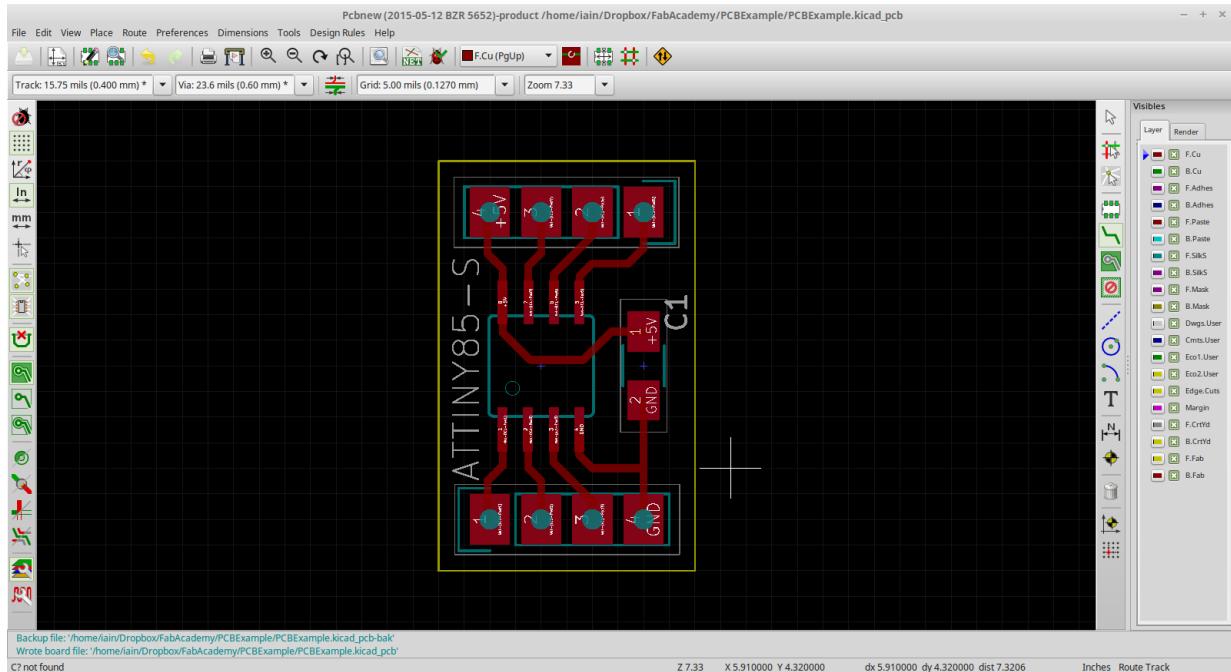


## Denford PCB Engraver Tutorial

This tutorial will lead you through using the Denford PCB Engraver to mill a small breakout board for a surface mount ATTiny85. The board was designed in KiCAD.



All the files for this tutorial can be found on my github: <https://github.com/ichalmers/Denford-PCB-Engraver-Tutorial>

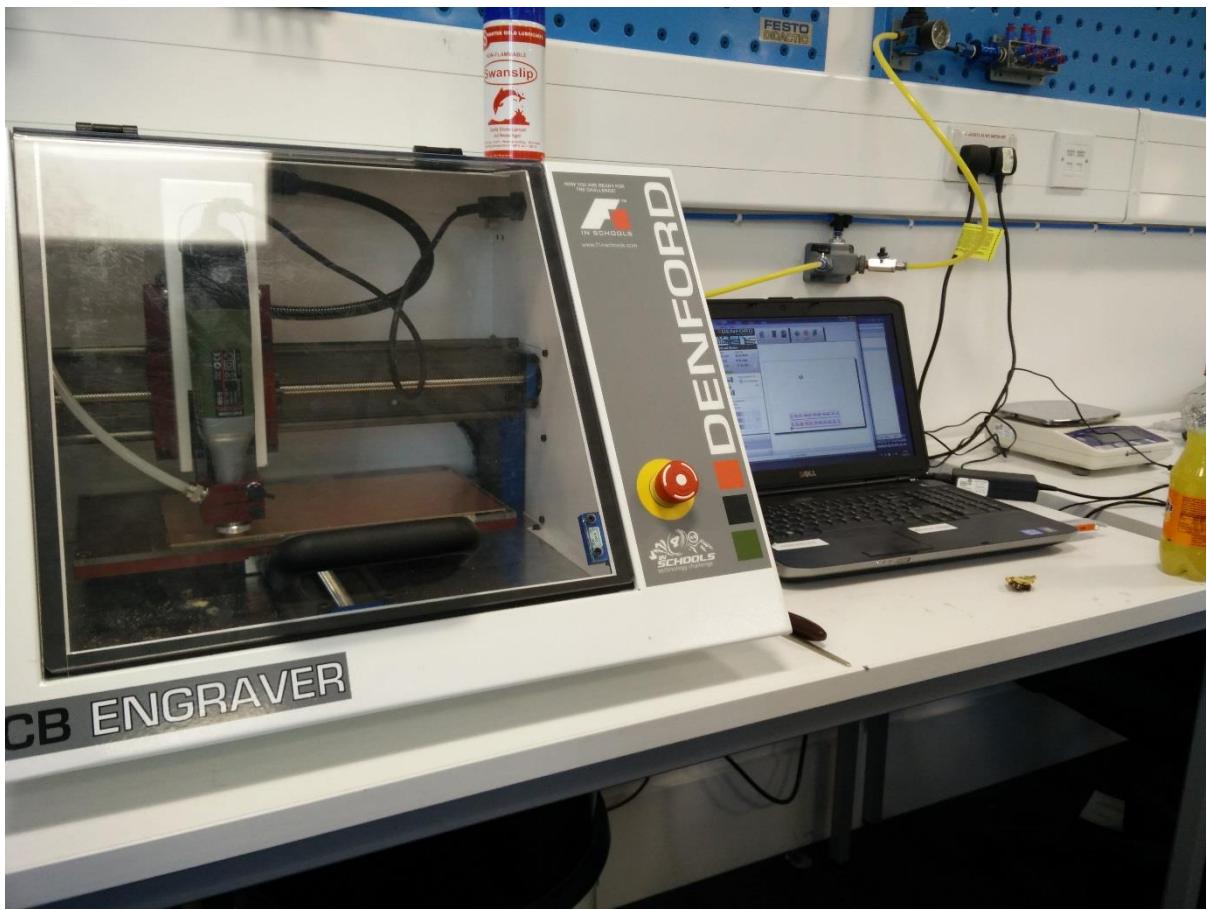
### Some points to remember before we get started:

- Think “Design For Manufacture”.
  - Endmills are 0.4mm.
    - Minimum distance between your traces should be 0.4mm.
    - I use a default trace width of 0.4mm in my designs too.
      - Helps identify old endmills.
      - Big enough traces that they are less likely to rip up.
  - 1206 SMD (surface mount) components are a good standard (and the size we stock in the FabLab at Strathclyde University).
    - Big enough to solder “easily”.
    - Can fit a trace between the pads.
- EACH ENDMILL COSTS OVER £10!
  - They are **VERY** easy to break, so take care!
- Single sided boards are best.
  - You would be surprised how complex a design can get before you really need double sided boards.
  - Double sided is possible but fiddly to align and get right, take ages to do and even longer to solder (because you need to solder wire in your vias).

- Best option is to do as much on a single side as you can, and then either use  $0\Omega$  resistors or pieces of wires to make any necessary jumps.
- **Automatic drilling of through holes.**
  - But try keep to SMD parts if you can. It's easier, faster and components tend to be cheaper.
  - Pin headers are a good exception to this – they benefit from the extra mechanical security of through hole compared to surface mount. It helps stop you ripping up traces if you are plugging/unplugging things a lot.
- **Use FR1 copper clad board.** It gives the best cut and has the least wear on the endmills.
  - FR2 is OK. Most of the stock in the lab is FR2 purely because it is easier to source.
  - **DO NOT USE FR4!** It wears out the endmills because it is so hard, and the dust is **TOXIC**.
  - Basically, only use copper clad board supplied by the FabLab...

## Loading The Design

Turn on machine (red switch on the side) and plug it into the laptop. (Not all laptops have the software, check the chart on wall).

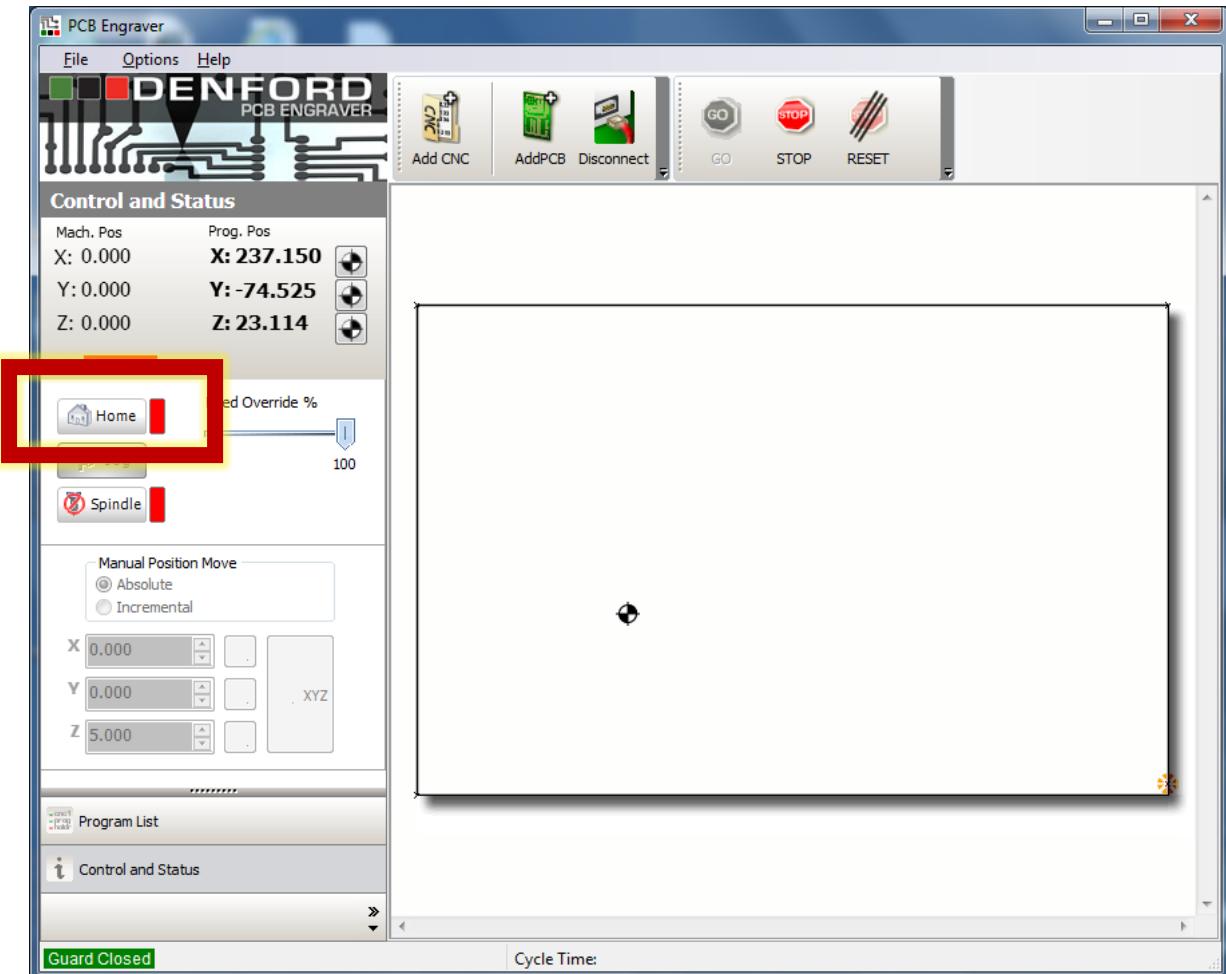


Open PCB Engraver software. The icon is on the desktop.

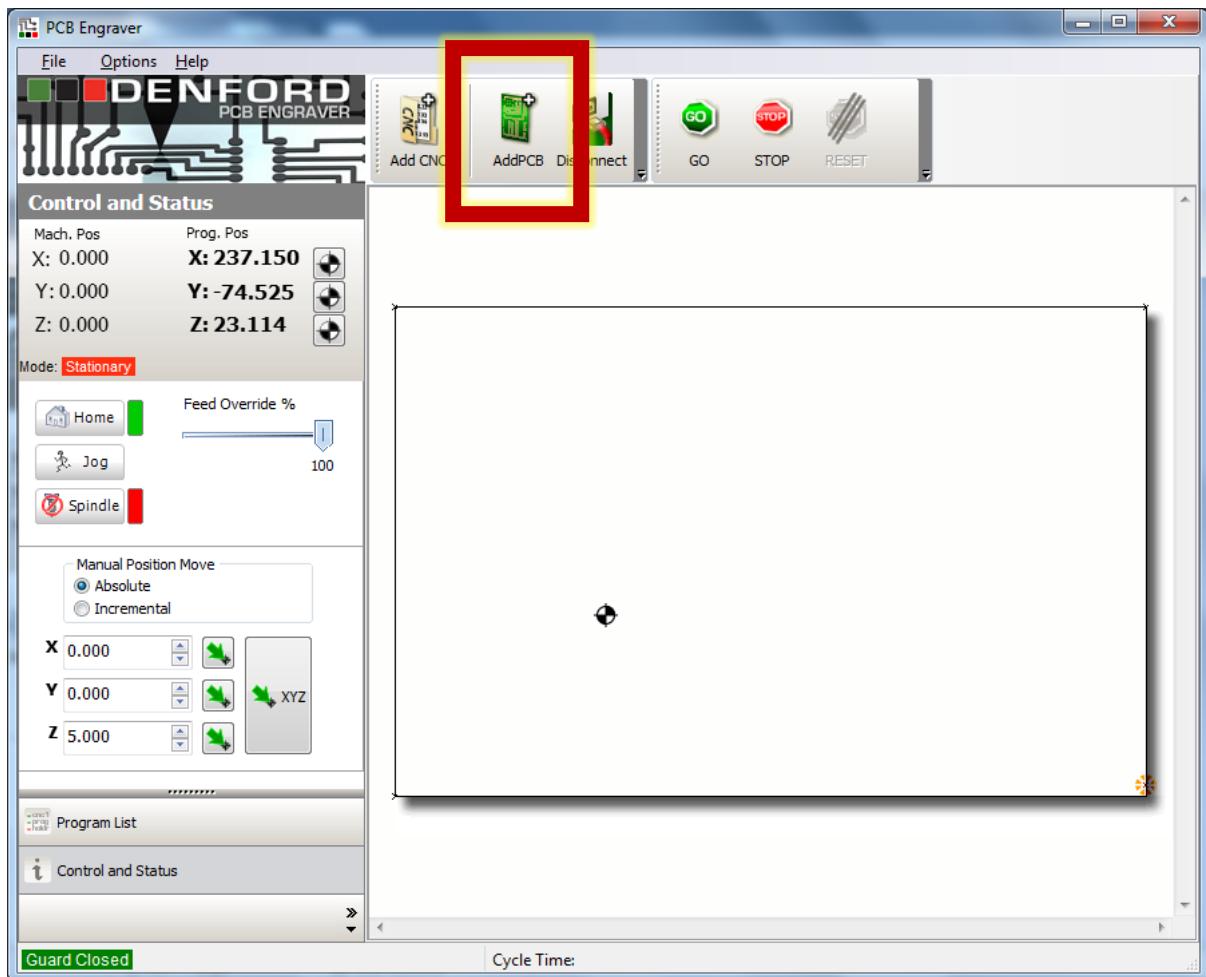


Your software will load the FLASH of the PCB engraver, and you will be presented with the default screen.

