



Intro to KiCAD

PCB Design software
York Engineering Society
2024

1 Forenote

After writing this lab script I noticed that there's quite a lot of steps to designing PCBs that it's easy to forget after designing them for several years. This script was designed to be as comprehensive as possible but due to that it can seem quite daunting. Please feel free to ask questions if you are not sure on anything and if you have any questions after the lab please either email engsoc@yorksu.org or drop a message in the YES discord and we will be more than happy to help!

2 Introduction

In this lab, we will cover the foundational aspects of PCB design using KiCad, from setting up the design environment to completing the layout and preparing the files for fabrication. You will learn how to properly configure your workspace, including verifying the correct software versions and importing necessary symbol and footprint libraries. We will walk through the process of creating a schematic, placing and routing components, and conducting essential design checks such as the Design Rule Check (DRC) and Electrical Rule Check (ERC).

By the end of this lab, you will have a completed PCB design for a stylophone, a simple electronic musical instrument—while gaining a strong understanding of the PCB design process, including exporting Gerber files for manufacturing. This hands-on experience will provide you with the skills needed to tackle more complex designs in the future.

3 Setting Up the Environment

Before beginning the schematic and PCB design process, it is important to ensure that your KiCad environment is correctly configured. Follow the steps below to set up your environment and prepare for designing the stylophone.

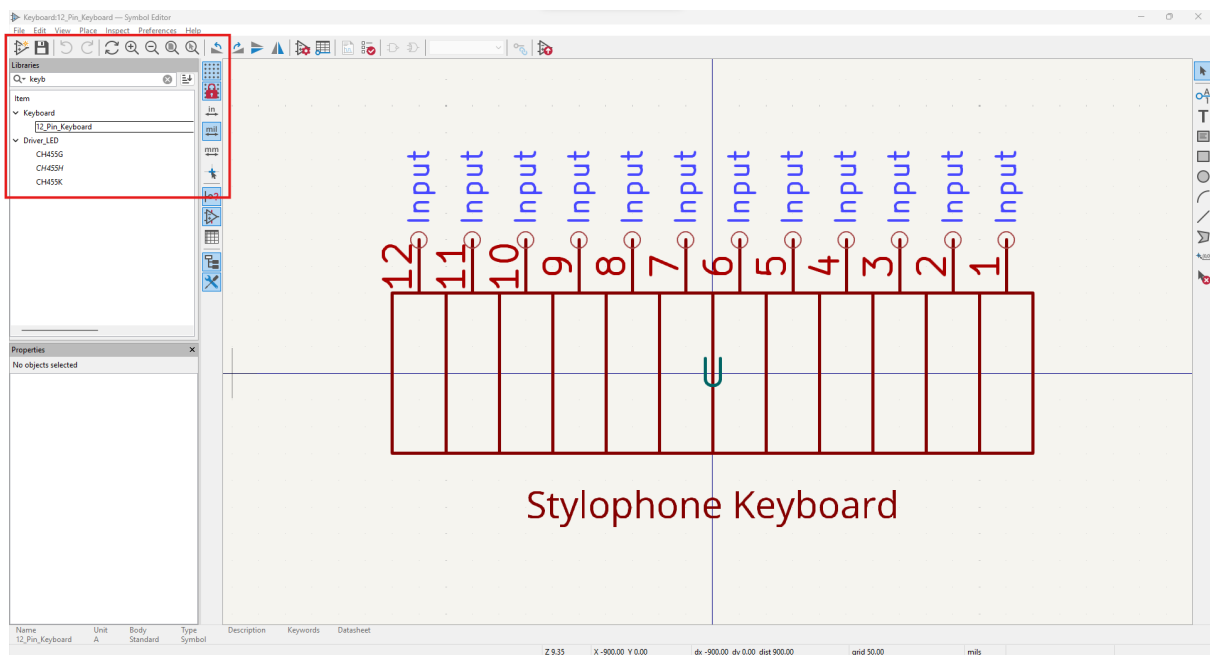
3.1 Install the Correct Version of KiCad

Some of the PCs in the lab have KiCAD installed however, if you are using a personal device or the managed PC doesn't have KiCAD, you can download the software from the official KiCad website.

3.2 Importing Symbols and Footprints

For the stylophone design, you will need to import the "keyboard" symbol and footprint library. Follow these steps:

1. Go to the Github repo and click code > **Download Zip**. Once downloaded, unzip the files
2. open Kicad, go to **file > new project**, give your project a name and save it.
3. Once the project is open, go to **Symbol Editor**
4. Click **file > Add Library > Project** and browsing to the location you unzipped the Github repo to.
5. Repeat the process in the **Footprint Editor** for the keyboard footprint (**Note: Footprints are saved as folders rather than files**).
6. Confirm that the symbols and footprints are available in their respective tools by searching for "keyboard" in the symbol or footprint search dialog.



