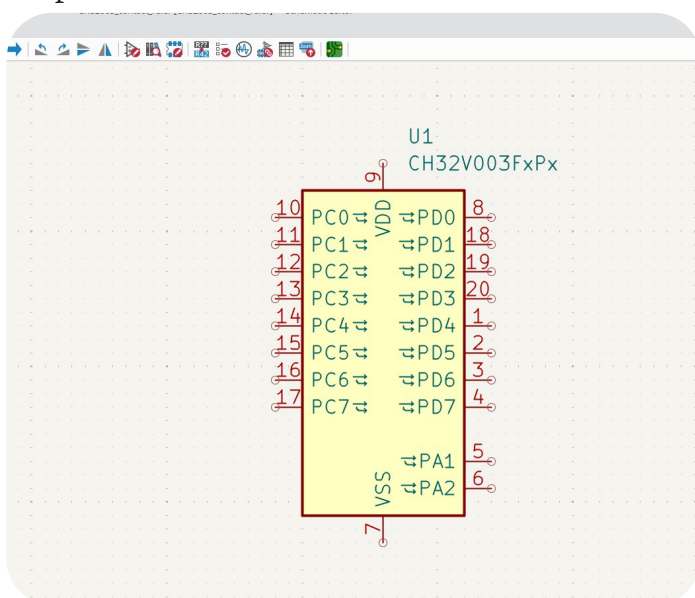
To begin this process, on the main project page click "File > New Project" and give your project a name. We are going to make a half useful, half novelty item in this tutorial, a CH32 chip breakout board built into a right angle ruler which is a useful tool, can be a business card, and is an example that you can do more creative things with PCB design!

To begin we can now double left click on the Schematic Editor icon and this will launch with a schematic already labelled for our project. You'll see that it's a blank page, a little like a technical drawing page. Let's start to add the first parts of our schematic.

We are going to add a symbol representing a CH32 chip, and the chip in real live we are targeting to use is the TSSOP 20 pin version. So let's left click on the "Place Symbols" tool icon. You should see a dialogue launch and the schematic symbol libraries will begin to populate. The first time you run this tool in a project it may take a couple of seconds to load all the library items. We can now simply type into the search area of the dialogue and type in "CH32". We should see a curated small list of CH32 related symbols and we are going to pick the "CH32V003FxPx" variant. If we double left click on this item in the list, it will be selected and the dialogue will automatically close, and we should now see that we have the schematic symbol attached to our pointer. Move the symbol to somewhere relatively central in the schematic sheet and left click to place it.
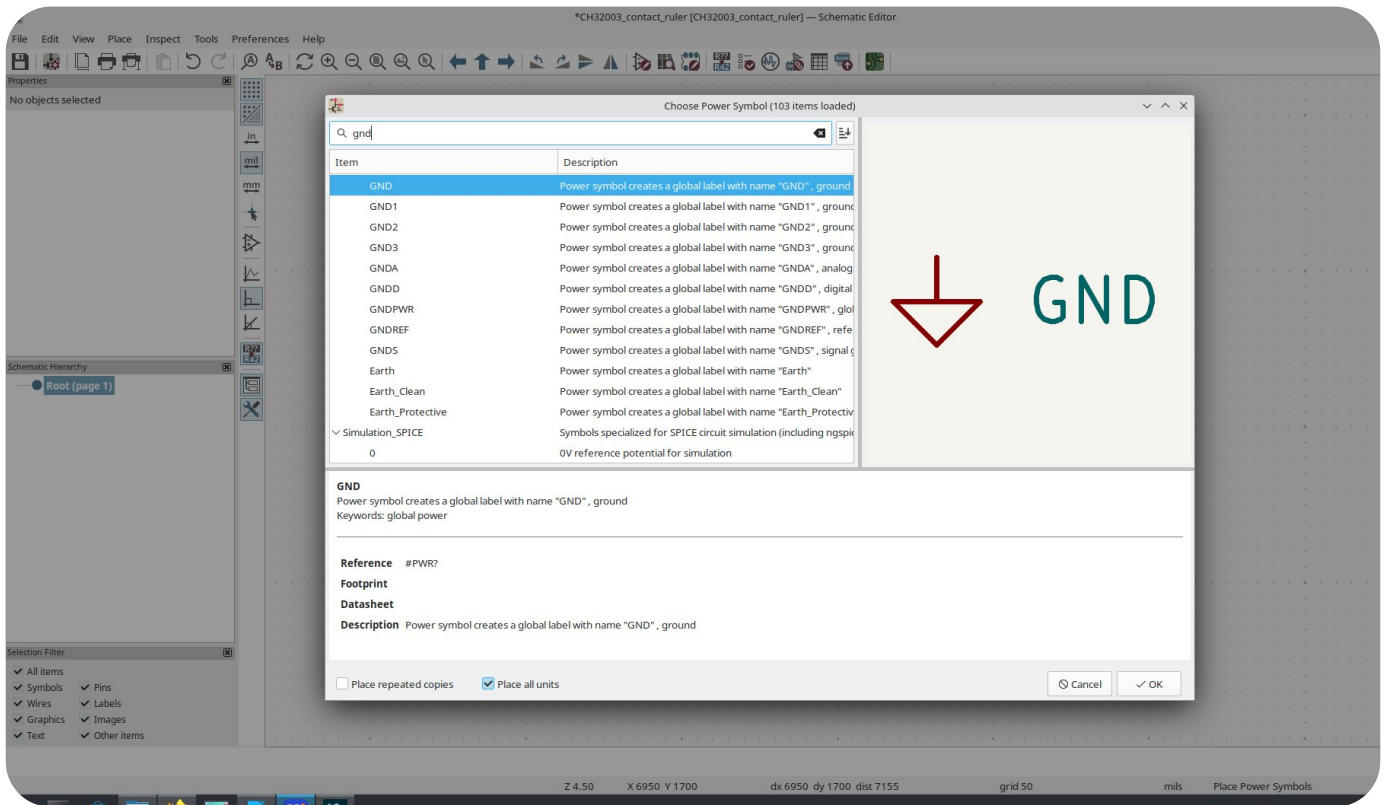

Now we have something placed in the schematic we might need to move around or zoom as we move on. Also notice that after placing our first symbol we still have the "Place Symbols" tool selected. This makes sense as often we might want to insert symbols one after another, but for now, let's click the "Select items" tool icon to return to the regular pointer. To move the Schematic page around we can right click and hold and then use the mouse or trackpad to move around the schematic. Similarly the centre mouse or touchpad button or wheel can be used to zoom in or out. However the F1 and F2 keys also act to zoom in and out, centring the zoom around the location of the pointer. This is very useful and applies not just to the schematic editor but later in the PCB editor also.

Similar to other tutorials in SOURCE magazine, we have described tools using the terms that appear when you hover your mouse over them, reading these tooltips is a great way to learn!