- If you opened the software before turning on/connecting the machine, you will need to hit the **Connect** button.



Make sure the machine lid is closed and click the **Home** button.

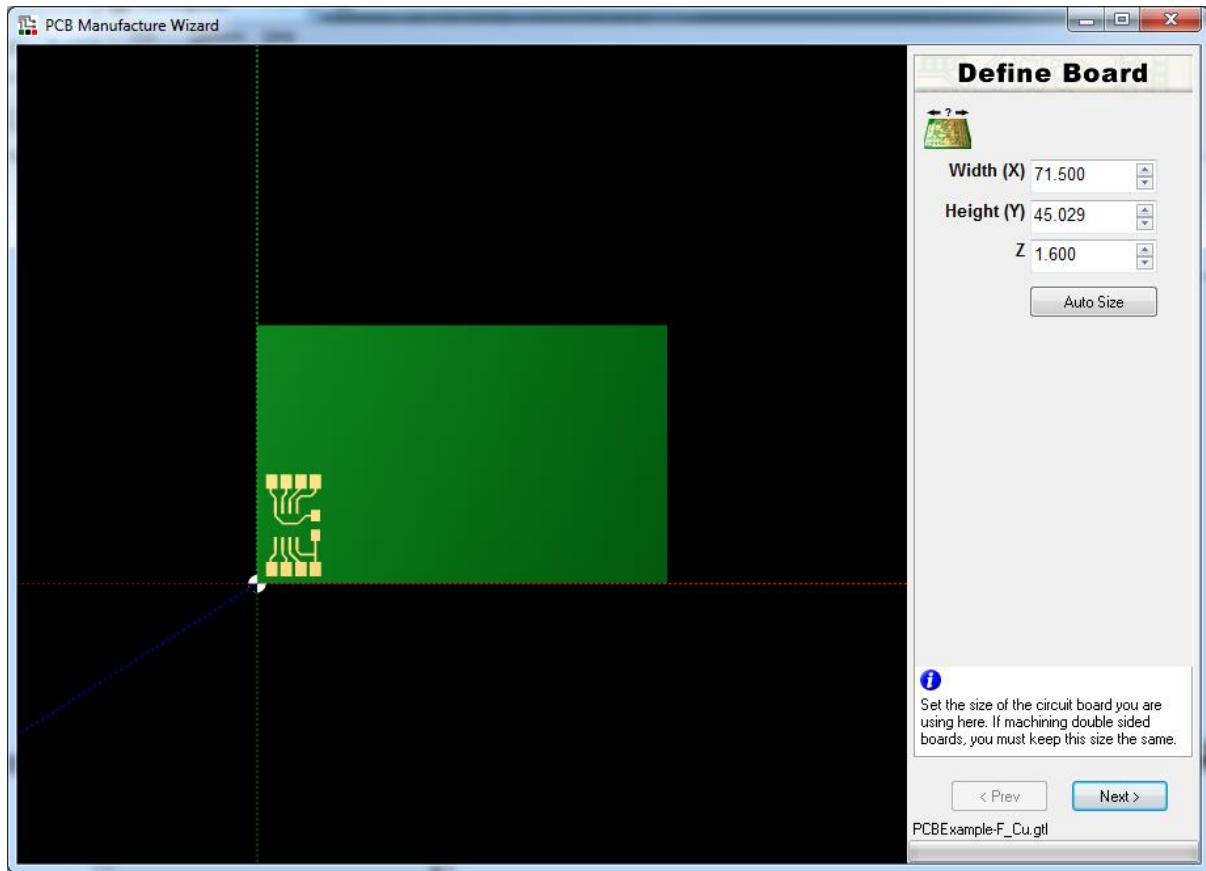


Notice that the other options are now enabled.

Click **AddPCB** button.

Select the gerber file for your top copper (**you might need to enable viewing off all file types, depending on the file extension of your gerbers**).

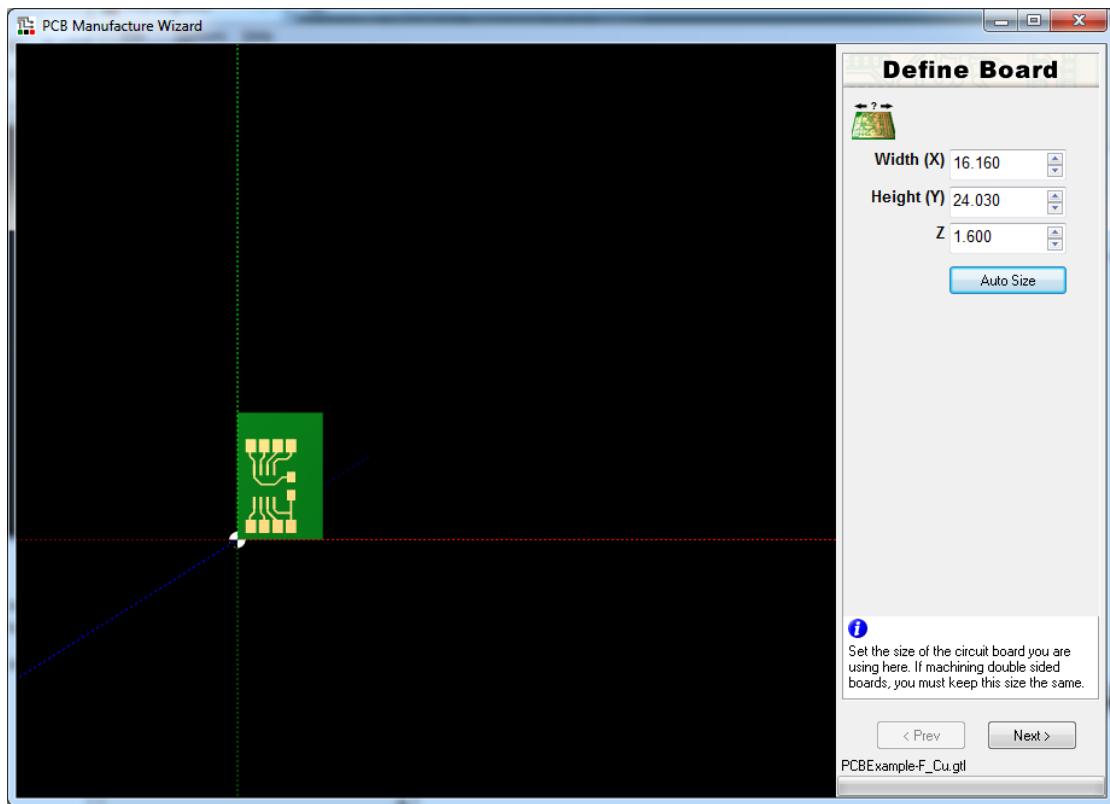
The PCB Manufacture Wizard will open.



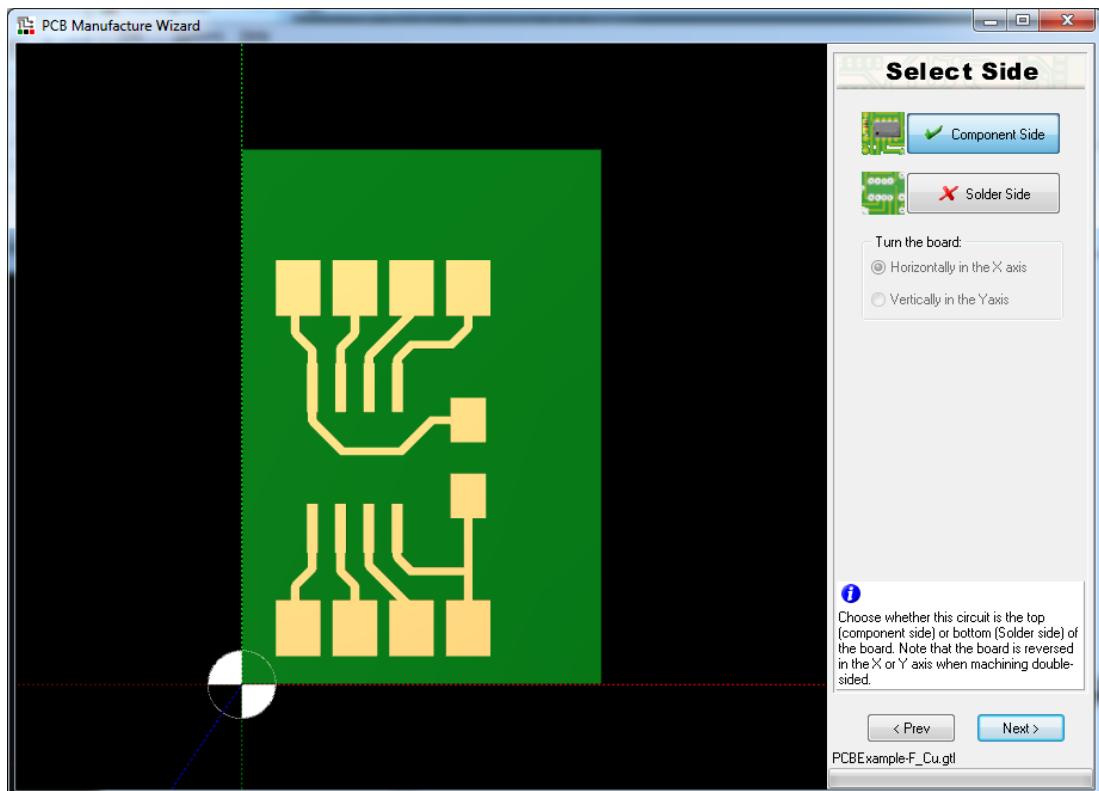
Check that your design looks OK.

- One error I have seen is that hexagonal pads exported from Eagle sometimes don't import properly (change them to be circles as a workaround if you have this problem).
- Getting ground/power planes to work can also be a bit picky. In general, it's better to NOT use a ground plane if your design doesn't need it. Any floating copper islands can then be connected to ground with a solder jump if they really need to be. You might have to play around with your gerber export and plane fill settings if you really need a real ground plane. It's also worth noting this means your circuit will take twice as long to cut (as it will cut out the traces AND the ground plane).

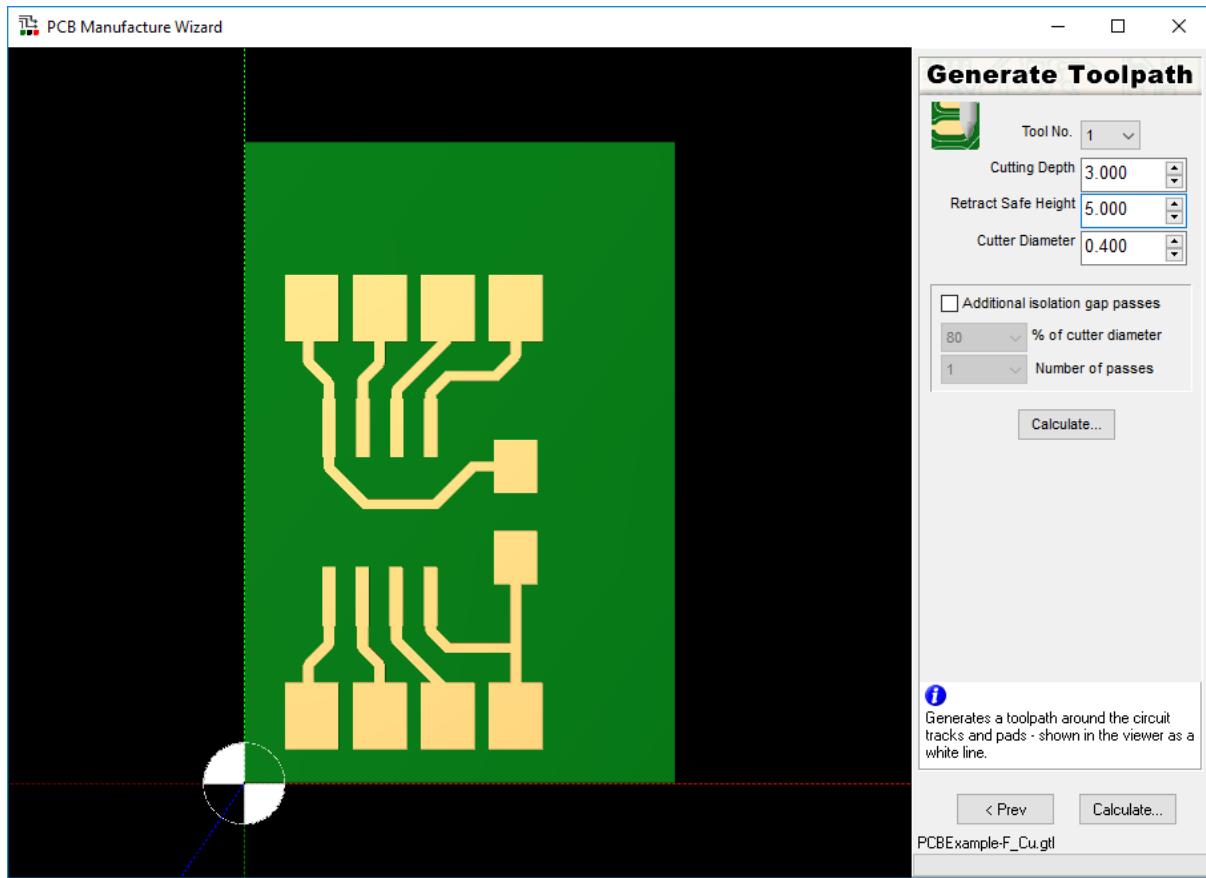
Hit **Auto Size**. This doesn't actually affect the final cut, but just makes the view tidy.