Once your environment is set up, you are ready to begin designing the schematic.

4 Schematic Design

With the environment set up, the next step is to create the schematic for the stylophone. In this section, we will place the components from the Bill of Materials (BOM), connect them to form the circuit, label the nets, and ensure that power is correctly defined.

4.1 Placing Components

For this project, the stylophone requires the following components from the Bill of Materials (BOM):

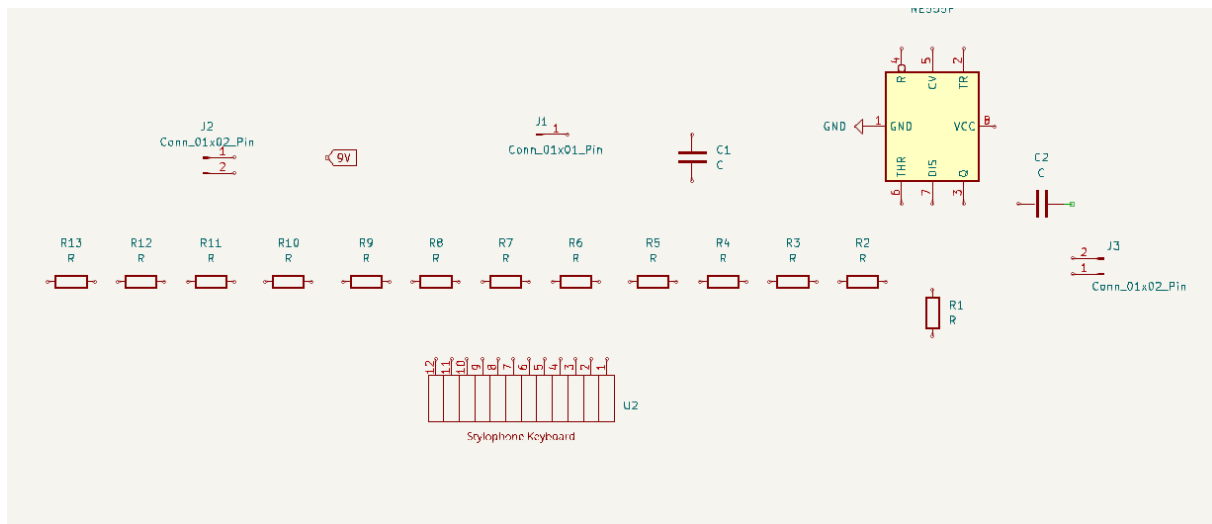
Component	Value	Type
R1	1 k Ω	Resistor
R2 - R13	30k Ω	Resistor
C1	100 nF	Ceramic Capacitor
C2	100 uF, 10V (Check Polarity)	Electrolytic Capacitor
LS1	8 Ω	Speaker
555 Timer	-	NE555P
Header Pins	-	Connector

Table 1: Bill of Materials for the Stylophone Project

To add these components to your schematic:

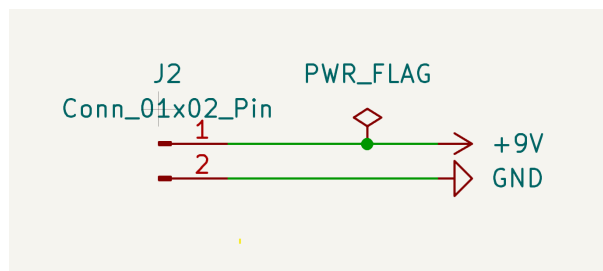
1. Open the **Schematic Editor** in KiCad.
2. Use the **Add Symbol** tool to place the components. You can search for the symbols by name in the symbol adder. For the timer, search for *NE555P*, for resistors use *R*, and for capacitors use *C*. Although components may have different values or chemistry's (electrolytic vs ceramic) all of their symbols are the same.
3. For the power, speaker, and stylus, place 2x conn_01x02_Pin and 1x conn_01x01_Pin.
4. finally place the keyboard symbol you imported before.

After this your schematic should look like this:



4.2 Identifying Power Inputs

Once the components are placed, it's important to identify the power source in the schematic. Use the **Power Flag** tool to define where power enters the circuit. This ensures that KiCad can correctly interpret the power connections. To define the power pins, click **Add Power** (below add symbol) and search for a **+9V**, a **GND** and a **PWR_FLAG**. Place these components near to one of the conn_01x02_Pin you placed before and wire them up (like in the image below).