Our circuit is going to be quite straightforward. We are going to break every pin on the CH32 chip out to a header socket connector but we'll also add a couple of bypass capacitors on the power connector and we'll add an LED and resistor as a power on indicator. As such we need to add symbols that represent the different power net connections, in this case the 3V3 and ground. Click on the "Place Power Symbols" icon and then left click in the schematic sheet. A dialogue will launch that is similar to the symbol selection dialogue we used earlier. First let's search by typing in 3V3. Select 3V3 with a double left click and again it will select the power symbol and close the dialogue. Again left click to place the symbol in the schematic sheet. We can repeat the above process searching for "GND" and place a ground power symbol also.
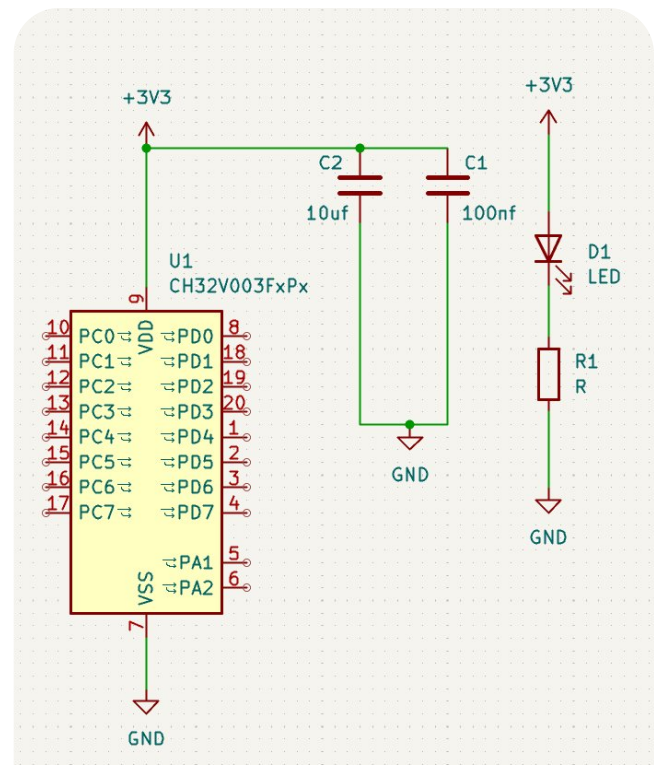
It's likely by now that you might want to reposition something. Using the "Select items" tool we can left click on an item to select it and then press the "M" key which allows us to move it. We now have the item attached to the pointer and we can move around and then re place the item by left clicking.
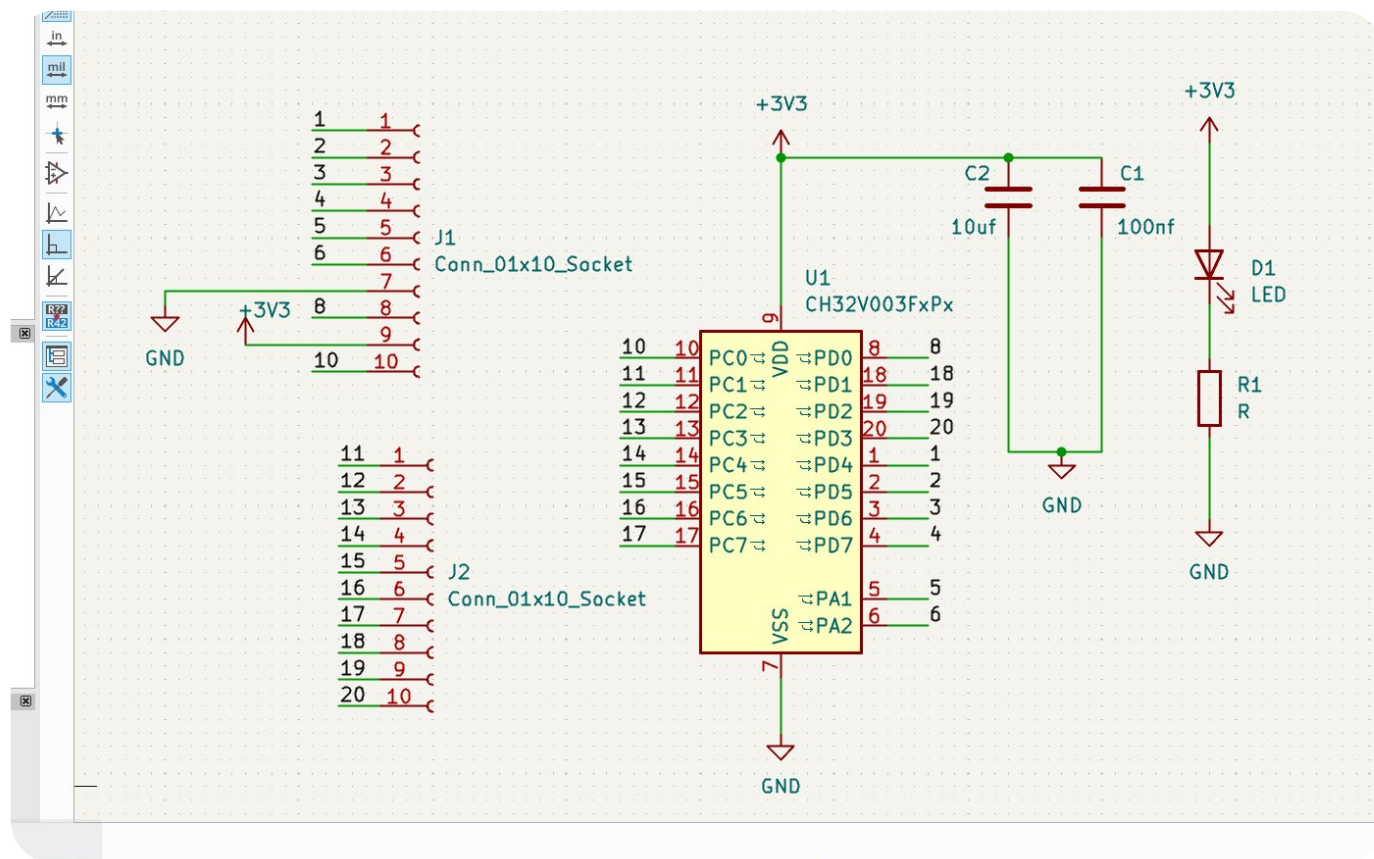
Alternatively after pressing the "M" key we can use the arrow keys on our keyboard to move the item. Each click of an arrow key will move the selected item by one grid point. Note that we can change the working units with the "Inches", "Mils" and "Millimetres" tool icons and we can alter the grid spacing by right clicking in the schematic and using the further "Grid" dropdown menu. We can also rotate schematic symbols by again left clicking to select an item in the schematic and then pressing the "R" key. Again once selected for rotation we can rotate an item by using the arrow keys on the keyboard. A left click again places the item in it's new rotated position.

Next let's add the other components in our circuit. We can use the same approach as above using the "Place Symbols" tool to add each component. For the LED we can simply search "LED" in the dialogue and for a resistor we need only search for "R" and for a capacitor we need only search for "C". Note that as we need 2 capacitors placing we search for the first one and place it as per usual, but for the second we can simply left click on the screen and press "enter" on the dialogue to select the same component as we placed last. This becomes a muscle memory after a while and it's super easy to add multiple symbols of the same type of component. Finally in terms of inserting all our symbols we can search for connectors to breakout our CH32 chip too.

For this we searched for "conn 01*10" and this returns a few suitable results. We went with the Conn_01x10_socket symbol as this automatically has a footprint associated with it for a 2.54mm pin header socket. However, any single row 10 pin connector will do as a symbol as you can swap the footprints, more of that, and why that is incredibly powerful, later in the process!

With all our schematic symbols placed in the schematic and moved and perhaps rotated to our satisfaction we can now start to draw wire connections between the different pins and components. A simple place to start would be to draw the 3V3 and GND wires from pins 9 and 7 on the CH32 symbol. Left click to start a wire and pull it to the connection point on the 3V3 power symbol or the GND symbol respectively. Note that as you drag the wire around it creates corners automatically as you move. To connect the two capacitors to the 3V3 side you'll need to create a wire junction. You do this by simply left clicking to finish a wire when it is on top of the wire you want it to join. If you are successful a small green circular node joint is created. If you don't see this then the wires are crossing without a connection.

With the exception of Pin 7 and 9 the pins on the 10 pin connectors are connected in order to the 20 pins on the CH32. You can do this with direct wiring, but it can be very difficult to read the schematic with wires crossing everywhere. A really useful technique here is to use the "Place Net Labels" tool. Left click on this tool and then left click near pin 1 on the CH32. In the Label dialogue that appears type "1" and then click "OK". Note that you can call labels whatever you wish, so you could write "Pin 1" or any explanatory name, however this is quite straightforward and so it's simple enough to just use the pin number. When you clicked "OK" you have now created a label "1" which is attached to your pointer similar to when we place a symbol. Click to place the "1" label directly next to pin 1 on the CH32 symbol. Then create a small wire between the CH32 pin1 and the connector on your "1" label. Now you can either repeat the above process and create another label called "1" and place it near the corresponding pin on the connector symbol or you can use copy and paste ("control and c", "control and v") to copy the original "1" label and paste a copy near the connector. Using copy and paste means you can't make a mistake creating a second matching label, if you were using longer names and one label contained a typo you will won't have the connections you think you have. So copy and paste the "1" label and place it near the relative pin on the connector symbol and then create a small wire to connect that "1" label and the connector pin. Note that in more complex designs you may use net labels to connect many similar points together, it's essentially the same idea as when we connect multiple points of a schematic to ground power symbols etc.