Click **Next**.



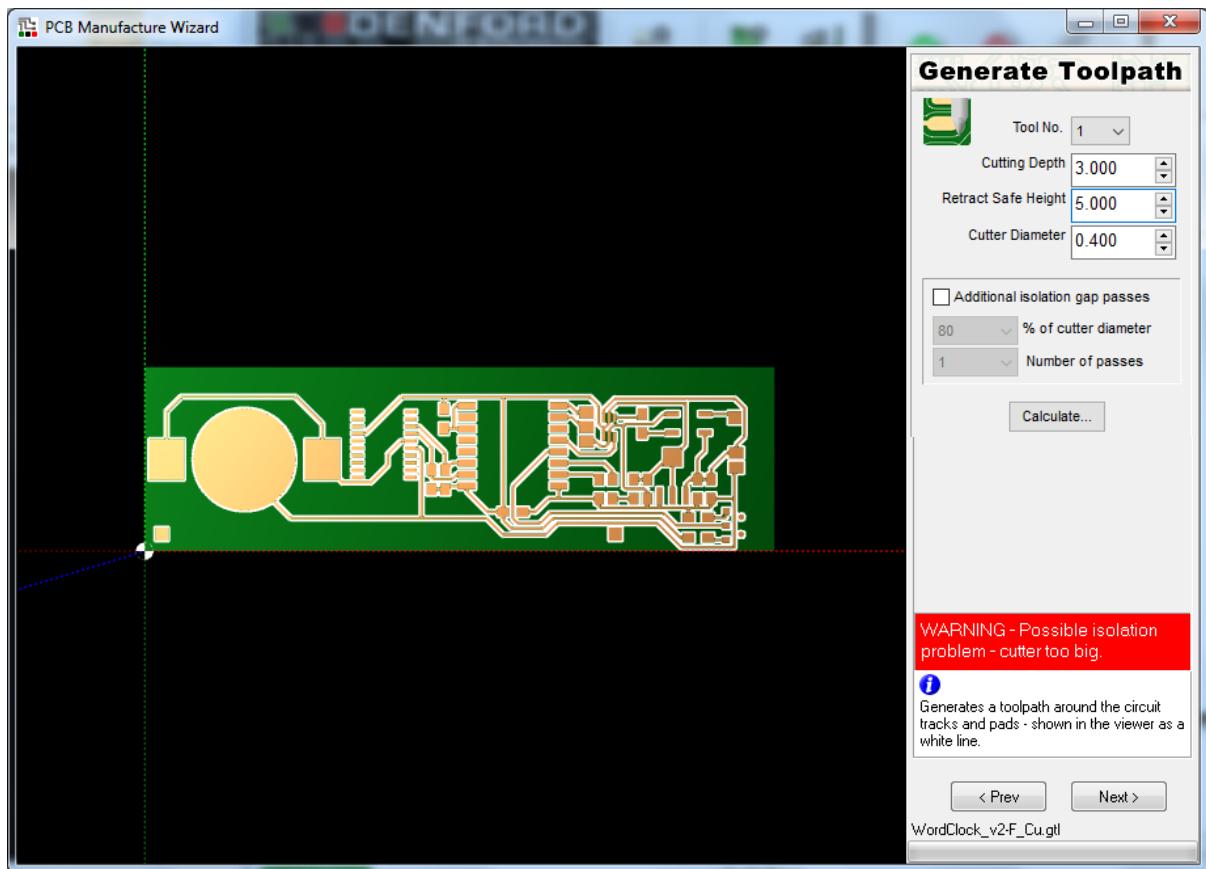
Click **Component Side** and click **Next**. If you are doing a through-hole design, it can sometimes be easier to do your design on bottom copper, and then select “solder side” here to flip the design the right way up.



In this step, we define how we will cut the traces of the PCB. The important parameters are:

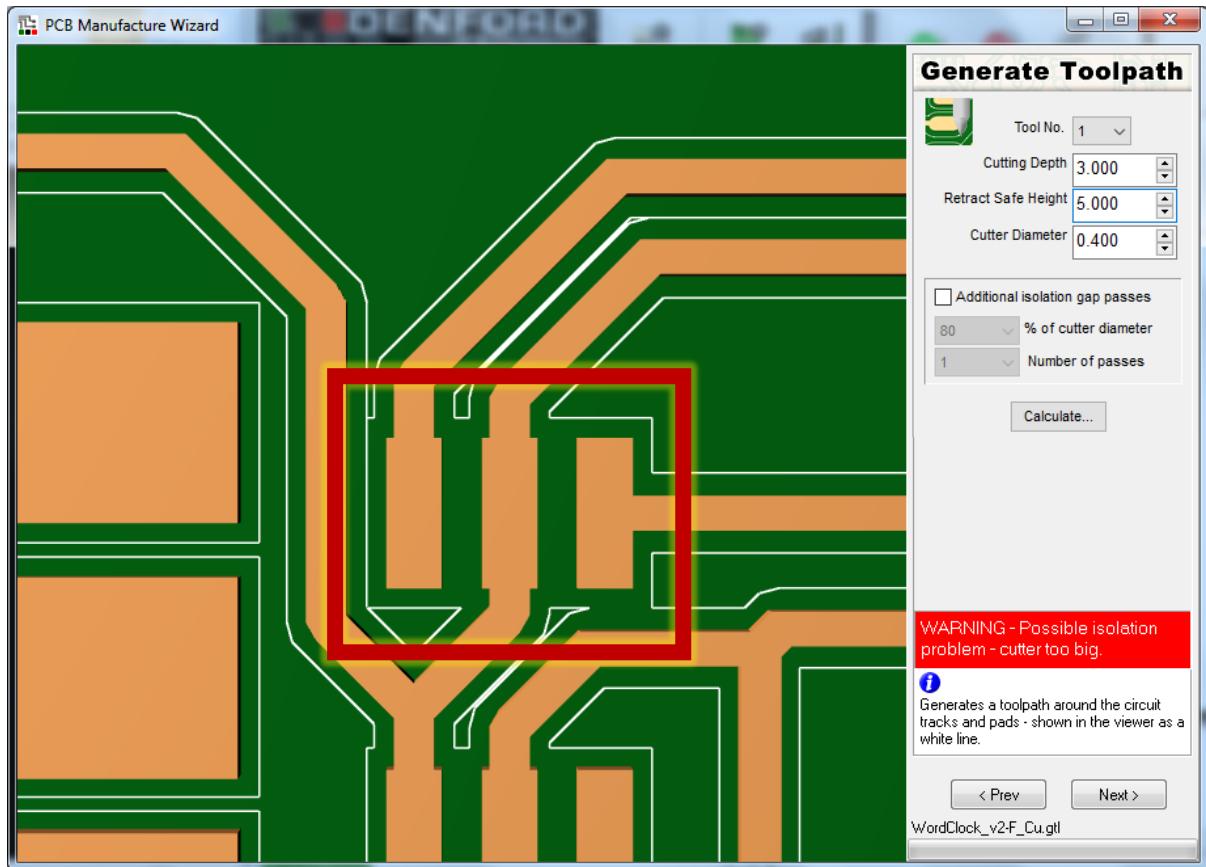
- **Cutting Depth:** This is how deep the PCB engraver will cut. We will be setting our zero level at 2 mm. By using a cutting depth of 3 mm, we will have 1 mm of “float” to compensate for an uneven PCB and copper thickness.
- **Retract Safe Height:** This is a **VERY IMPORTANT SETTING!** This is how high the drill goes when moving between cutting the PCB traces. I would highly recommend setting it to **AT LEAST 5 mm**. More on why later.
- **Cutter Diameter:** This is where we tell the software how big our endmill is. The ones we have in the lab are 0.4mm

Depending on your design, with the **Cutter Diameter** set to exactly 0.4mm you might get the following warning when you click **Calculate**.



This basically means that there were parts of your design that the software doesn't think it can fit a 0.4mm cutter through. This happens a lot, even if your PCB passed a design rule check for 0.4mm clearance.

Very unhelpfully, the software is rubbish and doesn't show you *where* the error has happened. Zoom in and scroll around and look for the problem points. In this more complex PCB design, it's around a very small footprint shown below.



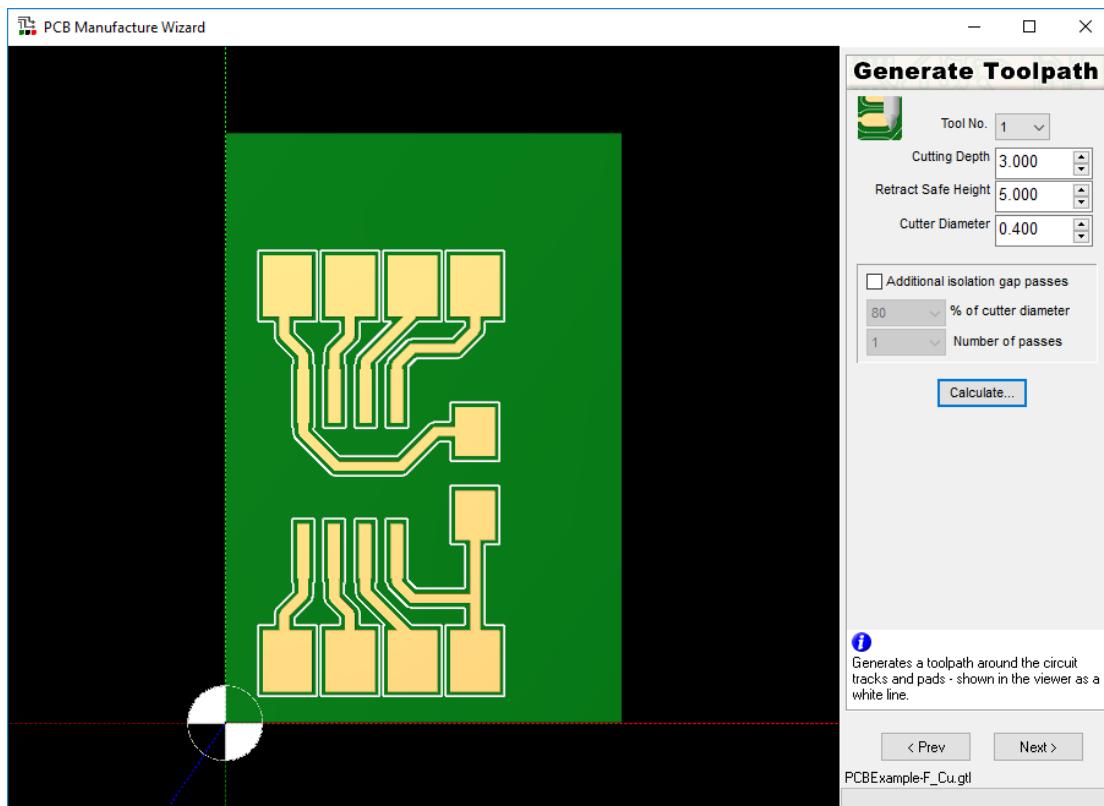
Notice that the white line (the cutter path) doesn't go between all of the pads. If we were to cut this design, those pads would still be connected together.