4.3 Connecting the components

4.3.1 Wiring

Now you have sorted power out, the rest of the schematic needs to be completed. The image below shows the final schematic. Sometimes it can be useful to put notes on your schematic so others can look at it and understand how it works.

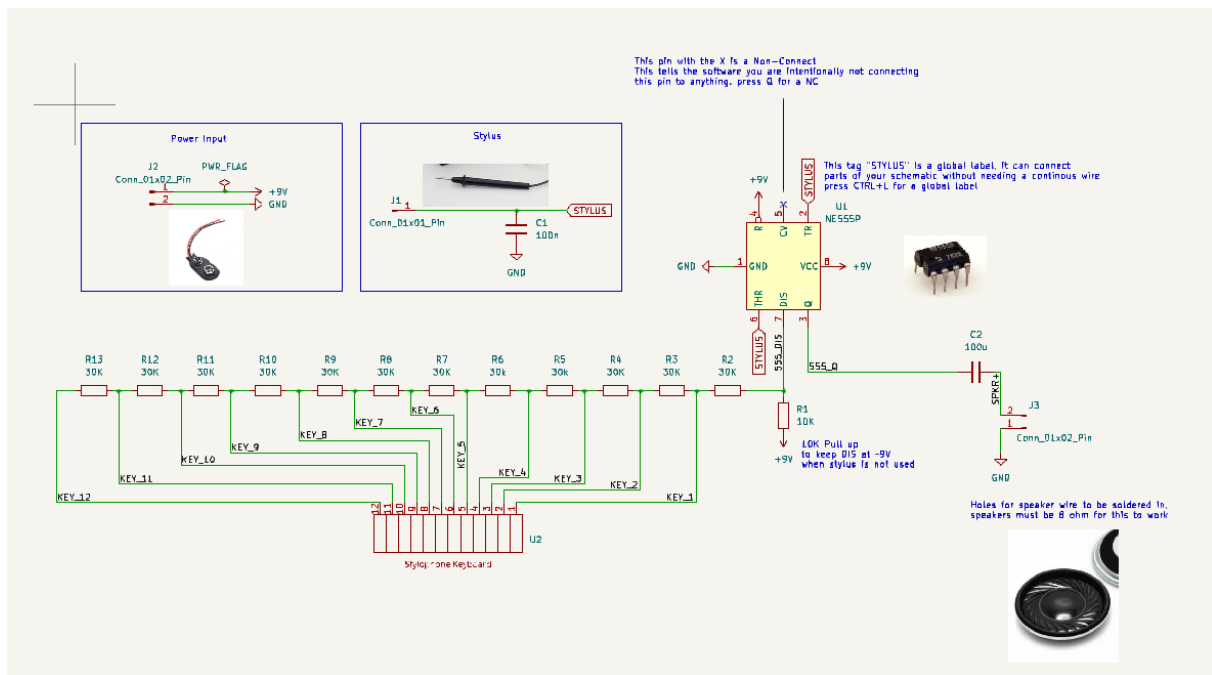


Figure 1: A full size image of the schematic can be found on the projects Github page

4.3.2 Labeling Nets

In this schematic, each connection has a label, we call these connections "Nets" as they tell the software which parts share copper pours. Labeling nets is important for a clean and easy-to-read schematic. By labeling connections (e.g., power, ground, and key signal paths) the software knows where connections need to go and ensures design rules are met. This is a bit complicated to understand at first. If you're not quite sure on how to use nets please ask (if you're doing this lab remotely please email engsoc@yorksu.org or send a message in the YES discord)

To label a net:

1. Select the **Label Tool** in the Schematic Editor.
2. Click on the wire or net you want to label, and assign a meaningful name (e.g., *VCC*, *GND*, *OUT* - Each key will also need its own label such as *Key_1*).

4.4 Assigning Footprints to Symbols

Once all of your symbols have been placed you need to assign footprints to them. It is important to take time during this step to make sure the footprints are correct as many components come with multiple footprints (we call them packages). If you are unsure on what footprint a component uses, you can check the datasheet where it will be listed. For example the NE555P package we need to use for this lab is in the 8 pin DIL (Double in-line) package.

As this is a very time consuming part to check every datasheet of every component we have provided the BOM (Bill of Materials) for this project.

To assign footprints, select the **Assign Footprints** tool.