Before we move on, note that we can click on the text fields of components and edit them to add useful information to our schematic. So for example we want one of our capacitors to be 10uf and the other to be 10nf. Left clicking on the field with the value "C" we can either then press the "E" key to edit the value, or in this instance we can double click on the "C" to launch the same edit menu. We can replace the text in this field with the values we want to describe. Also, handily, we can select these small text fields and use the M and R keys to reposition the item for readability.
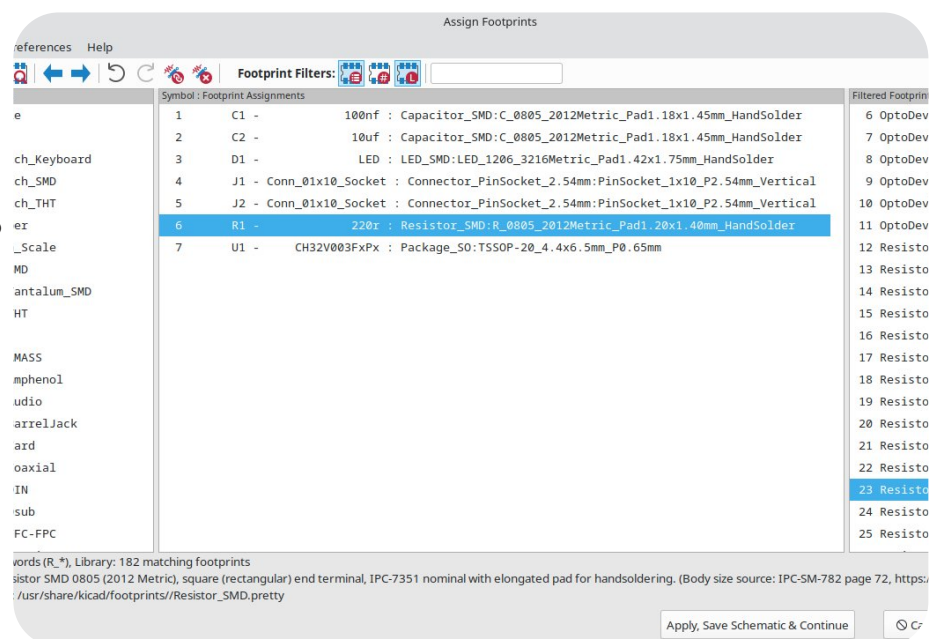
You should now have enough knowledge to create this complete schematic, creating lots more labels for pin connections, wiring and adding perhaps more power symbols.
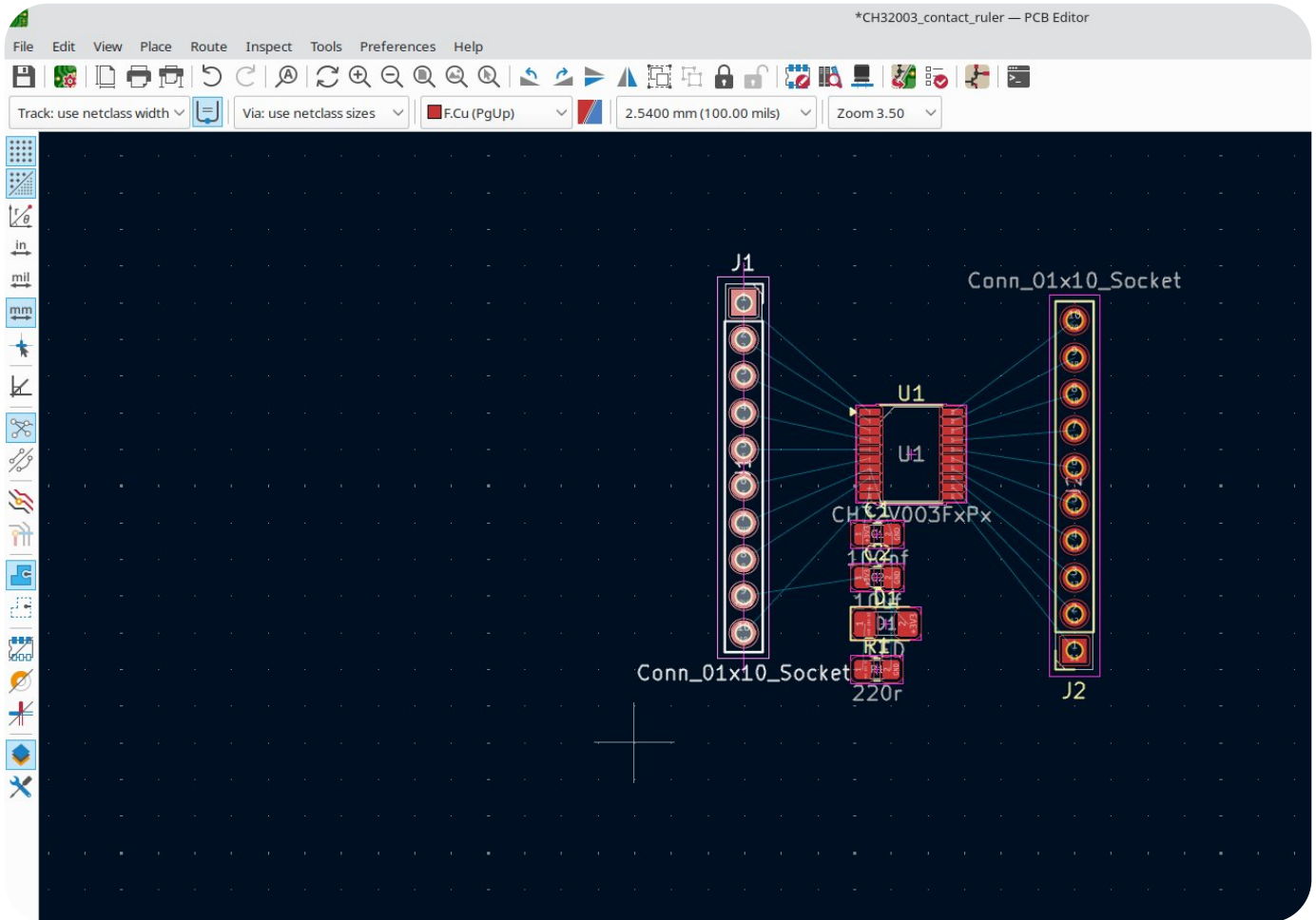
Every now and again it's an excellent idea to click the save icon to save your progress!

On a small schematic such as this one KiCad should have automatically annotated your components as you placed them. So for example as we have two different capacitors and these are labelled C1 and C2, similar to J1 and J2, our two 10 pin connectors. On larger projects you might need to click the "Annotate Schematic" tool icon to get it to flush through and re annotate after significant changes.

The next job is to associate PCB footprints we want to place into our PCB with our schematic symbols. To do this we use the "Assign Footprints" tool.

Similar to the first time loading the schematic symbol library the first time you load the built in footprint libraries it may take a few seconds. Once they are loaded you should see a dialogue box where in the centre is a list of all the schematic symbols in your project. You'll notice towards the centre at the top of the dialogue is a search input box and to the left of this are three tool icons. These tools are specific ways of filtering results. The icon on the left of these three is "use symbol footprint filters" selecting this will immediately add the footprints that are baked into the symbols we have placed, so for example clicking this will add the correct 20 pin TTSOP package for the CH32 symbol. In our case this is really helpful, but also know that you can assign a different footprint to any schematic symbol, so you aren't completely tied to the baked in suggestion. You can experiment with all three filters, but we also used the right hand, "filter by library" filter for this project. This means that for example when you left click to highlight our connectors, J1 and J2, in the list the right hand column of footprints will be filtered to just the connector library components. You can scroll through to then select the footprint you like, in our case a 2.54mm pitch single row ten socket pin socket in vertical orientation. As you highlight footprints in the right hand column you can click the "View selected footprint" icon if you want to check the appearance of the footprint. We finished off our netlist by adding hand solderable footprints in larger SMD packages. Once you have selected all your footprints you can then click "Apply, Save Schematic and Continue".

We can now start to lay out our PCB design. Either in the main project page, or from the Schematic editor we can launch the PCB Editor. You'll notice that it looks similar in some ways to the Schematic editor, although by default the PCB editor is in a dark theme. You'll notice that the background of the PCB editor is dotted and these dots are a user adjustable grid. We can adjust the grid density by using 4th dropdown from the left at the top of the page. Let's initially set our grid to 2.54mm, the common 1/10th of an inch spacing used in many through hole components. We are next going to click the "Update PCB from Schematic" button and then click "Update" in the dialogue that appears. This reads the netlist and imports the pile of footprints we associated with the schematic symbols, as well as a "ratsnest" of fine blue lines connecting the circuit together. These lines are of course symbolic, and the game now is to layout the components and then lay copper traces and flooded copper areas to replace these lines.