There are two ways to solve this problem:

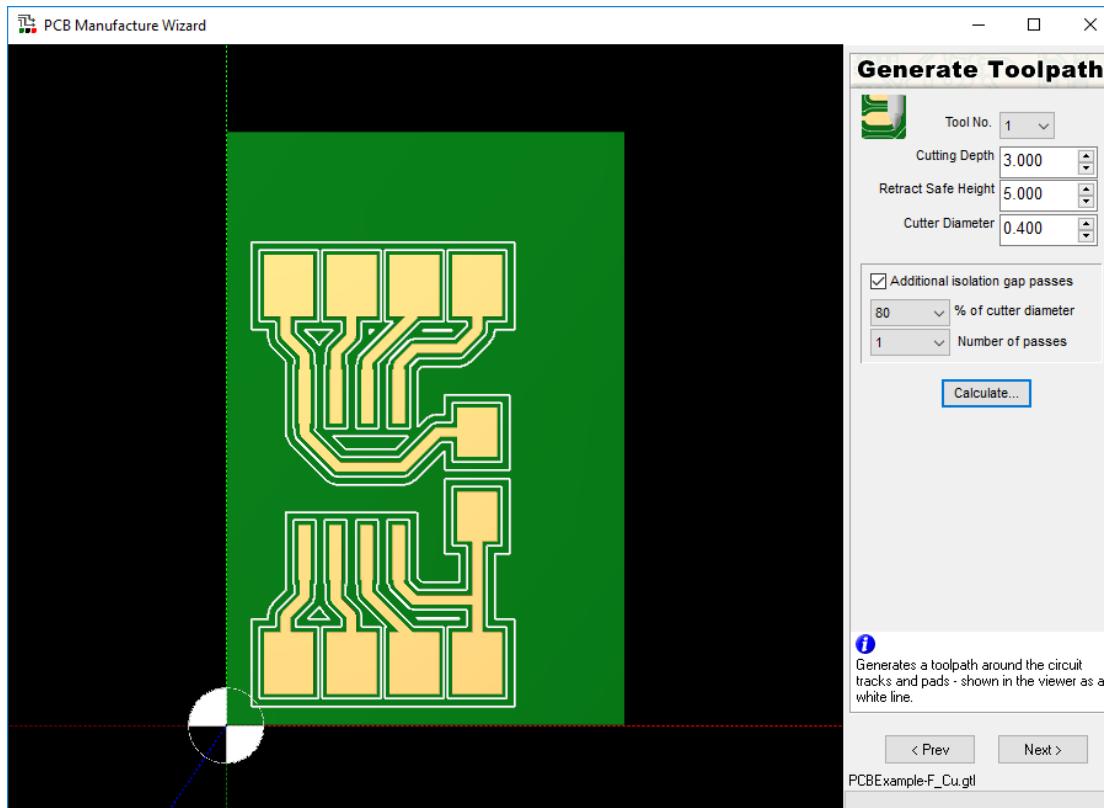
- 1. Edit your design to have more clearance between pads/traces.**
- 2. Tell the software your endmill is smaller than it actually is.**
  - Usually setting the endmill size to 0.39mm will do.

In the case of the PCB shown, setting the endmill size to 0.395mm was enough for the path to be generated without issue. Just be sure to not go too small! For example, telling the software your endmill is 0.2mm will result in you getting super thin traces. I wouldn't suggest going lower than about 0.35mm – after that, fix your circuit.

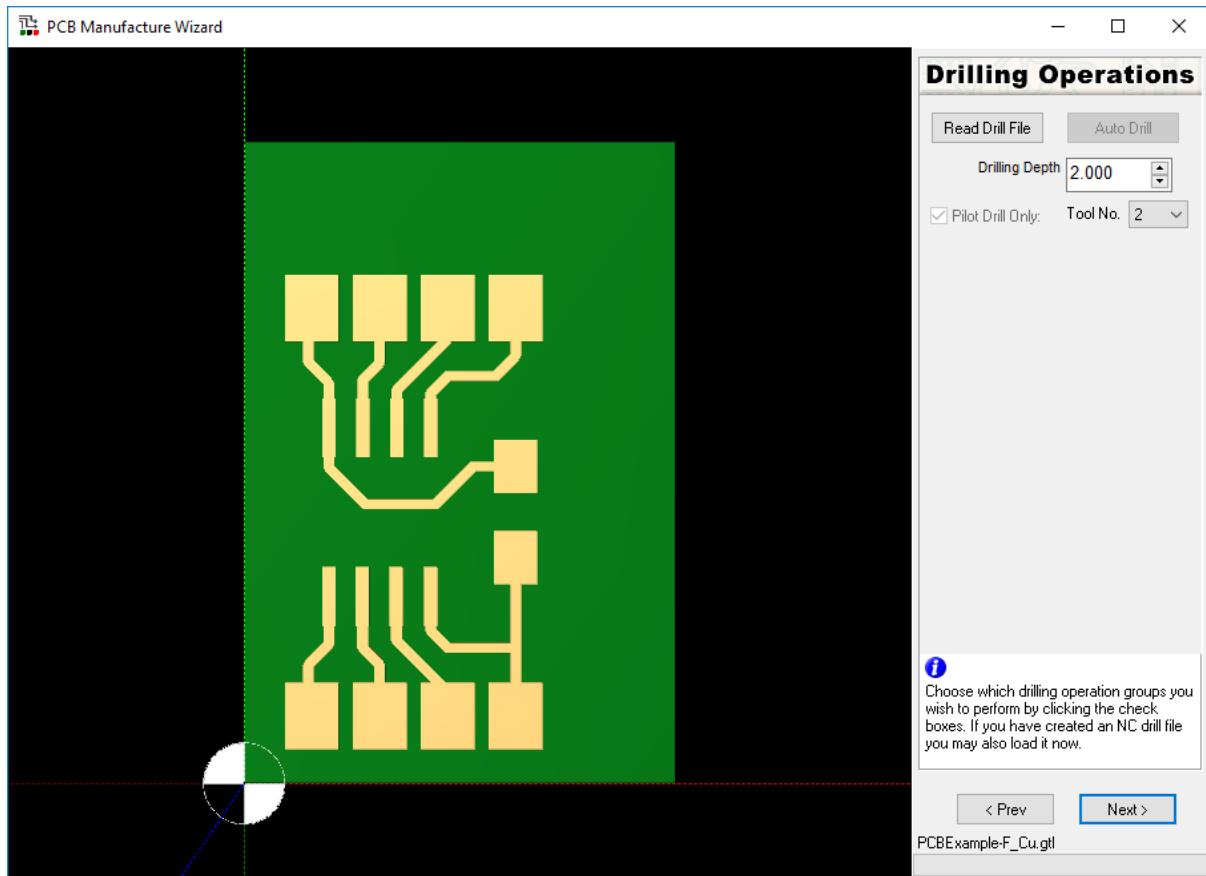
For the basic PCB included with this tutorial, the cut path will generate without any warnings. The cut path is displayed as a white line around the PCB traces.



The **Additional isolation gap passes** option allows you to remove more copper so that you have a larger clearance. It does this by doing an additional cut around the traces (notice the double white line in the image below). In general, I don't think it's necessary. All it does is double the time it takes to cut your circuit.



For this tutorial, disable additional isolation gap passes and click **Next**.



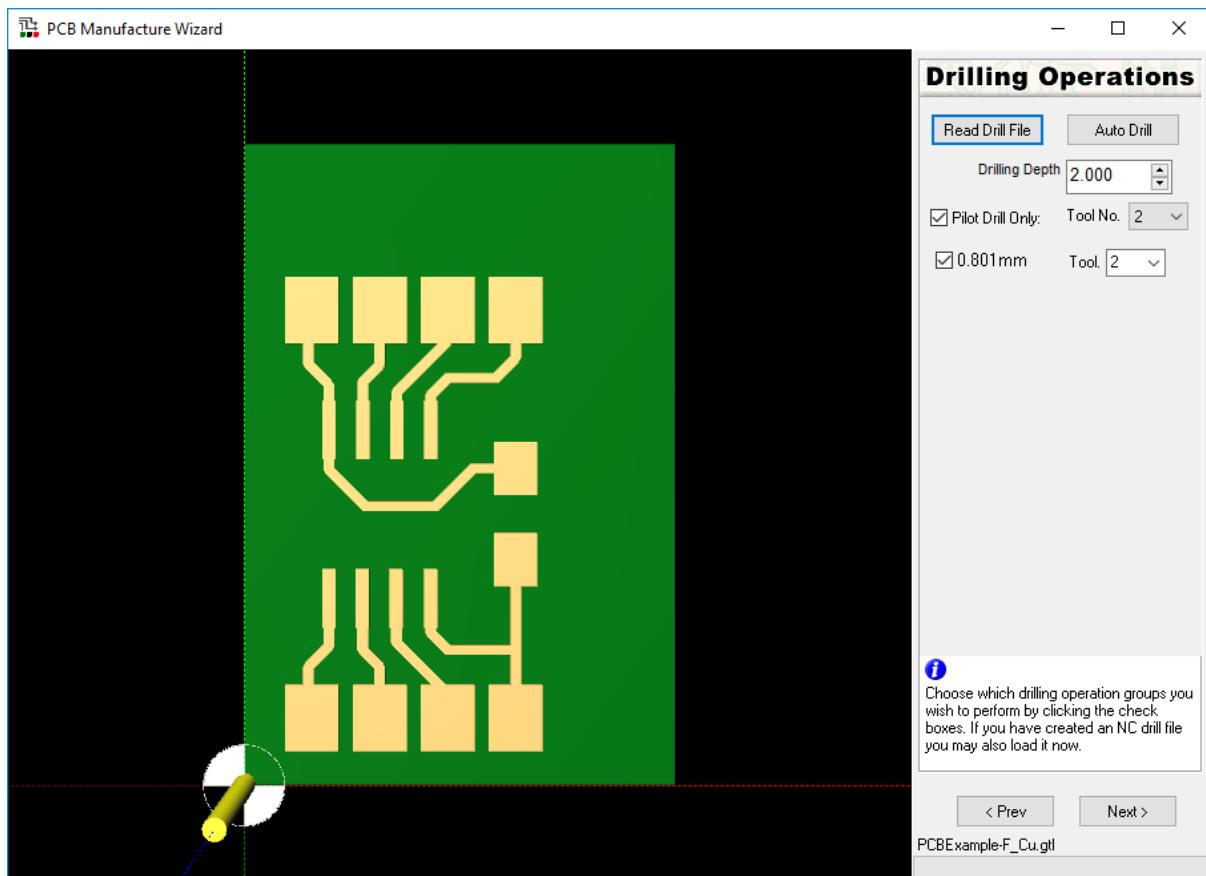
This screen allows you to define your drilling operation.

Getting the Denford software to accept your drill file can be a bit of a black art. An EXCELLON drill file generated from Eagle using the SparkFun CAM file tends to import OK (although the drill sizes will be meaningless). From KiCAD, I've found that you need to hand edit the drill file to add three zeros to the end of each coordinate. Compare the drill files *PCBExample\_original.drl* and *PCBExample\_fixed.drl* for an example of the edits you might need to make.

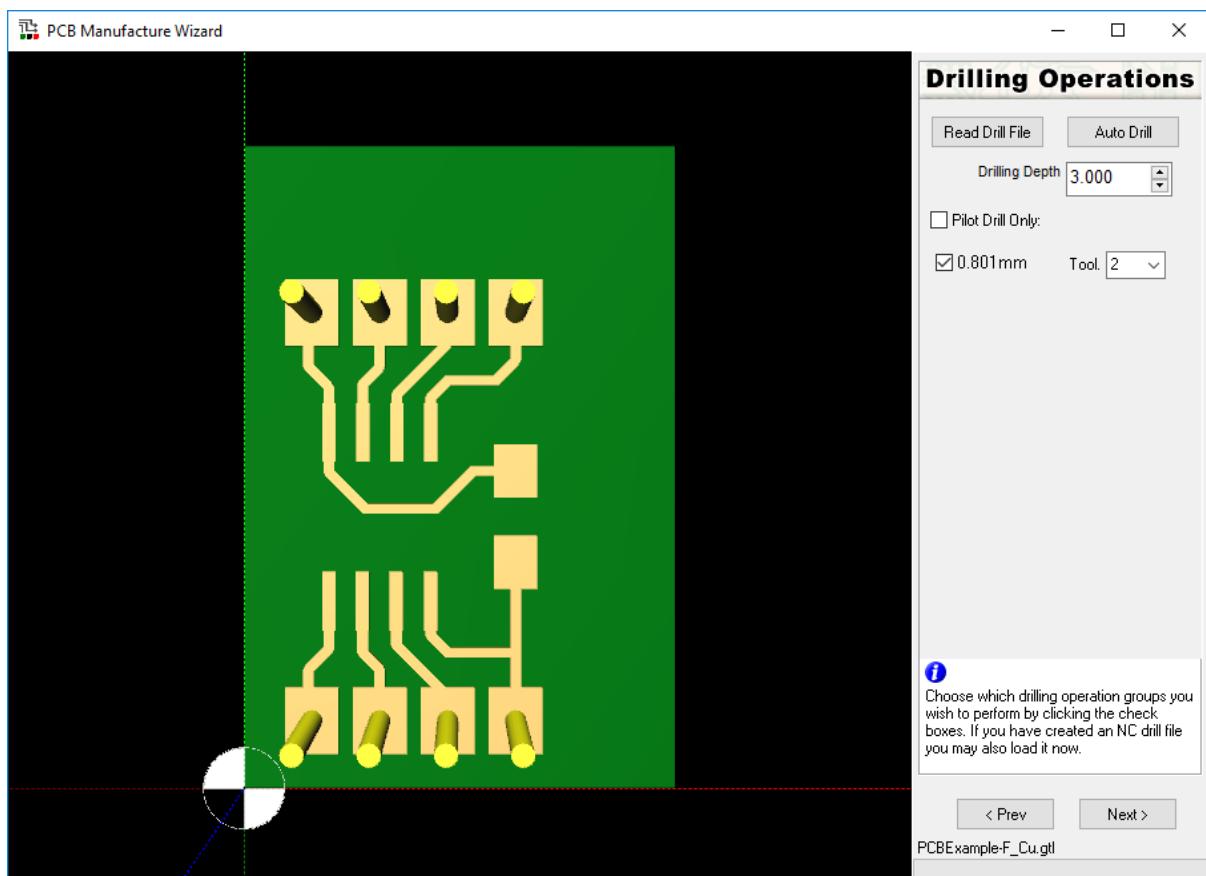
Don't bother with the **Auto Drill** option – it doesn't do anything...

Click **Read Drill File**, and choose the drill file you generated. Notice that the drill file needs a .drl extension. If you use Eagle, the default is for a .txt extension. Just chance it to .drl and the engraver software should be able to see it.

If you import a drill file and you only get yellow markers around the origin like in the image below, you know you have a problem with the file format.