Follow the package assignments as in the image below

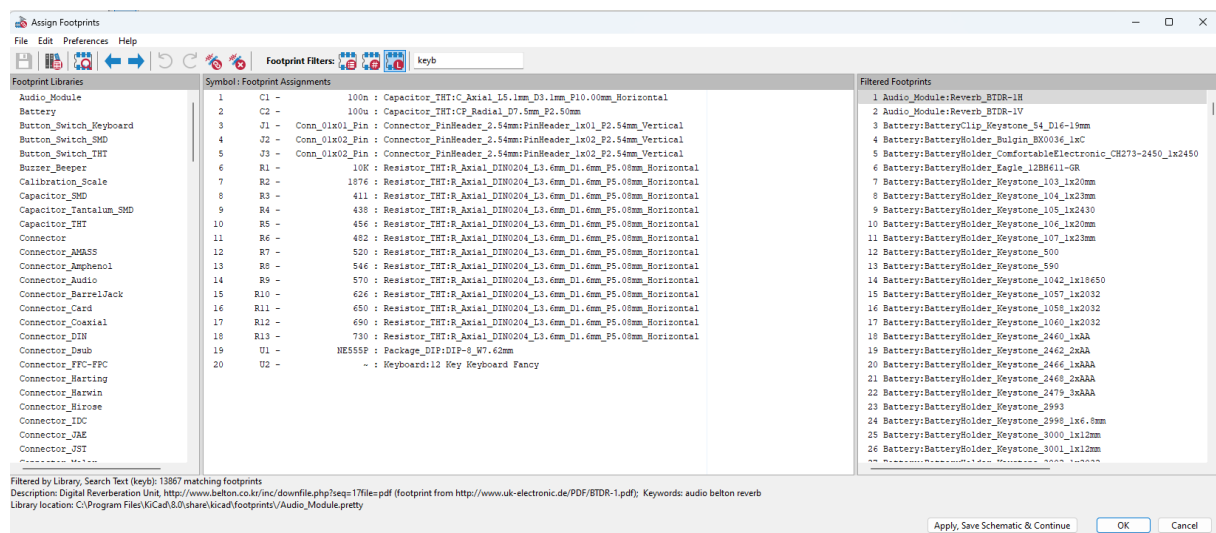
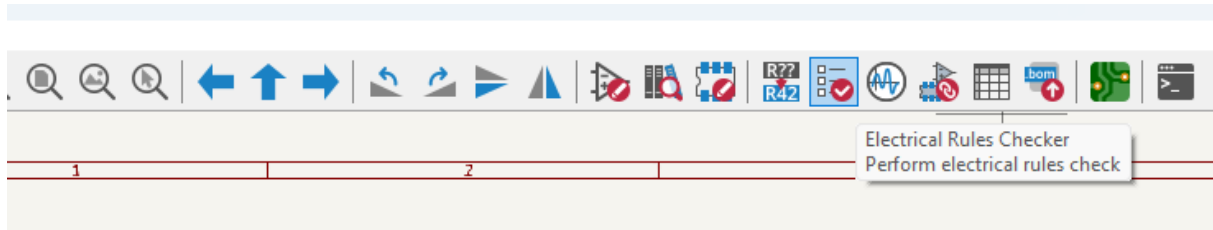


Figure 2: A full size image of the schematic can be found on the projects Github page