Similar to the Schematic editor we can select footprints by left clicking on them, note that you can also select parts of footprints, so it's worth just having a little practice of selecting components.

Sometimes when items are stacked on top of each other KiCad will offer a list of items so you can select a specific item in the pile. You can also use the "M" key for moving and the "R" key for rotation of selected items, and note that you can move items a single grid value by using the arrow keys.
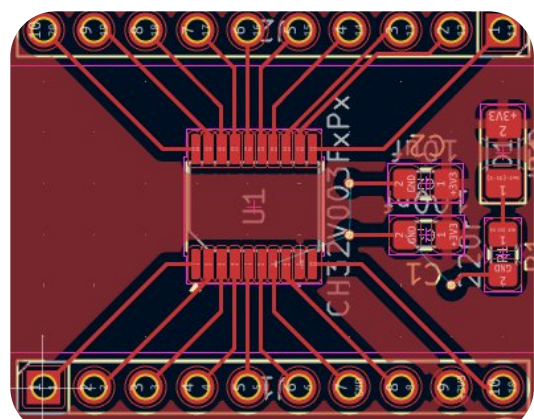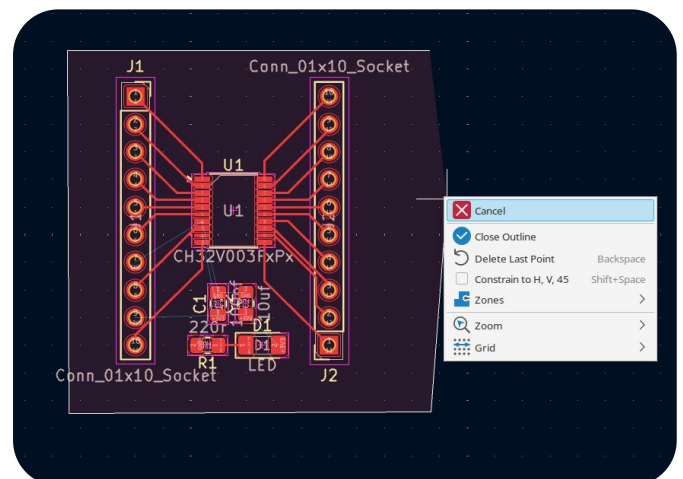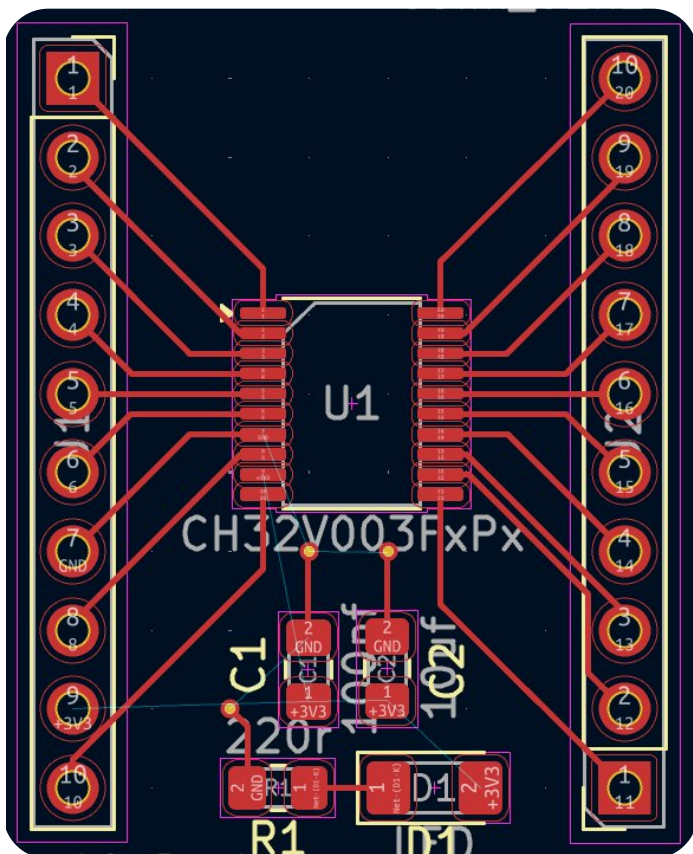
We can begin to organise our PCB by box selecting the entire pile of component footprints and dragging them in to the centre of the page. Next we can grab individual components and move them apart a little so we can begin to see what component is what. Having set the grid to 2.54mm this is perfect for us to begin by placing the two large 10 pin connector footprints. Keeping the grid at 2.54mm whilst we do this is useful as we can make the distance between these components multiple of 2.54mm which means that they will align with an electronics breadboard in the future. We spaced our connectors 7 steps of 2.54mm apart. We then moved and rotated the CH32 chip so that the pins were connecting to the correct connectors on each side so laying the copper traces would be straight forward. Similarly we laid our capcitors and our LED and resistor at one end of the chip with plenty of room around them.
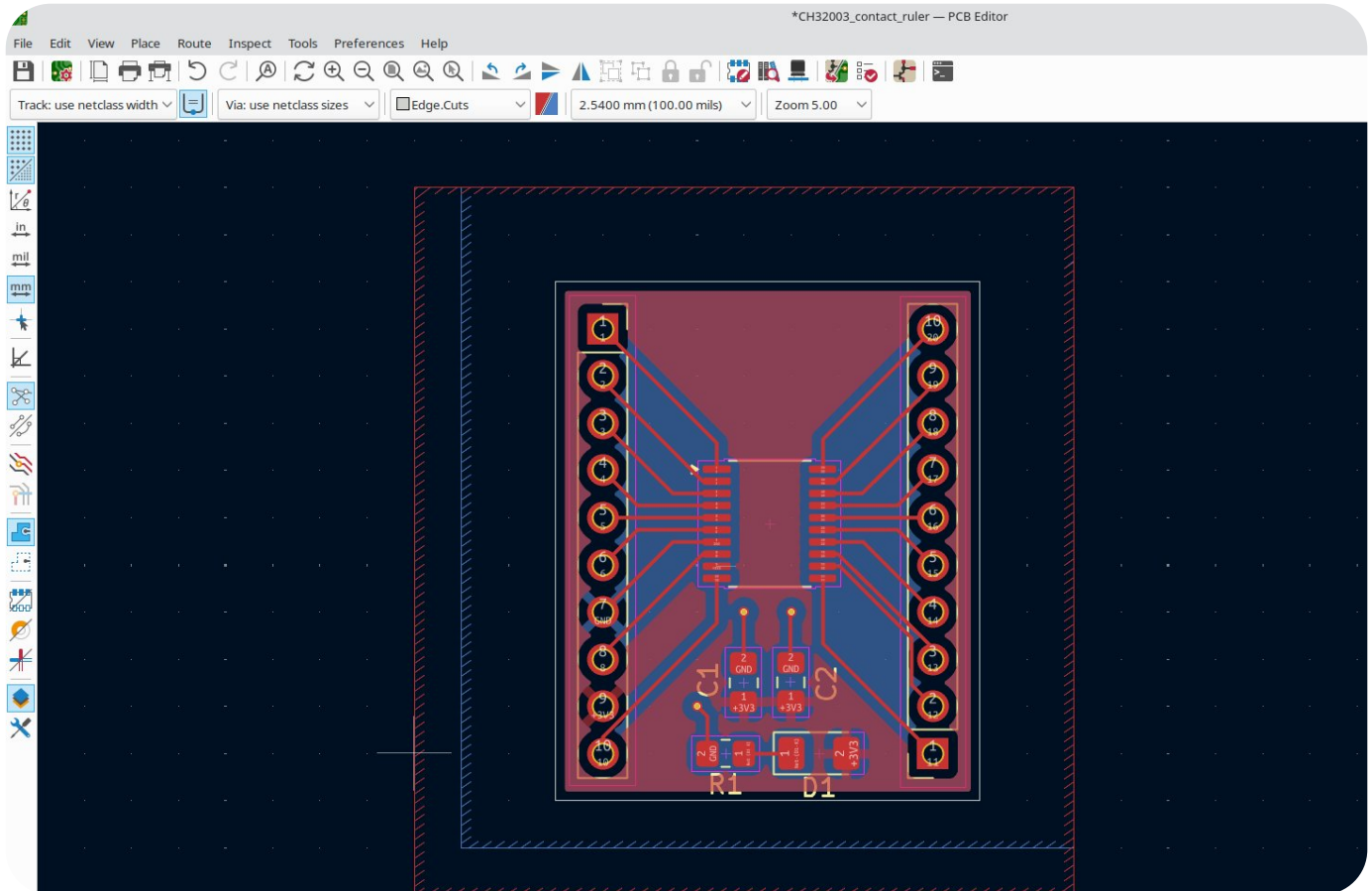
Note that later we will connect all the $3V3$ connections and the GND connections with flooded copper areas, so we don't need to think as much about laying individual traces to connect these.