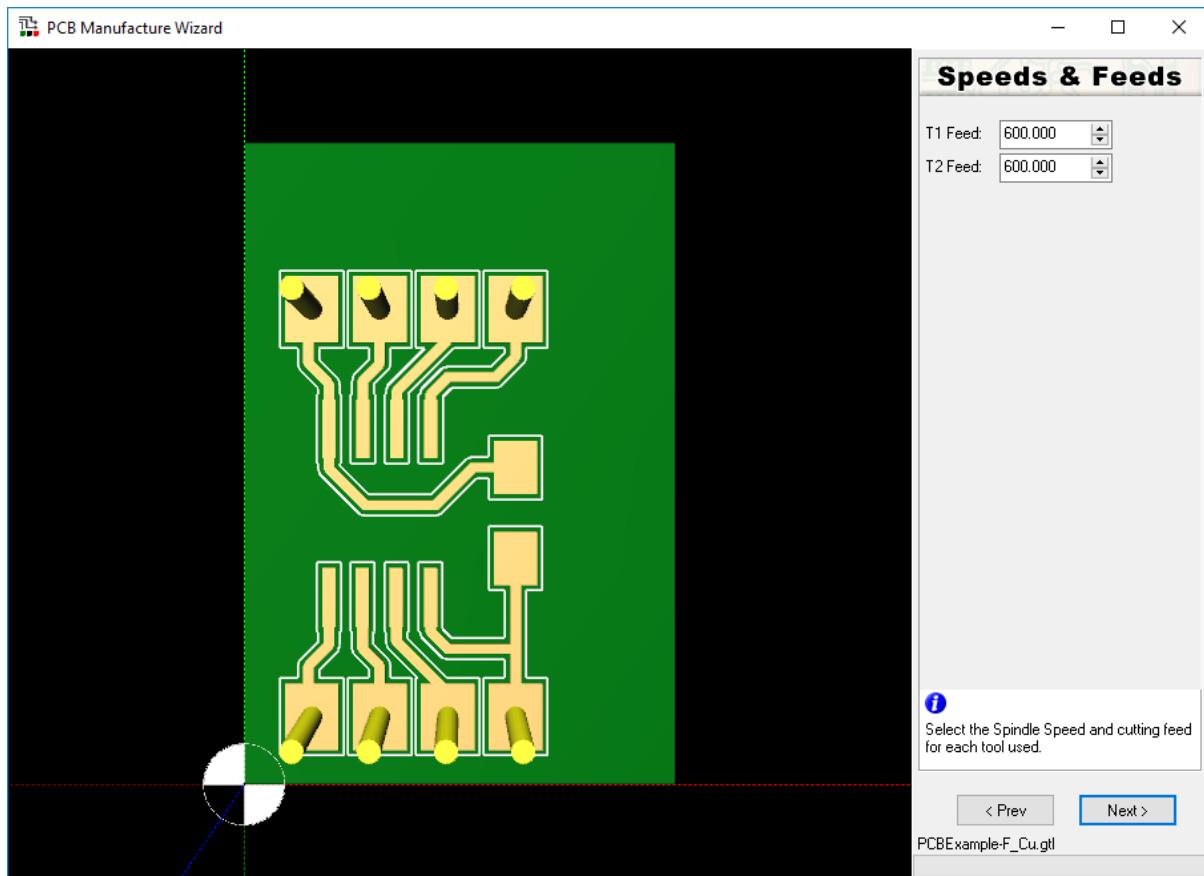
Assuming your file imports OK, you should see yellow markers showing your drill hits.



The import settings are:

- **Drilling Depth:** This is a **VERY IMPORTANT SETTING!** Set this to 3 mm. We will be using the floating head on the PCB engraver to control the drill depth to be 2 m.
- **Pilot Drill Only:** If you have this box checked, then all your drill hits will be done using the same drill bit. If you uncheck this box, the machine will ask you to change the drill bit for every different hole size. Notice that the drill size never seems to import properly: in the image above, the holes were 1 mm. Instead, make a note of which tool number matches to which holes and make sure you use the right drill bit later.

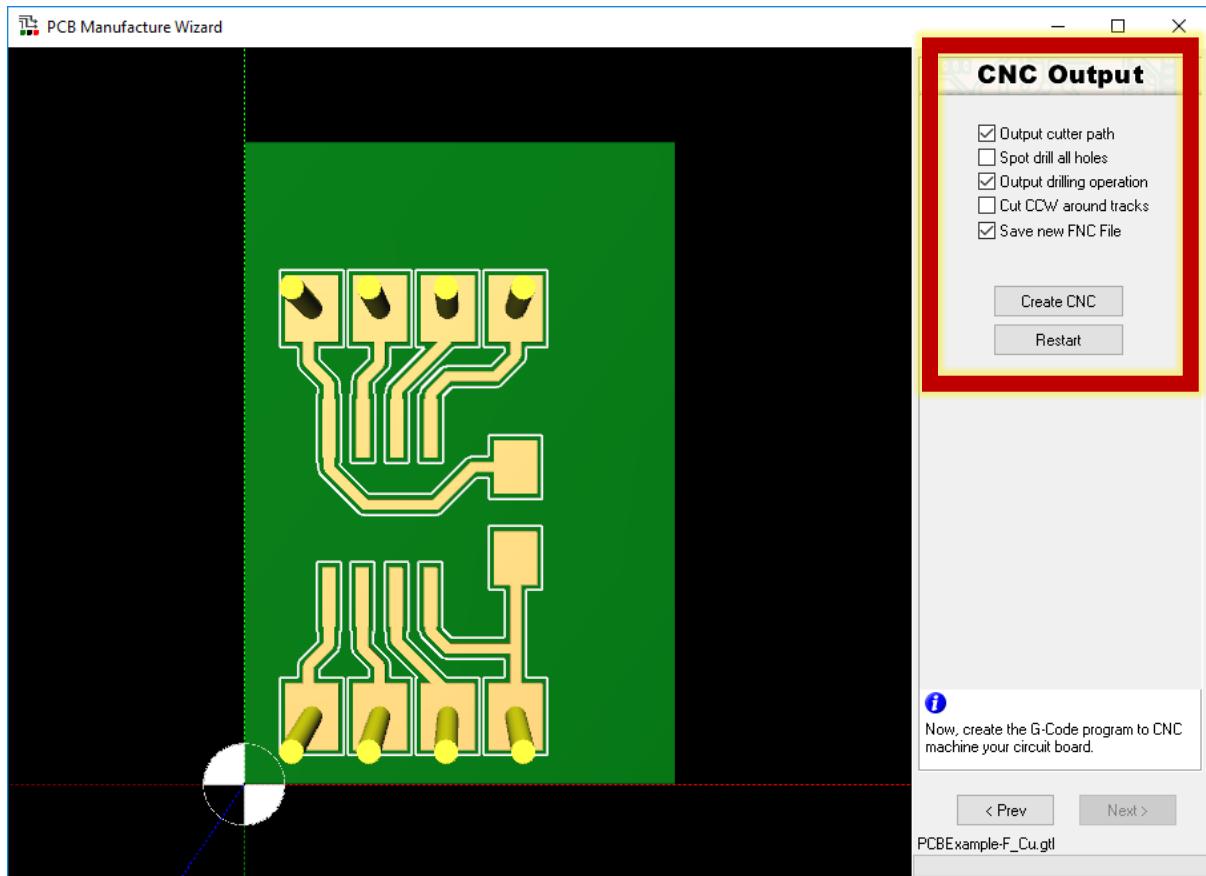
Click **Next**.



### THIS IS THE MOST MISLEADING AND USELESS PART OF THE SOFTWARE.

The settings on this page actually have **NO** relation to how fast the machine will actually travel! We broke a lot of bits before we figured that out...

So, ignore this page, and just click **Next**.

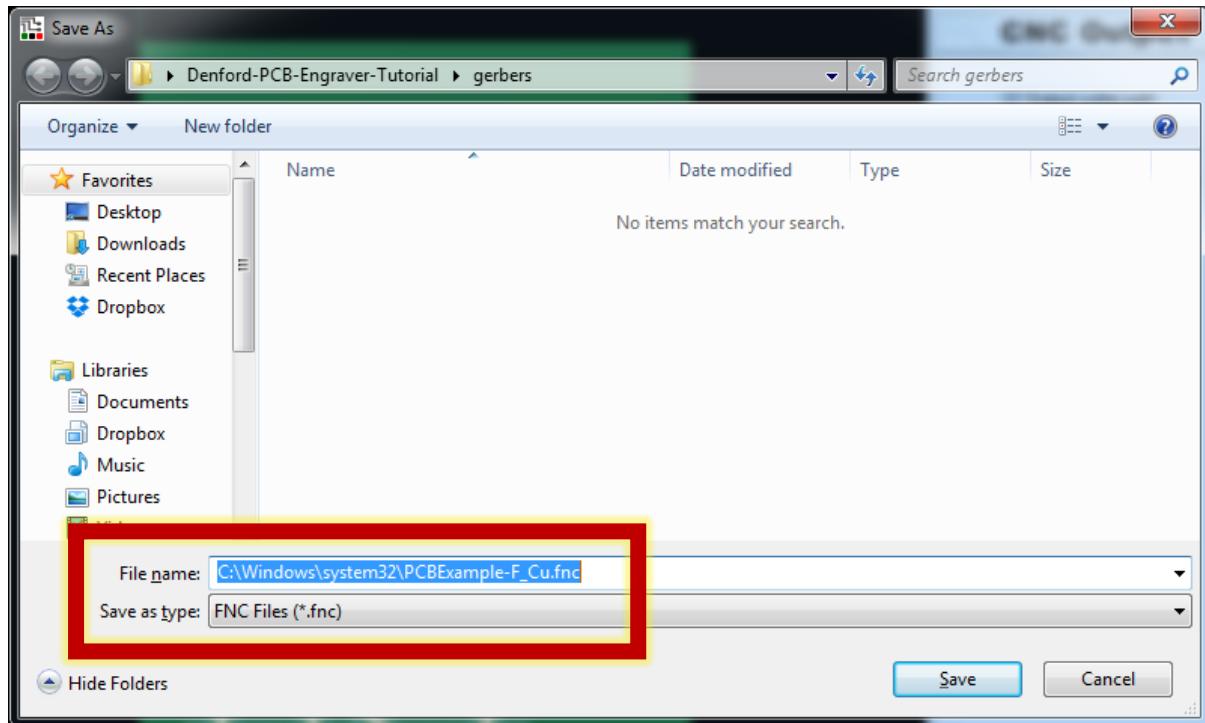


This is the final step in the wizard. Take a final look at your circuit – this is what will get cut/drilled.

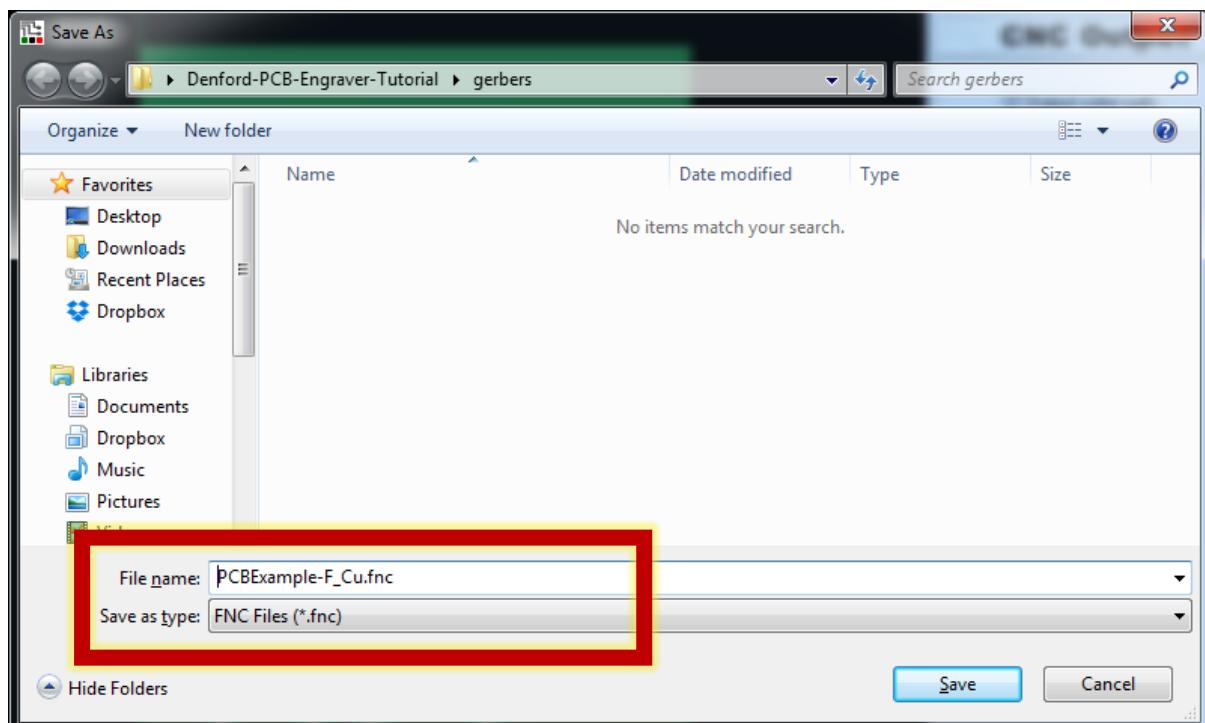
The settings are:

- **Output cutter path:** This should be **CHECKED**. This is what actually outputs the trace cutting path.
- **Spot drill all holes:** This should be **UNCHECKED**. If you leave this checked, the final step of cutting the traces with the 0.4 mm bit will be to tap the centre of each drill hole. This is useful if you plan on drilling the holes by hand, but we're automating that step. Spot drilling just wastes time.
- **Output drilling operation:** This should be **CHECKED**. If it's not, then the machine won't drill any of the holes we imported.
- **Cut CCW around tracks:** This should be **UNCHECKED**.
- **Save new FNC File:** This should be **CHECKED**.