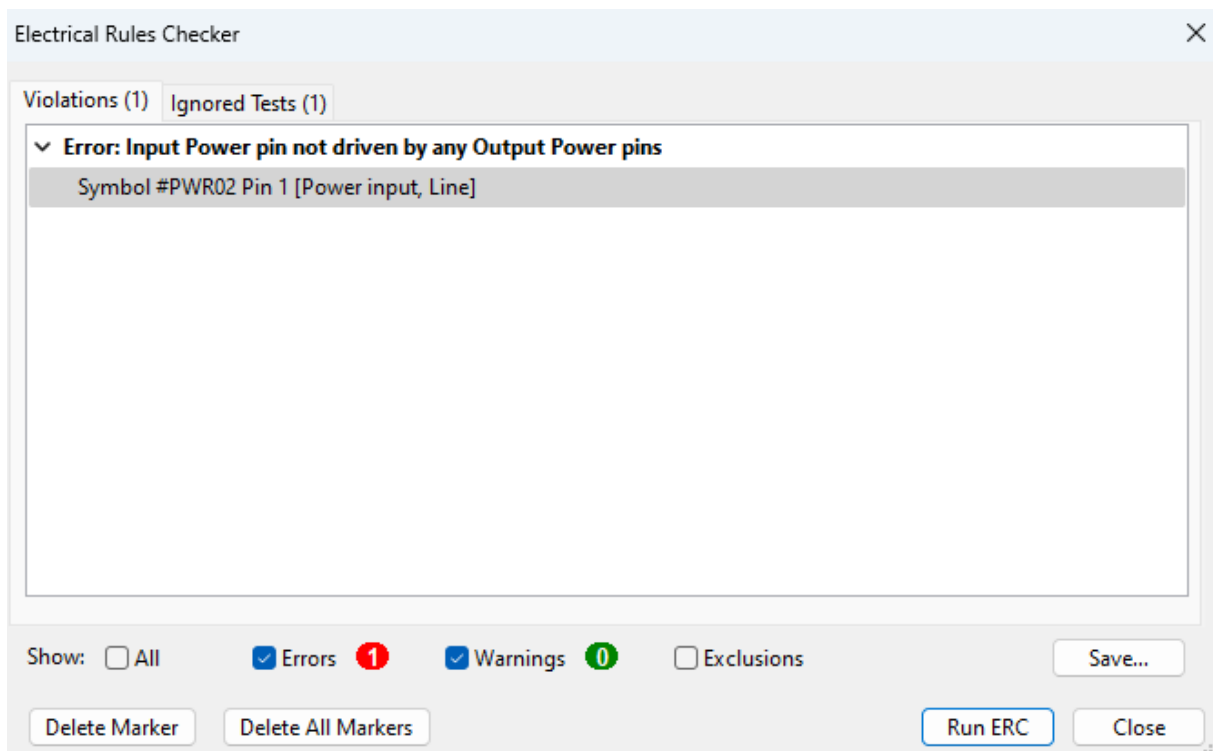
Please double check all of the footprints are assigned correctly. it is possible to fix later on but is far more difficult.

4.5 Electrical Rule checker (ERC)

The Electrical Rule checker helps you detect any issues with your schematic before moving onto the PCB layout.



When you run the ERC, it will highlight any issues with the schematic. some of these are style warning and some are errors. for example: due to the NE555P symbol having an incorrect pin type for GND in this version of KiCAD, there will be an error about an Input pin not driven by an output pin, this error can be ignored however please feel free to ask if you are unsure



Before moving onto the layout, take some time to make sure your schematic matches the one above, errors can be fixed later but its considerably easier to sort them before moving on.

5 PCB Design

For this section you need to move to the PCB Editor, this can either be done from the "Project" screen on through the schematic editor with the **Switch to PCB Editor** button

in the toolbar

5.1 Drawing the outline

For this lab we will start by drawing the board outline, sometimes you are not exactly sure how big the board needs to be but it is a good practice to have a rough idea. Board size will often dictate cost from the PCB supplier, for example; JLCPCB, the supplier we use, increases their prices for 2 layer boards larger than 100x100mm or 4-6 layer boards larger than 50x50mm.

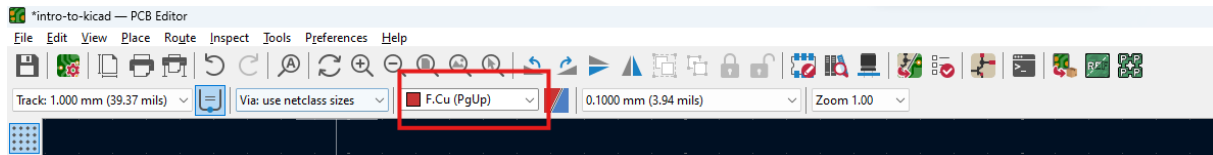


Figure 3: Layer selection drop down

To draw the board outline:

1. Click the layer select drop down (seen in Fig 3).
2. Select **Edge Cuts**.
3. Its recommended you set the grid size (next to the layer select) to 5.000mm as this will make drawing easier.
4. For this board, draw a rectangle with a width of **100mm** and a height of **60mm**
5. After drawing the outline, don't forget to change the layer back to **F.Cu**.
6. Lowering the grid size to 1.000mm will make layout and placement easier.