Once you're happy with your initial component layout we can begin to use the "Route Single Track" tool to create the traces between the components. You left click on the centre point of either a SMD pad on the CH32 or other small components, or in the centre of a pad on the connectors. You then drag the pointer around and connect the trace to the other end of the white line to for connections. For this project we are just using the default trace thickness, but note that you can of course vary their characteristics. You can at any time right click and select cancel whilst laying a trace. Note that by default we are working on the "F.Cu (PgUp)" layer. This means we are placing traces on the top layer of copper on the board. For this design, we don't really need to place traces on any other layers as we can make most of the connections without crossing anything over. However, shortly we will use a "Via" to connect some parts of the top copper layer to the bottom copper layer. In a more complex design you can use via's to switch between layers, and continue to lay traces and can even return to the original layer with another via etc.

Continue to connect everything, apart from any GND or $3V3$ connections. To connect these we are going to create some flooded areas on the PCB. Making sure we still have the "F.Cu (PgUp)" layer selected in the layers dropdown click the "Draw Filled Zones" tool. Next left click on the page near your components in a position where you might begin to draw a rectangle that would cover all your components. When you left click you will see a dialogue appear with a search input box. Begin to search for "$3V3$" and select it from the list. This basically states that we are going to create a flooded area on the top copper layer, and this flooded area will only connect to the "$3V3$" parts of the circuit. Click "OK" and you will be returned to the PCB editor, moving the pointer you will create a line so begin to draw the first line of a rectangle around your components (don't worry if the rectangle is not perfect as this doesn't define the shape of your PCB). Left click to place the end of the first line, then drag to create the second line, repeat for the third line, finally you can either carefully draw the final line to connect to the original start point, or you can, after the third line , right click and select "Close outline" from the dropdown list which will automatically finish your shape. Then if you click "B" it will flood this area with a red colour creating connections to the $3V3$ parts of the circuit.
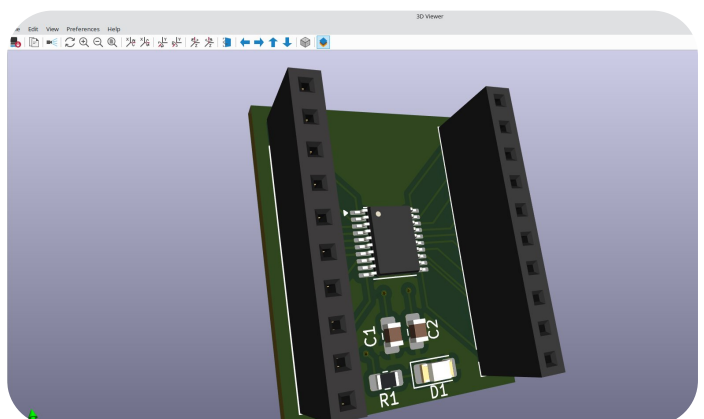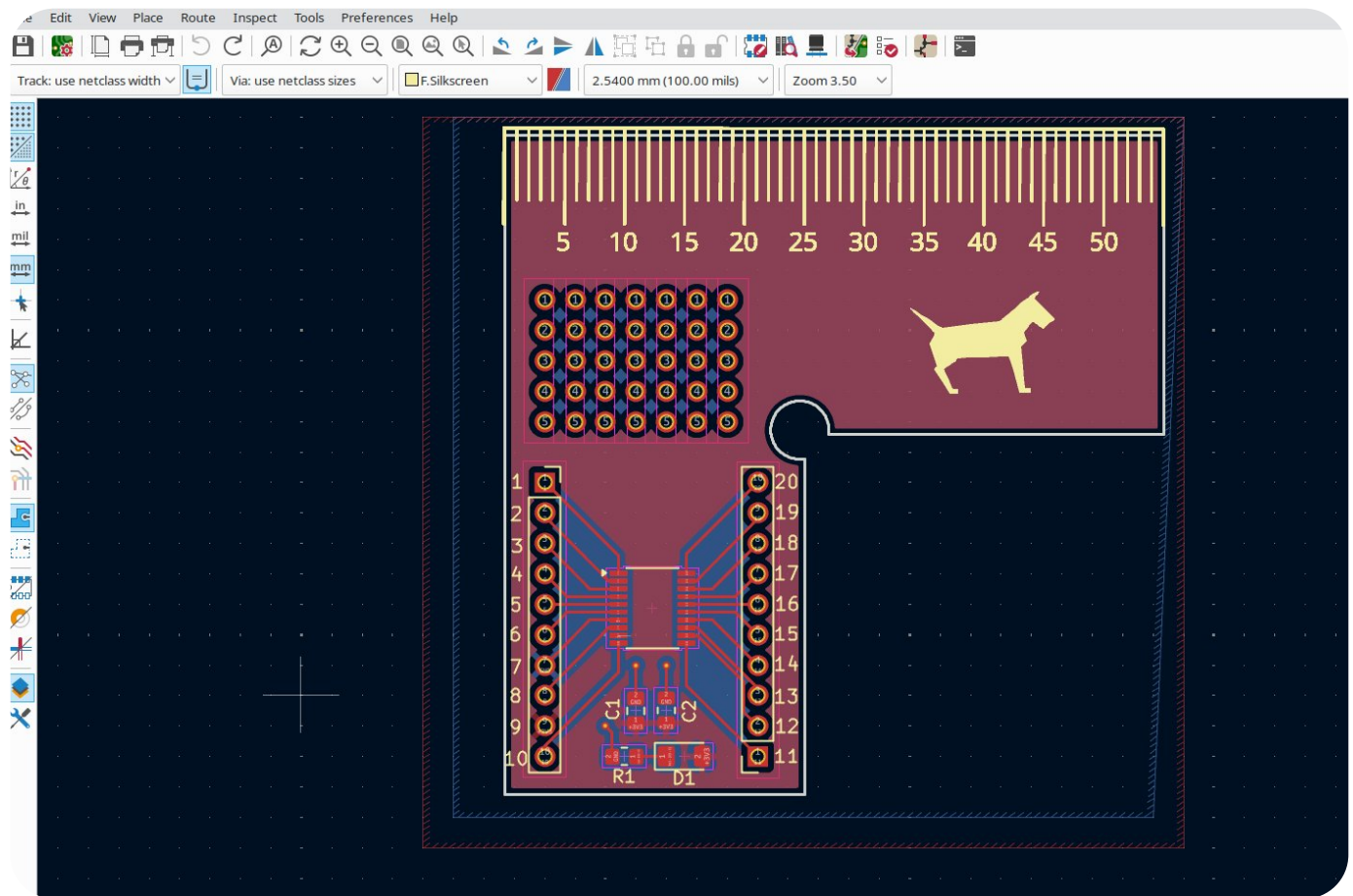
Next, switch the layer to "B.Cu (PgDn)" and repeat the above process, but this time search for and select the GND net value. When you complete the area and press "B" it will try to flood the area in a blue colour and will connect to some, but not all the GND points. Of course as this flood is on th ebottom of our PCB at the moment it can' t connect to the ground pads of the surface mount components which are placed on the top layer. To do this we need to switch back to the top layer "F.Cu (PgUp)" and for each of these SMD GND pads we need to lay a small trace out from the pad and then, without clicking we need to press the "V" key to lay a via (a tiny metal coated hole through our PCB connecting layers). After pressing V we need to then right click and select "Finish" so we don't create any traces on the bottom layer. Finally we need to connect on the top copper layer, one GND pin on the CH32 chip to a GND pin on the connector. After laying a trace and a via, or indeed making any changes which effect the flood areas, we need to press "B" to re flood the flooded areas. Using this technique you should be able to fully complete all connections on the PCB.