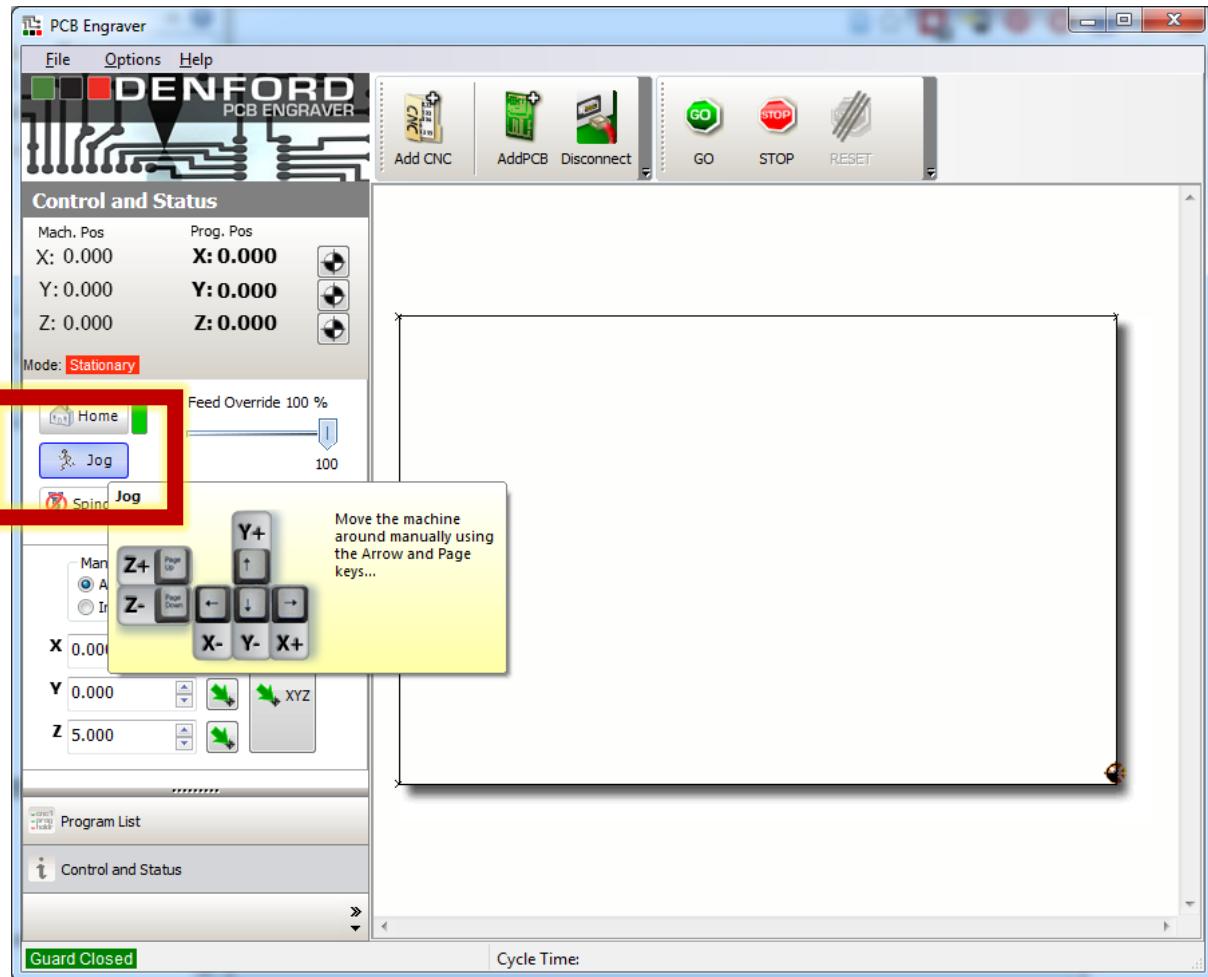
Click **Create CNC**. The save file dialog will appear. **BE CAREFUL!** Even though it **LOOKS** like it will save to the folder you opened your gerber file from, the “File Name” box will actually specify a completely different path: usually either the desktop or the Windows System folder...



So, **MAKE SURE YOU CHANGE THE “File Name” TO BE JUST THE FILE NAME YOU WANT!**



Your PCB will automatically be imported (although you might not be able to see it yet depending on where the current origin is set). The next step is to place our FR1 in the engraver. To make things easy, we'll reposition the bed to the front of the machine using the **Jog** option.



Click on **Jog** and use the arrow keys to bring the workbed to the front of the machine.

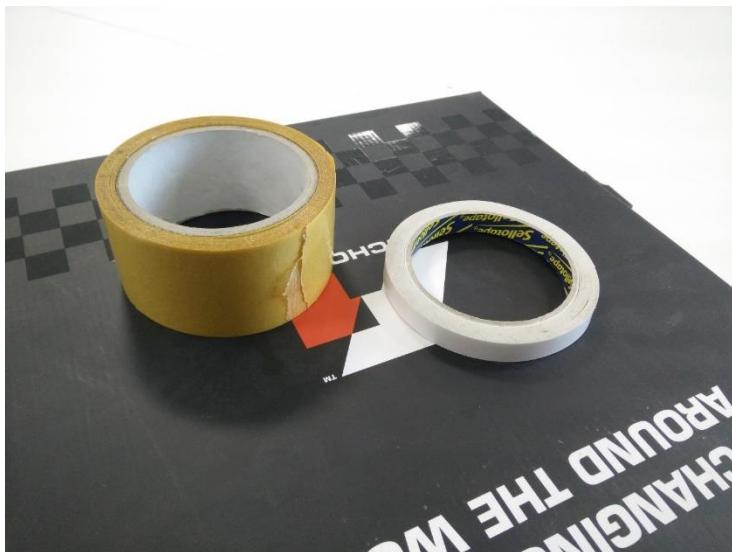


Notice that we have a sacrificial piece of FR1 already on the machine bed. This is to stop the actual acrylic bed from getting damaged.

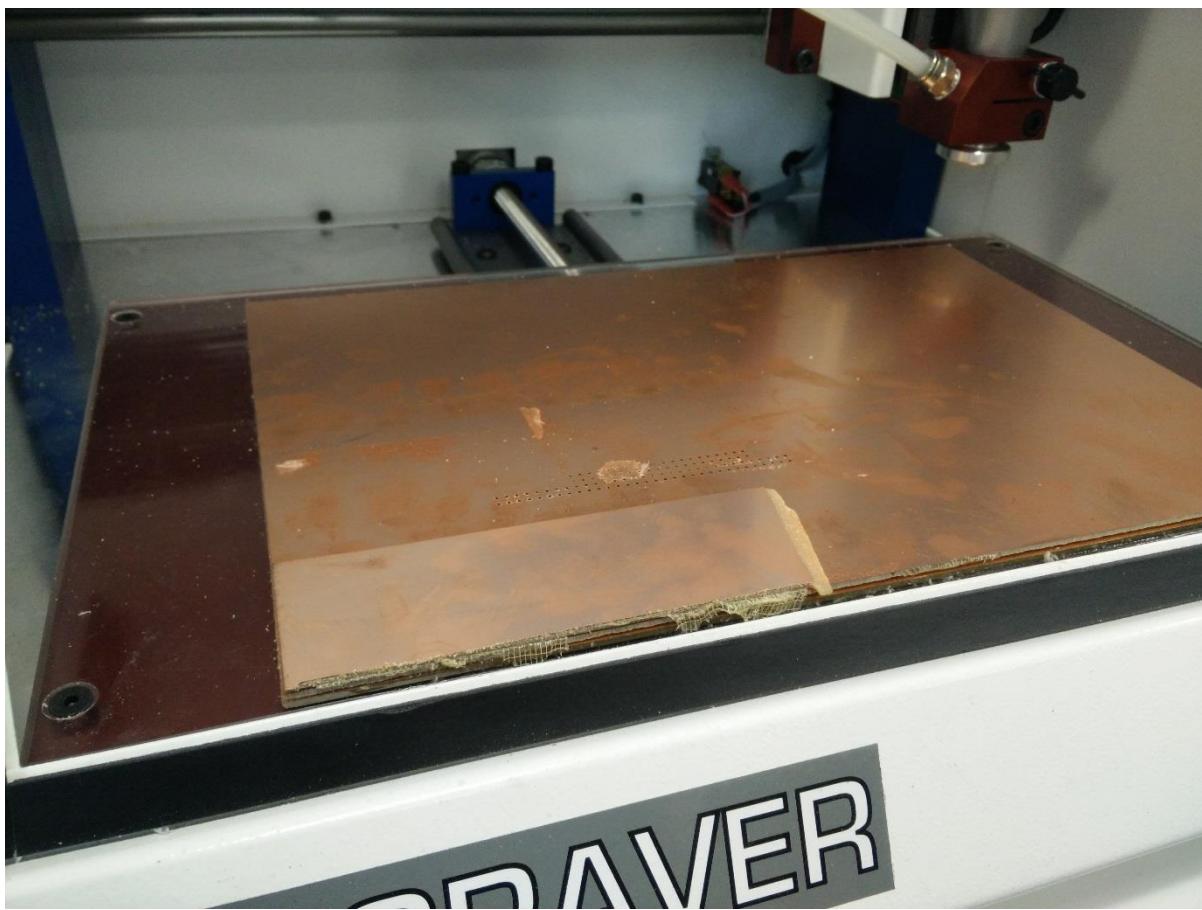
When it comes to choosing a piece of FR1 there are two options:

1. Pick a piece a bit bigger than your actual circuit –Lindsey can cut it to roughly the right size for you using the bandsaw. When you’re finished, just pry it off and either use it as is or ask Lindsey to trim it down for you. This is the easiest option.
2. Use a piece larger than your circuit, and then cut the circuit out on the machine using a large diameter endmill. This option is harder (it involves hand-editing some G-Code) but gives you more flexibility. For example, you can cut more complex shapes or cut multiple circuits from the same piece of FR1.

In either case, stick your piece of FR1 onto the sacrificial piece using some double sided tape. If your circuit is quite small, you can get away with using the weaker double sided tape (which is good, because it’s easier to take off and to clean!). If you have quite a large piece, use the stronger carpet tape usually used for the CNC machine. The downside is that you’ll need to use some white spirit/acetone to clean it, and probably use the heat gun to get your PCB off the sacrificial layer.



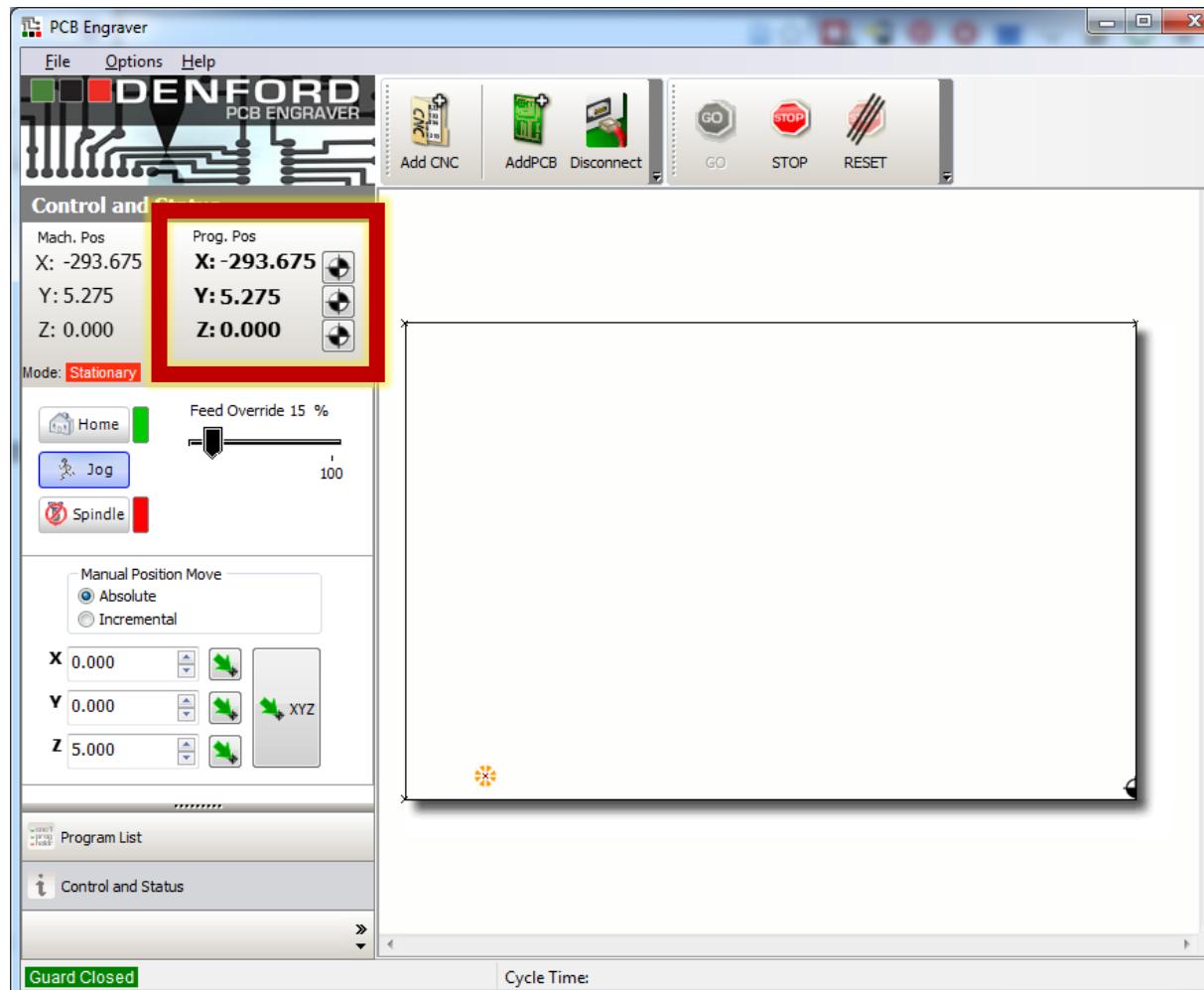
Try and make sure that the edges of your FR1 are roughly parallel with the edges of the workbed. You should end up with a setup similar to the one shown below.