5.2 Importing parts

As all of the hard work associated with the parts has been done during the schematic design stage, importing parts is simple! click the **Update PCB from Schematic (F8)** as seen in Fig 4

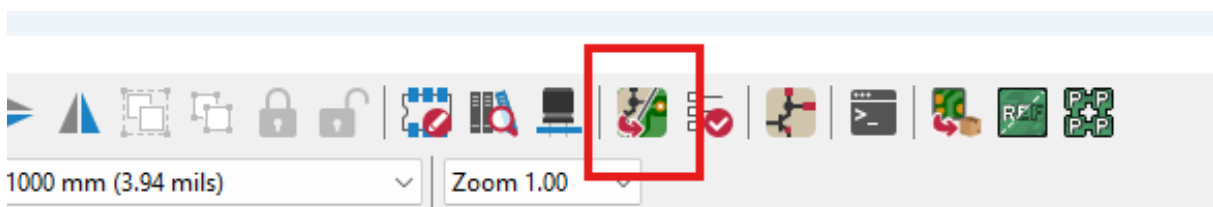


Figure 4: Update PCB from Schematic (F8) Button

5.3 Layers, Copper pours, and vias

Layers, Copper pours, and vias are integral parts of any PCB design. This section will provide an overview of each of these items.

5.3.1 Layers

The default for KiCAD is a 2 layer board. This means you have access to a front and back layer for copper pours, routing and graphics. Most PCB manufacturers support anywhere from 2-32 layers however most projects use between 2 and 6 layers. For this board you only need 2 layers.

5.3.2 Copper pours

A copper pour is a full (ideally uninterrupted) pour of copper in the pcb that generally serves as either a Ground (GND) or power plane. This is good practice for the following reasons. Solid planes have less resistance than traces which lowers the power consumption and heat dissipation requirements in high power applications. GND planes also provide a shorter return path for current making the overall circuit more efficient, if you would like more information of the use of planes have a look at [1] in the reference section of this script.

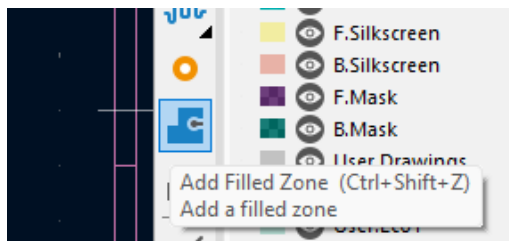


Figure 5: Fill Area (Ctrl+Shift+Z) Button

To add a pour / plane to your board:

1. Click the **Fill Area (Ctrl+Shift+Z) Button** (seen in Fig 5).
2. Click on one of the corners of your board edge and wait for a **Copper Zone Properties** box to pop up. *Note: Check your layer is set to **F.Cu** first!*
3. For this board you want both layers to have a GND copper pour. Select both **F.Cu** and **B.Cu** and under the *Nets* section select GND.
4. After pressing OK, select the other 3 corners of the board edge



Figure 6: Copper fill zone

This fill can be larger than the board outline if you are struggling to line it up exactly. Once you have completed the pour press **B** to fill the zone.

5.3.3 Vias

Vias allow you to switch between layers with traces. they can also be used to connect pins to Filled planes. This also allows you to route traces on both layers of your board which is necessary for more complex designs. To add a Via use the **Add Via (Ctrl+Shift+V) Button** as seen in Fig 7.

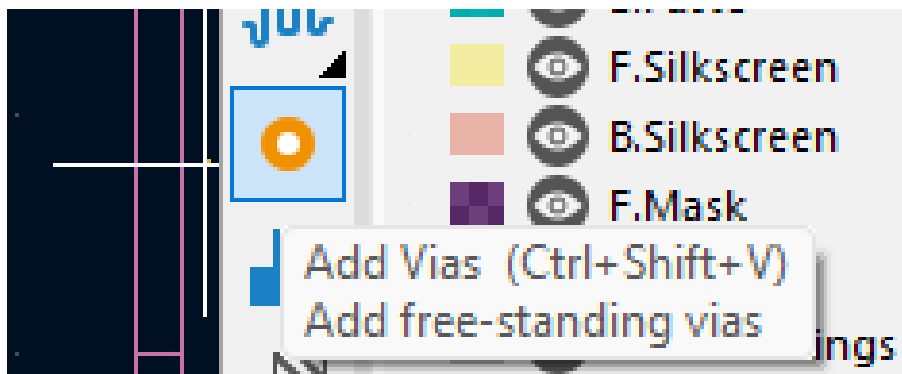


Figure 7: Add Via (Ctrl+Shift+V) Button

5.4 Layout

Finally! after all this work you are now ready to layout your board. Layout is one of the most important sections of your design as a good layout will make routing your design considerably easier. Click and drag the parts onto your board. We have provided a suggested layout however you are completely free to lay your board out however you would like. Our recommendation is a layout similar to the one in Fig 8.