As ever, make sure to save your progress. Note that in our example our flood boxes are way larger than the PCB. This in fact doesn't really matter as the flooded areas don't actually define the PCB shape or edges. To do this we need to add geometry in another PCB layer called "Edge cuts". You can achieve this in a couple of different ways. The standard way is to move to the "Edge.Cuts" layer in the layers dropdown and then use the various "Draw ...." tools to create lines, arcs, rectangles, circles and more. As a quick attempt at this let's move tot he "Edge.Cuts" layer, grab the "Draw Rectangles" tool and then left click and drag to draw a rectangle that just covers our components. Having done this we can then use the "3D viewer" tool to show a quick render of our PCB where we can view the board, components and the outline of our board. You can experiment with changing the board outline in the "Edge.Cuts" layer and then relaunching the "3D Viewer" to see your changes.
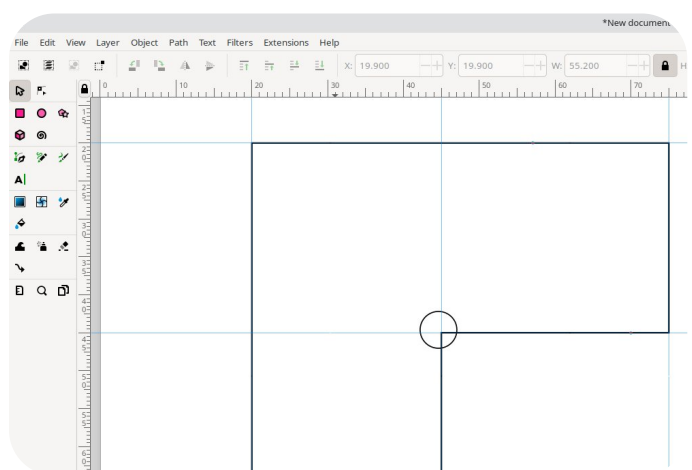
Looking at our more advanced right angle ruler project picture, you'll notice that we have added a collection of none connected pads above our main circuit layout. We've done this as a useful experimenters areas or "kludge board" area where we could solder in small items/circuits. We've also added this as it's a good example of the fact that you can add components directly in the PCB editor, without them being included, or connected in your schematic editor. To do this we can simple click the "Place Footprints" tool icon. Then, similar to our symbol placing in the schematic, we can search for a desired footprint and place it in the PCB. We simply 7 x 5 pin 2.54mm spacing vertical header pin footprints, making sure to space them correctly with each column 2.54mm apart.

We mentioned there are other ways to create edge cut geometries. This includes that you can import vector graphic items, (SVG's, Dxf's etc) into layers of your PCB. As a personal note I have used KiCad for a long time, and the internal KiCad drawing tools have increased in capability and quality a lot since the early days, however I am reasonably proficient in Inkscape and so, for me, it is easier (lazier) to use those tools and import SVG's created in Inkscape to

KiCad layers as edge cut geometry, or indeed as silkscreen decoration. As a final part of this tutorial we'll look at this a little, but we are including in this issue of SOURCE a separate article about how to create the ruler drawings in Inkscape and edge cuts that we used in this KiCad project. So we aren't going to fully walk through completing this KiCad project, but rather show you how you could!

We can import SVG to the edge cut layer (or any layer for that matter) by, from any layer in the PCB editor, selecting "File > Import > Graphics". In the resulting dialogue we can then browse to the desired file and we can select the destination layer.

For edge cut geometry we would definitely create the geometry in Inkscape, or other vector drawing package, at the correct scale for the project, but, for more decorative imports, you might use the "import scale" option to scale an existing vector design. So for example in our complete right angle ruler project we wanted to add our Concretedog logo which we have as a file with the logo at around 100mm length scale. Adjusting the import scale, and setting the destination to "F.Silkscreen" allows us to use the existing file without resizing it in Inkscape. Finally, we can use the KiCad text tools on the F.Silkscreen layer (and the B.Silkscreen later) to add text items. It's pretty straight forward, you click the "Draw Text" tool and then click on the PCB area, in the dialogue you can insert text, adjust the size and font etc and then click OK to place. You can then use the "M" and "R" key's as usual to move and rotate, and the "E" key to launch and edit the text item as needed.

Once you are happy with your board, your edge cuts, and your silkscreen and graphic items, you'll probably want to get your board manufactured. KiCad makes it pretty easy to export gerber files and drill files that are needed for production. Note that you don't always need to do this, some service, including the excellent and opensource friendly OSHPARK PCB service, you can just upload your KiCad pcb project file and it will render in the website and you can check it over and place an order. Other common services include JLCPCB and PCBWAY, these service now will receive the gerber and drill outputs from KiCad in a zip file so you

don't need to make any adjustments (there are a LOT of options in gerber creation). If you find your KiCad gerber files don't quite render correctly with a PCB service the PCB service will almost certainly have technical information about specific settings they require for gerbers and drill files, take your time to check them out, and adjust your export settings accordingly. As a starting point though, click "File > Fabrication Outputs > Gerber" and in the resulting dialogue create a new folder for your gerber output and then click "Plot" and you'll see some gerber files listed when created, before you close this window though, click "Generate Drill Files" to also add the drill file into your folder. Zip that folder and it should upload to most PCB services for you to check in a web browser before placing an order.

Well done! If you've made it this far you should have all the skill needed (after reading the Inkscape tutorial elsewhere in this issue) to completely replicate the right angle ruler project. Or indeed, you can find this project on the SOURCE issue 3 Repo here!

Pay what you feel via Paypal here
P PayPal

Pay what you feel via KoFi
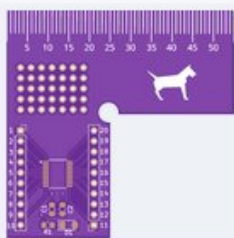Support me on Ko-fi

---

Product Details   **Price**

2025-09-18 10:10:32 | W2025091817102823 | PCB

Merchandise Total: $22.44

Shipping Charge: $13.53

Customs duties & taxes: $7.19

Order Total: $43.16

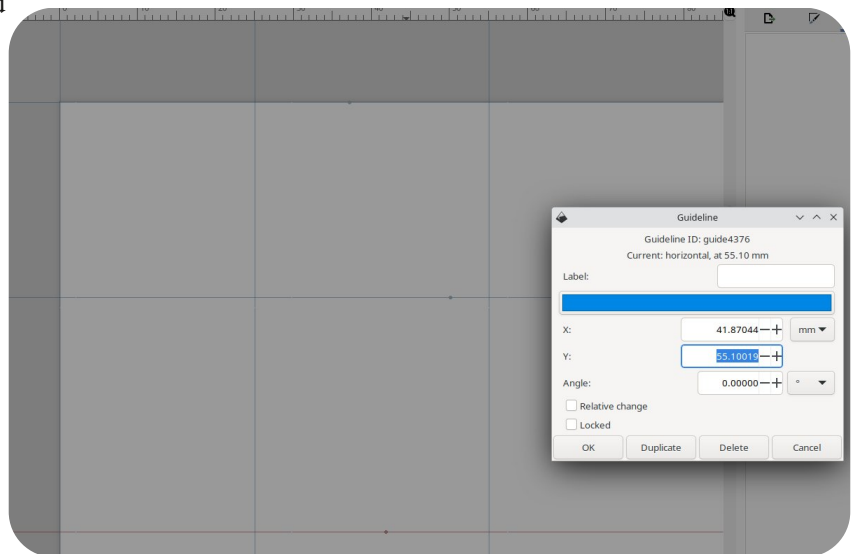| PCB | Small-batch PCB | $22.44 |
	Order #: P7-5962046A	
	Build Time:5-6 days	100pcs
	Product Details	

Inkscape for KiCad Projects, Edge Cuts and Silkscreen Rulers

Earlier in this issue we looked at KiCad and we used the KiCad tools to quickly draw the geometry that defined the edge cuts of our PCB design. However, expanding the project we also went on to use Inkscape to draw some more complex edge cut geometry and to create a graphic of a ruler imported onto the PCB silkscreen. In this Inkscape tutorial we'll explore using guides and snapping to help us create accurate drawings and also how to use the Ruler path effect tool to, well, draw rulers!