Now to set the origin, which is the bottom left corner of our PCB. Using the jog setting, move the cutter to roughly the bottom left hand corner of where you want your PCB to be on the FR1.

1. **CHANGE TOOL IF NECESSARY** – You should have a drill with the drill bit in, set at 2 mm (i.e. NOT have the 0.4 mm endmill drill in!).
2. **Check motor ON and at 20k rpm**

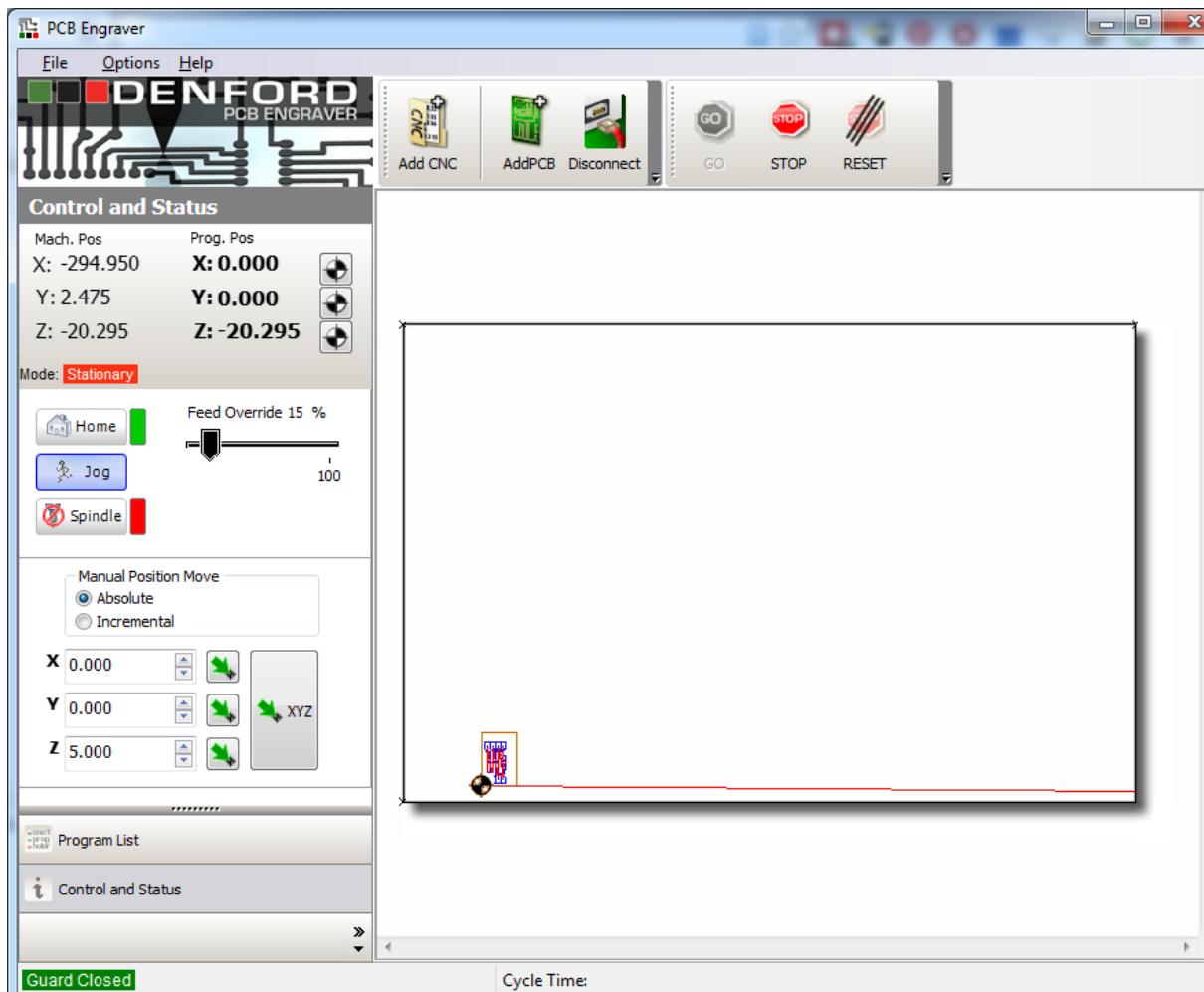
3. **VERY IMPORTANT!** Now, reduce the **Feed Override** setting to **20%**. This will slow the movement speed of the cutter. You can now fine tune the XY origin point of your PCB. You can use the **Page Down** and **Page Up** buttons to move the machine head up and down. **BE CAREFUL NOT TO PLUNGE THE ENDMILL INTO THE PCB!** You'll break the endmill if you do.



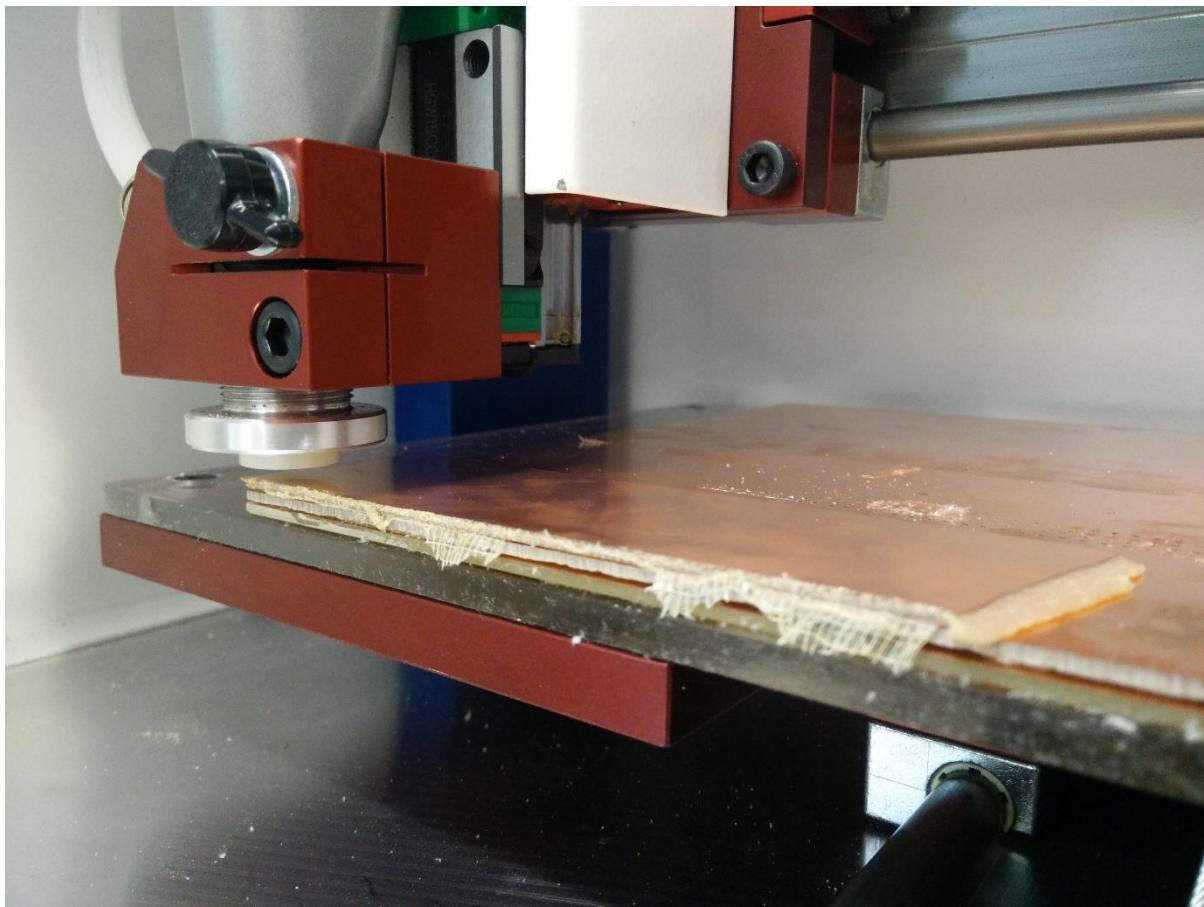
When you are happy with the XY position, set it as the XY origin by clicking the target symbol next to the X and Y position indicators.

TODO – Take pictures of how to change the drillbit. **Make explicit that the trace endmill should NEVER be removed!**

TODO – Add advanced section on how to set cut depth on trace endmill.



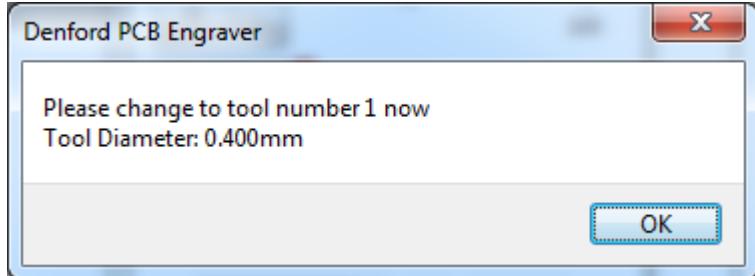
Notice that our PCB has now come into view on the virtual workbed, and that the XY program position of the machine now shows (0, 0). The machine head should now be sitting slightly above your PCB origin point.



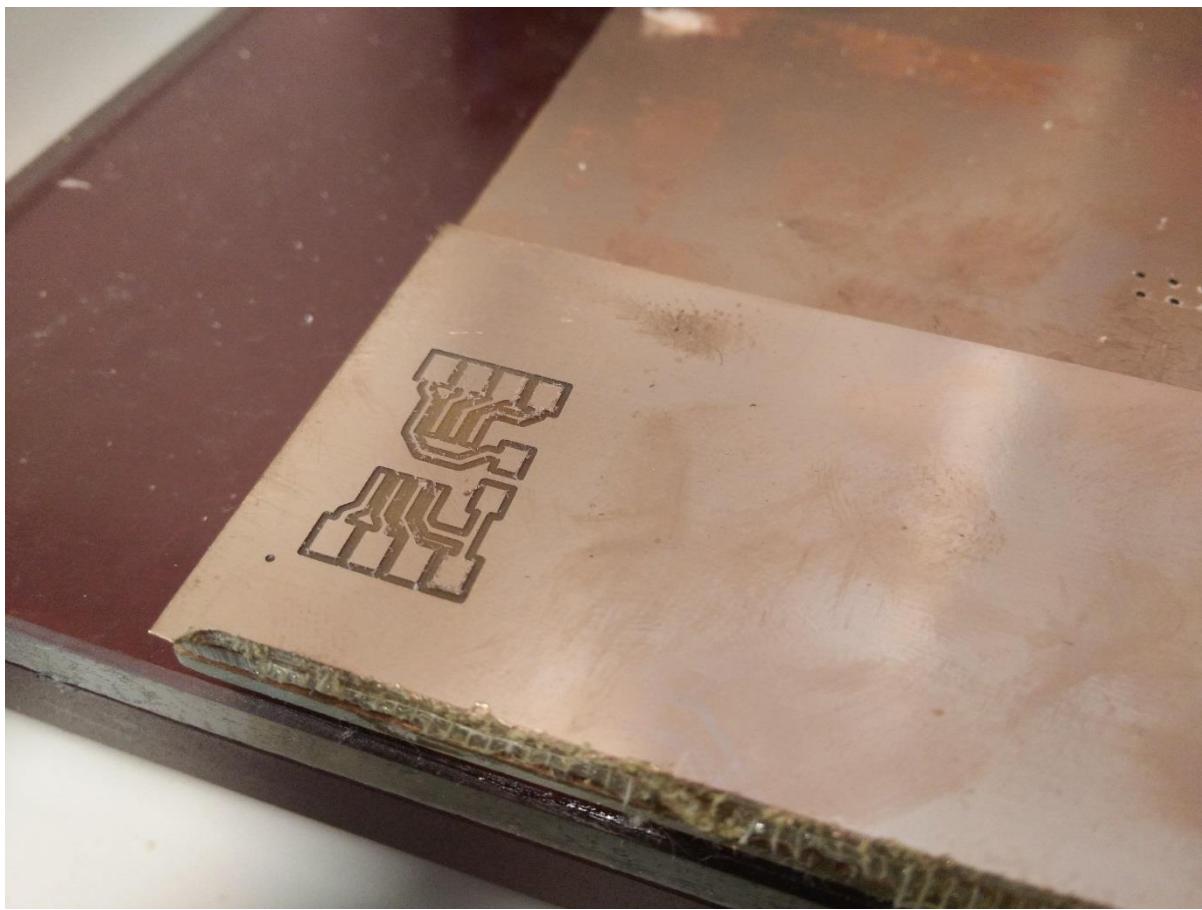
Now we need to set the Z height. Start by clicking the **Spindle** button to turn on the spindle. **MAKE SURE YOU TURN ON THE SPINDLE!** If you don't, you'll break the bit! Also turn on the vacuum – it's the big red button on the handle. Now using the jog setting, lower the endmill into the FR1 until the drill bit just bites into the copper and no more. Now, set this as the Z-axis home position by clicking the target next to the Z Prog. Pos, just like we did for the XY.

At this point you might want to retract the endmill out of the copper and check that it left a small drill in the copper, like the one shown below.

We're finally ready to cut the traces of the PCB! Hit the **RESET** button. **CHECK THAT THE FEEDRATE IS SET TO 20%!** When you're ready, turn on the vacuum and press the **GO** button. Depending on what tool was used last, the software might ask you to change to tool. Make sure you have the drill with the 0.4 mm endmill in.