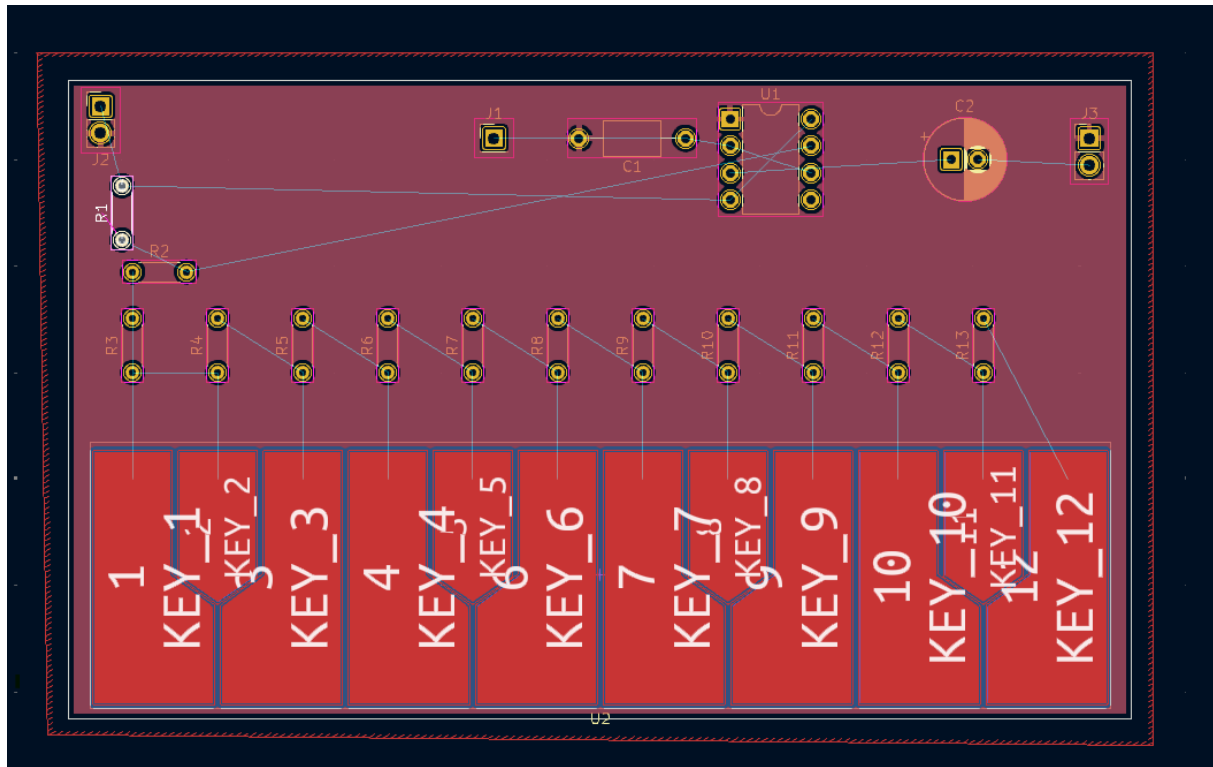


Figure 8: Recommended PCB layout

5.5 Routing

Now you have your layout, you need to connect all of the nets together. The blue lines (which are called **the rats nest**) are useful to show you where each pin needs connecting. Before beginning the routing it good practice to re-fill the copper zones by pressing **B**. this will automatically connect all of your GND connections through the copper pour.

As mentioned in the copper pour section, it is best to have an uninterrupted ground pour if possible. We have decided to do the majority of the routing on the top layer and tried to reserve the bottom layer as much as possible.

To route traces make sure the layer selected is **F.Cu** then click on a pin and drag the trace to the pin it corresponds to. Our recommended routing can be seen in Fig 9 but you are free to route your board however you would like.