To begin download and install Inkscape and open it and select a new document. Before we draw anything we can set up some grid lines which will make drawing our PCB edge design really trivial. To begin hover your pointer over the vertical ruler on the left hand side of the preview window. If you left click and hold and then drag you should be able to pull a vertical guideline into the document. If you release the line and then hover over it the pointer icon will turn into a hand. With the hand pointer visible double left click and you will launch a "Guideline" dialogue box. In this box we can type the X and Y co-ordinates of the point on this line.

As it is a vertical guideline we are only interested in adding a X axis value, set this to 0 and "OK" to close the dialogue. Your vertical guideline should now be on the left hand edge of your document.

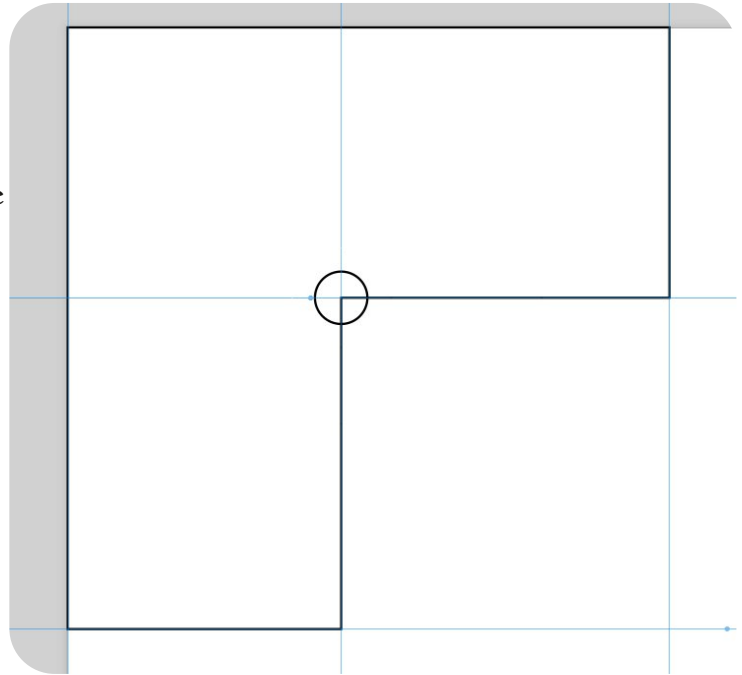Repeat the above process, but this time set the X axis value to 25mm and a third time with X axis set at 55mm. Next we are going to do the same task but with horizontal guide lines. Unsurprisingly you need to drag the guideline from the horizontal ruler section across the top of the preview area. Set a first line with the Y axis value of 0 and a second with a Y axis value of 25mm and a third at 55mm.

Next in the top right hand corner of the screen you can click to enable snapping with the "Toggle snapping on/off" icon. Left click to activate this. You can also click the snapping dropdown icon next to it and you need to make sure that the "Guide lines" option is ticked. We are now ready to begin drawing our PCB edge outline. Left click to select the "Draw Bezier curves and straight lines" tool. Hover over the lower left corner of the convergence of our grid lines on the 0,55mm point and left click. Now move the pointer to draw a line clicking at the upper left corner of the design snapping to the guidelines. Continue around the guidelines to create the right angle shape we desire. On returning to the place we began to draw click on the 0,55mm point again and then right click to finish the line. You should now have a continuous stroke along the right angle shaped path.
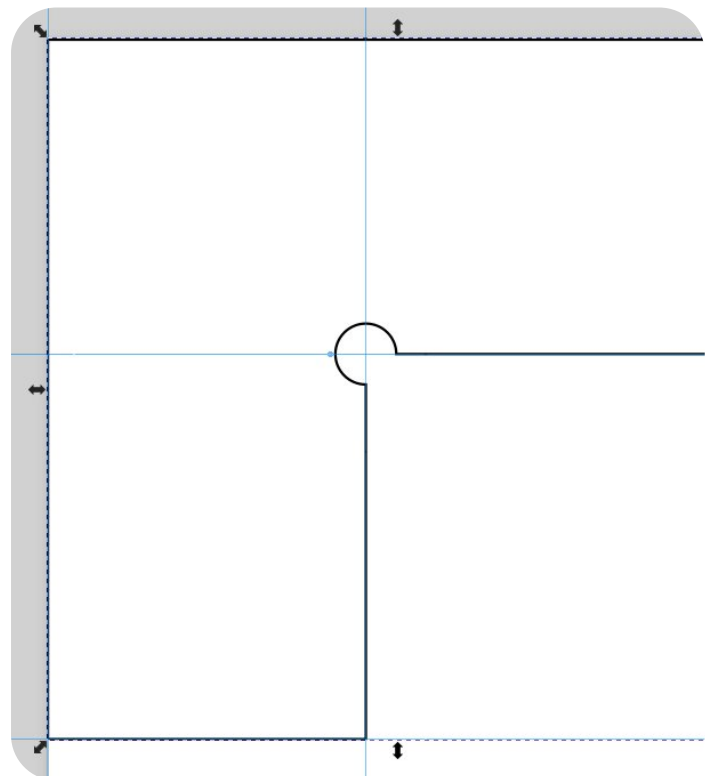
Make sure that for your drawn items the fill and stroke are set so that only a thin stroke appears. Used the "Object > Fill and Stroke" menus to access these settings.

Adding a hole on the internal corner of the design means that there will be no radius on the internal corner when our PCB is routed, this means we can slide the angle rule onto a sharp right angle object and it will fit correctly. To add this hole we are going to use the "Create circles ellipses and arcs" tool. Left click on the tool and then hold the control key and left click hold and drag to draw a small circle. With the circle selected make sure the lock icon is active and then type in a dimension of 5mm. Next, double check in the snap drop down menu that the "Centers" option is checked and then you should be able to place the circle exactly centred on the internal corner of the right angle PCB edge drawing.
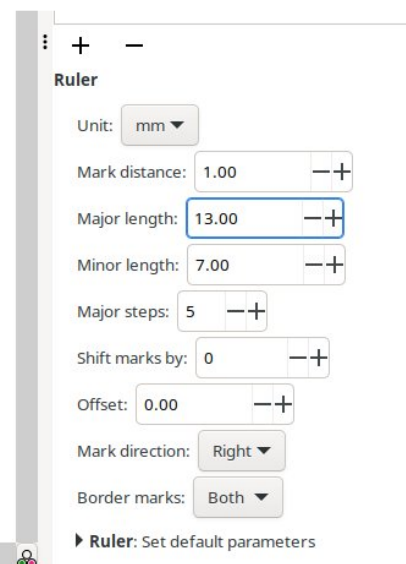
 Next press and hold the control key and then press the "A" key to select both the drawn items in the project. Then click "Path > Difference" and this should then correctly create the cutout circle in the

corner of the right angle ruler design. We can then use "File > Document Properties" and in the dialogue click the "Resize to Content" icon. This then sets the canvas to the size of our drawing. Use the "File > Save As" function to save this file with a filename leaving the filetype as the default .SVG. This file is now ready to be imported into the KiCad project onto the "Edge Cut" layer to define the edges of your PCB. The process of importing graphical elements to layers is described in the KiCad tutorial elsewhere in this issue.

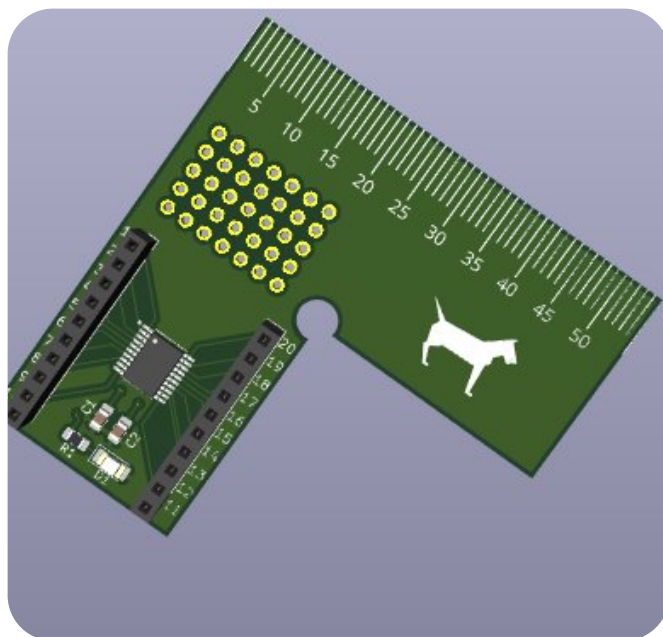To create the ruled section for our design we used a Path Effect. In a new project draw a single horizontal line using the "Draw Bezier curves and straight lines" tool. Then use the dimensions input boxes to set this line to be 55mm long.

With the line selected click "Path > Path Effects". You should see a, largely blank, Path Effects tab launch on the right hand side of your screen. In the Path Effects tab click the "+" icon in the bottom left corner of the tab. You'll see a dialogue window open which has a list of all the available path effects, either scroll to find, or use the search input area to find the "Ruler" path effect and left click it.

In the Path Effects Tab you'll now see the dialogue for the Ruler path effect and you'll see the current ruler settings applied to your path. We need to adjust the various values to create the correct ruler. First swap the "Unit" value to "mm" Set the "Mark distance" value to "1", you should now see that your line has markings every 1mm. The "Major length" and "Minor length" set the height of the marker lines with the "Major steps" setting the interval between a Major Length marker. So we set ours to be Major length of 13mm and a minor length of 7mm and then set the Major steps value to 5.

With this set up correctly we then used the text tool to add text markers for each of the major steps, starting with 5mm and ending at 50mm. Finally we need to highlight the Ruler path effect and then left click "Path > Stroke to Path", failing to do this will create some errors in the exported file. Again use control and "A" to select everything again and then use "File > Document properties" to click the

"Resize to Content" icon. Then use "File > Save as" to save the ruler design as an SVG. Again the process of importing this into the front silkscreen layer of KiCad is described in the KiCad tutorial in this issue, however the ability to make rulers and scale is useful for all manner of projects!

Pay what you feel via Paypal here

Pay what you feel via KoFi

Support me on Ko-fi

Opensource Certification with OSHWA

The Open Source Hardware Association has a free certification program which allows people from anywhere on earth to apply for certification. If successful your opensource hardware project is listed in a directory of OSHWA certified projects, given a unique OSHWA identification number and you can then use certain OSHWA logo's and your identification number on your certified project and it's documentation etc.

The first question is why certify? Certification essentially performs some checks on a project to establish that good practice in sharing something as opensource has taken place. The main are here is if aspects of a hardware project, the hardware design, the documentation, potentially any firmwares, have opensource licenses. For many, a shared project with no licence attached is problematic, it isn't clear what you can and cannot do with that project and, particularly if you want to use aspects of a project in your company, organisation or simply as a part of a personal project, you run the risk of permissions being changed or removed in the future. For many then checking an opensource project is licensed, and then how it's licensed, is the first thing undertaken.

When a project is certified successfully with OSHWA the team there have checked that all elements of the opensource project have been licensed, so not just the actual hardware, but that documentation, firmwares and other elements all have a clear license. These licensees are then stated clearly on the projects certification page. So for many, the author included, seeing an OSHWA certification number is a clear indicator that I will be able to establish exactly what I can and cannot do with an opensource hardware project.

Additionally, and perhaps somewhat personal opinion, it's a clear visible marker of being part of the global opensource hardware movement. I love sharing images of things I've made with an OSHWA UID number!

So how do you get certified?

It's really straightforward. There is an online form here where you provide links to your project and describe the licenses attached to each part of your project. As an example we can use the PCB Magnet Clamps that we designed with FreeCAD elsewhere in this issue. We created a repository on Github for this project, but, whilst this and similar platforms such as Gitlab etc, are common places to place open source projects, the actual place a project is hosted is not important in terms of OSHWA certification. We uploaded both the FreeCAD files and the STL files for our project to the repository and added a license and a readme with an image. For this fairly simple project we applied a CERN OHL V2 S license, which is a fairly common license for hardware elements, and we simply applied this license to the whole repository, so everything on that repository falls under that license.

We make mention of some opensource licenses in this article. Choosing a licence is a very unique thing you should do with care and research, don't take anything written here as legal licensing advice!

In a larger project you might want to have multiple licenses applying them to different aspects of a project. So for example a project might have the hardware design under a CERN licence, the documentation under a Creative Commons License, and some software examples, or a firmware for the project under a GPL license. So long as it's clear what is licensed and how it is licensed, and of course that the licences are allowing for open source usage in some way, then these will pass OSHWA certification.

With your project online and licensed you are pretty much ready to jump into the certification application. Over on this page https://application.oshwa.org/ click the "Get Started Button" and you'll jump into the first form section. Don't worry there are only 4 sections and they are pretty straightforward. All the

sections are pretty self explanatory, but the first section covers the basics such as selecting if you are applying as an individual, a company, or an organisation, and then some contact details, name and address, a private contact email and a public email that will be published in the projects OSHWA certification page.

The second section simply asks for common details, the projects name, the version number of the project if applicable, a brief project description, a link to the project and a category selection with common categories your project would fit into. The 3rd section enquires about what licenses have been applied to different parts of the project and what is really nice in this section is that there are clickable drop downs that give examples of projects and how they have licensed their different aspects. Note that, as in the case of our Magnet PCB Clamp application, as they are so simple, we haven't included any documentation, whilst this might be lazy on our part, it doesn't mean that the project can't be certified.

The final section is some simple terms and conditions that you must agree too. There's one that is slightly scary sounding that "I acknowledge that OSHWA has the right to enforce violations of the use of the mark. This enforcement may involve financial penalties for misuse in bad faith." this, might sound punitive, but is to stop abuse of the certification system, for example if an organisation or individual closed the source after successfully certifying with OSHWA and continued to use the OSHWA certification identifier and branding then OSHWA can take action.

CERTIFY A PROJECT

BASIC INFORMATION   **SECTION 1**

This part of the form asks for basic information about the responsible party for the project to be certified, such as name of the individual, company, or organization certifying the project and contact information for any future correspondence from OSHWA regarding your certification.

This certification is on behalf of a: Select one   ⦿

Name of Individual, Company, or Organization
Responsible for the Certified Item   Enter name

If not an Individual, name of Individual with Authority
to Bind the Company or Organization   Enter name

Contact Information

With all that squared away there's a captcha to solve and then you click the "Submit" button. Once submitted your application will go in a queue to be reviewed by a real life volunteer who will be checking over everything. At this point if anything is unclear or not correct they'll get back in touch, but all being well you will after a couple of weeks (this may vary!) notification of a successful application and a link to the projects OSHWA certification directory page.

Over on this page you can see the details of the project, but more crucially as the project owner you can grab download png/SVG files of you OSHWA certification mark and unique identification number. Notice that your ID number begins with a region/country code, It can be fun to see at the point of you application how many successful applications there are from your country, and secondly, it's fun to sometime try and work out where a rarer OSHWA country code is in the world.

Finally, wear your OSHWA certification with pride and share wildly. Know that people seeing your project with the OSHWA certification instantly know that they will have clear guidance on what they can and cannot do with your project.

MAGNET PCB CLAMPS

JOSEPH HINCHLIFFE ✉

OSHWA UID

UK000078

PROJECT WEBSITE

A small magnetic spring clamp desi
other small items for precision work
3 small 3D printed parts, a nut and l
spring.

CERTIFICATION DATE
August 07, 2025

COUNTRY
United Kingdom

Licenses

HARDWARE
CERN-OHL-S-2.0

SOFTWARE
NO SOFTWARE

Certification Mark

OSHW
UK000078

The *OSHW Certification Mark* show
makes it easy for users to quickly fir
details. This mark can be used thro
documentation, and promotional n

For more details, take a look at the

SVG   PNG

OSHW UK000078

Thanks for Reading!

Thanks so much for reading SOURCE issue 3. I can't quite believe I'm now looking at putting together issue 4! If you recall I committed to making 4 issues and I'm then looking at whether I want to continue. It's many day's of work and so far hasn't attracted massive donation amounts (Circa 90GBP across Issue 1 and 2!). However, I love the interactions around it when people get excited about something they've read here or comment that they found it useful. So on to Issue 4, I'm tentatively looking at two great opensource projects, GNU Radio and Blender, so there isn't a cohesive theme for issue 4, just cool opensource stuff!

Take good care all.
Jo AKA Concretedog.

SOURCE Issue 1 download here

SOURCE Issue 2 download here

Pay what you feel via Paypal here

PayPal

Pay what you feel via KoFi

Support me on Ko-fi