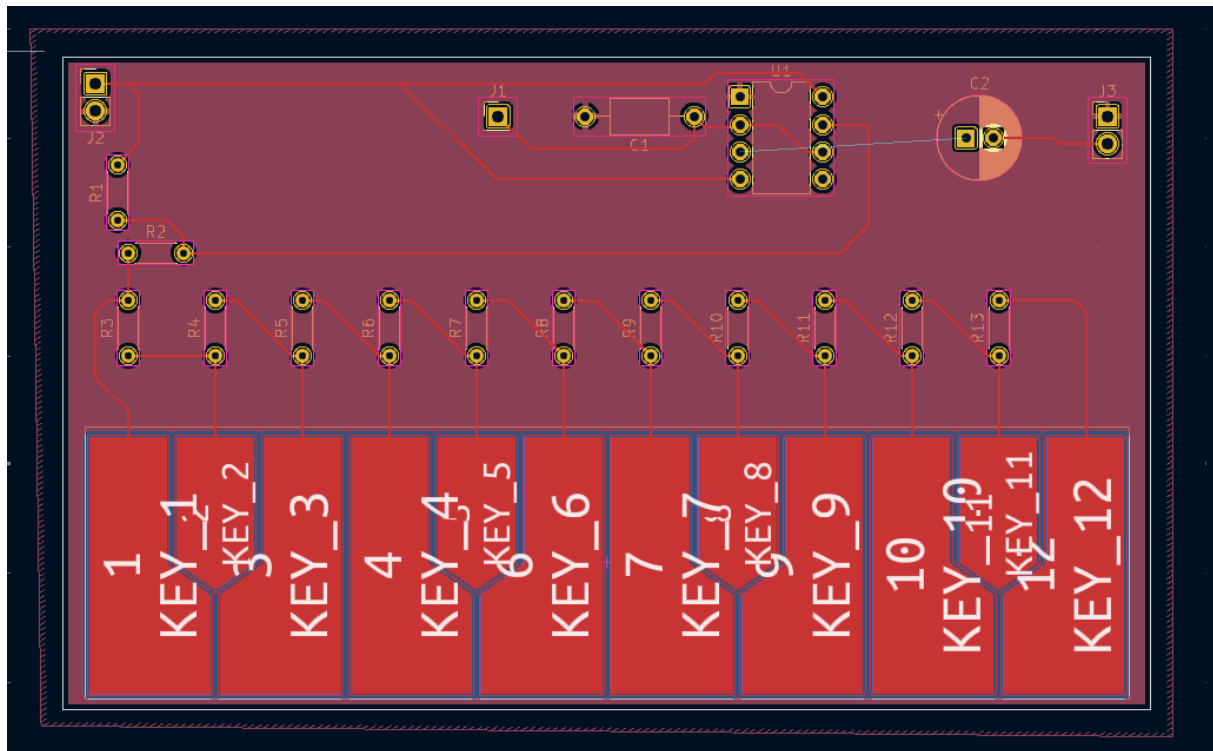


Figure 9: Front side PCB layout recommendation

As you can see in Fig 9, there is one pin left to connect but this is not possible as the pin is blocked in. In this case we can use the bottom layer to route the last pin. Use the layer selector to swap to **B.Cu** and route the final pin (seen in Fig 10).



Figure 10: Back side PCB layout recommendation

5.6 Design Rule checker (DRC)

After finishing routing, it is a good idea to run the Design Rule checker. Similarly to the ERC this makes sure you haven't forgotten to connect any pins or any of your traces are too close together. To run the DRC click the button seen in Fig 11

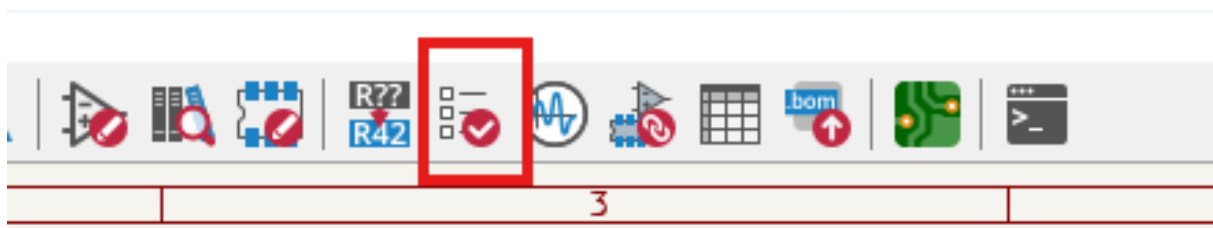


Figure 11: DRC Button

Click Run and fix any errors if they come up. if there are no errors, **congratulations!** you have completed your PCB. the next steps will outline some general advice on good practices and how to export your design.

5.7 Exporting your design

PCB manufacturers require Gerber files. with the PCB editor open:

1. Click File >Fabrication Outputs >Gerbers.
2. Leave all default settings but set an **Output Directory** or all of your gerber files will be generated into the current working directory.
3. After clicking **Plot**, it is good practice to generate **Drill files**.
4. After clicking **Generate Drill Files**, Generate the drill files and map files.

After generating all of the required output files, add them to a ZIP archive. If you would like us to get your boards produced please email this ZIP folder to eng-soc@yorksu.org

Thank you for following this lab script, if you have any feedback please fill in this form with your comments:

Google form

References

- [1] T. Williams. The ground plane: Lord of the board. *EMC Journal*, 72, 2007.