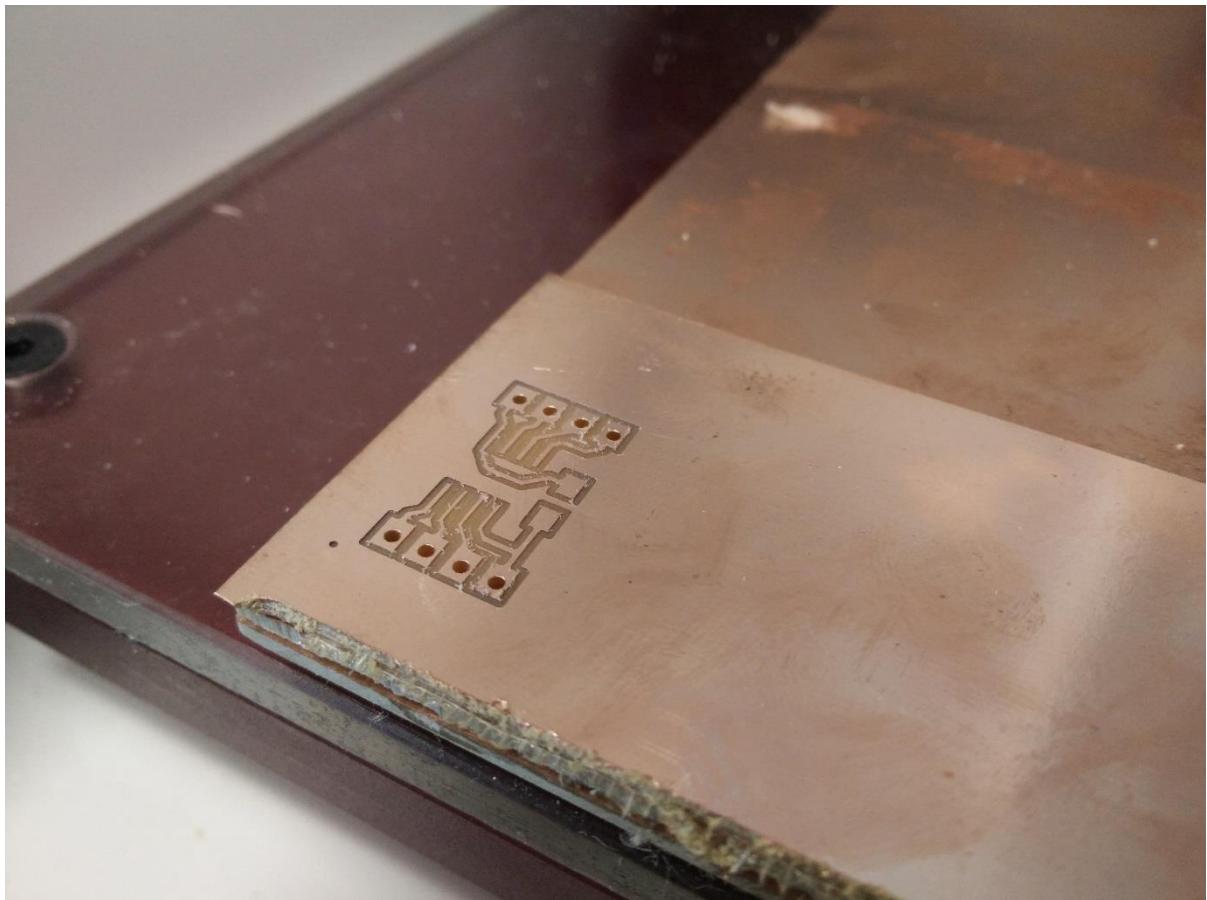


When the traces are finished cutting (it will take about three minutes for this simple circuit), the machine will ask you to change the tool to drill the holes. If everything is going to plan, your traces should look something like this



We can get the 2<sup>nd</sup> motor setup while the traces are being cut. Switch the motors over, click OK. For the first few holes, **keep your hand over the emergency stop button!** If the drill doesn't completely clear the copper between the first two holes be ready to hit stop! This is why we set the **Safe Retract Height** to 5 mm.

When the drilling is finished, turn off the vacuum and you should have something that looks like the image below. If you are taking the easy option, you're now finished with the Denford. Carefully remove the PCB. You might want to use one of the spatulas. Just take your time, and be gentle!

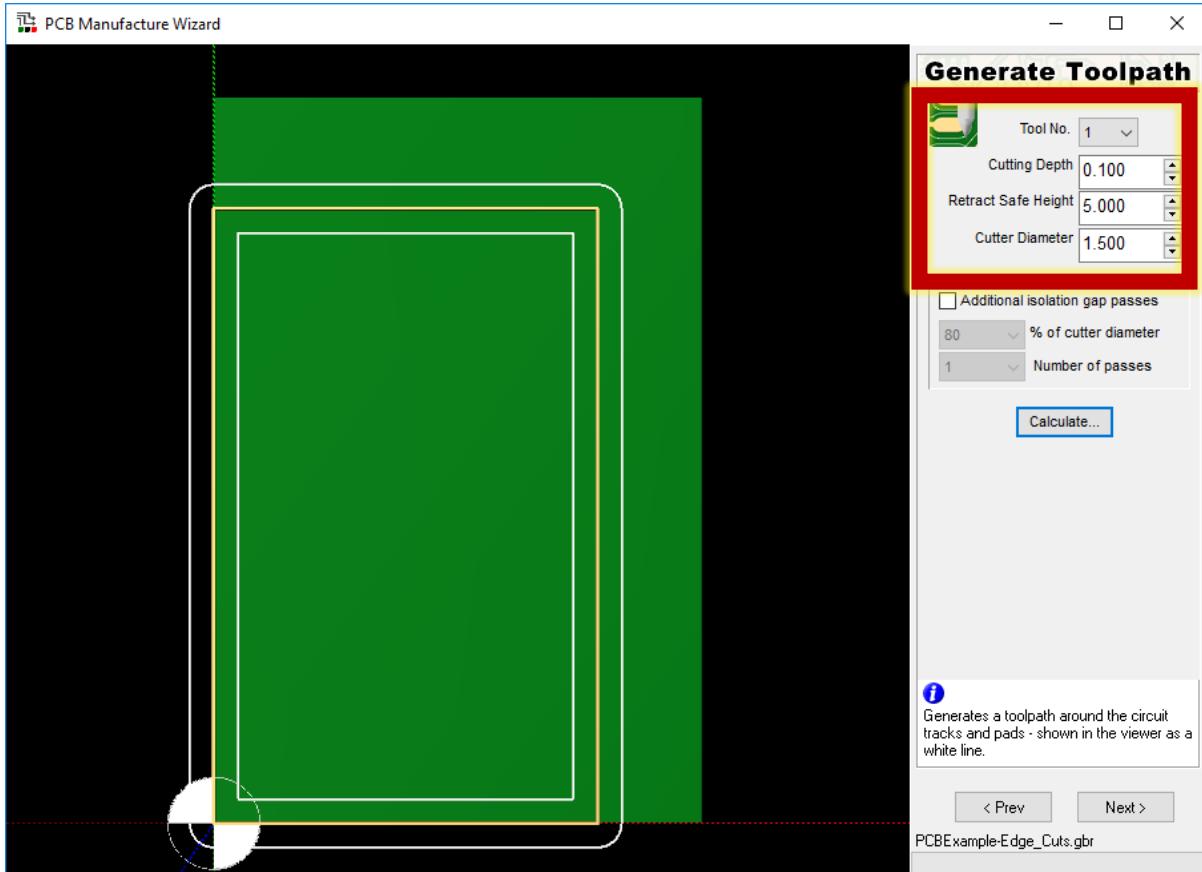


Hopefully, you now have a lovely PCb ready to be cut out, cleaned up and soldered.

## Cutting Out A Board

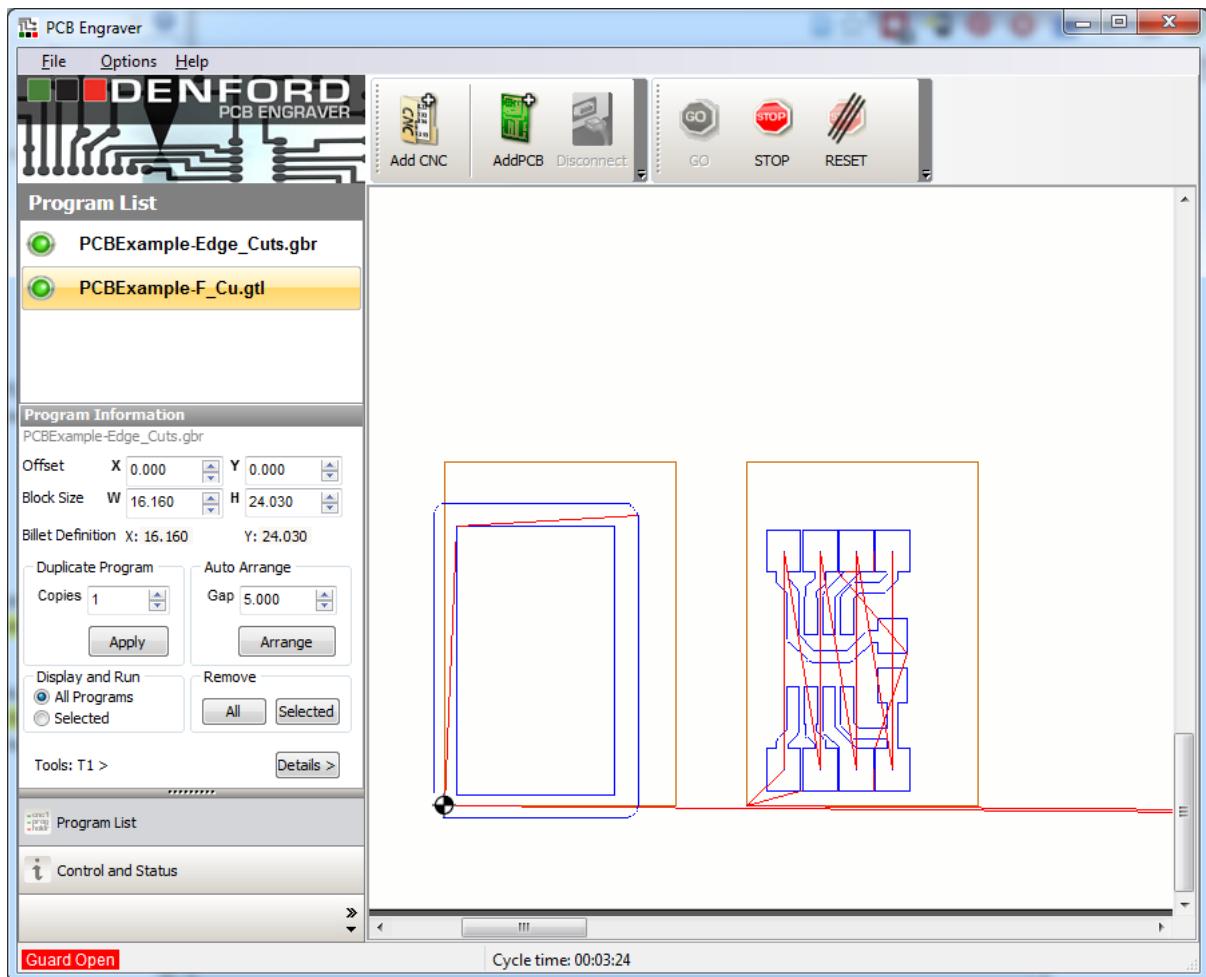
Skip ahead to the “Board Cleanup” section if you are getting Graham/Lindsey to cut the circuit out for you using the bandsaw.

To cut out a board, we need to do some manual tweaking of G-Code. To start, import the PCB outline gerber. Don’t make any changes until you get to the toolpath generation. When you get to the toolpath generation, make sure to change the cutter diameter to 1.5 mm and the cut depth to 0.1 mm.



Notice that if we were to cut this path directly, we would end up with a cut outside **AND INSIDE** our PCB outline, which will probably destroy our PCB. We would also be trying to cut through the entire PCB in one pass which would probably break the endmill.

For now, click through the rest of the wizard. Your outline will get imported and be placed next to the PCB we just milled. Make sure you know where you saved the .fnc file.



Click on **Program List** on the left hand window and right click on your edge cut program (in the above example, mine is called *PCBExample-Edge\_Cuts.gbr*) and delete it. Now, in a text editor (e.g. Notepad) open the .fnc file we just created to cut out the board. You'll have a file similar to the one shown below.

```
(Denford Gerber to NC Wizard
-----
(Filename: PCBExample-Edge_Cuts.gbr
(Date: 29/08/2015
(Time: 14:47:18
(Board Side: Component Side
(Track Cutting Depth: 0.000mm
(Track Cutting Tool: T1 Diameter: 1.500mm
[TOOLDEF T0101 D1.5000
(Drilling Depth: 0.100mm
-----
G21
[BILLET X16.160 Y24.030 Z1.600
G90
G91 G28 X0 Y0 Z0
M05
G90 M6 T1
M03 S23000
G0X0Y0Z5.000
```

```
G1Z5.000F600.000
G0Z5.000
G0X0.800Y19.526
G1Z0.100
X0.800Y19.526
X0.800Y0.794
X11.900Y0.794
X11.900Y19.526
X0.800Y19.526
G0Z5.000
G0X13.500Y20.320
G1Z0.100
X13.500Y20.320
X13.500Y0.000
X13.477Y-0.191
X13.408Y-0.372
X13.299Y-0.530
X13.154Y-0.658
X12.984Y-0.748
X12.796Y-0.794
X12.604Y-0.794
X0.096Y-0.794
X-0.096Y-0.794
X-0.284Y-0.748
X-0.454Y-0.658
X-0.599Y-0.530
X-0.708Y-0.372
X-0.777Y-0.191
X-0.800Y0.000
X-0.800Y20.320
X-0.777Y20.511
X-0.708Y20.692
X-0.599Y20.850
X-0.454Y20.978
X-0.284Y21.068
X-0.096Y21.114
X0.096Y21.114
X12.604Y21.114
X12.796Y21.114
X12.984Y21.068
X13.154Y20.978
X13.299Y20.850
X13.408Y20.692
X13.477Y20.511
X13.500Y20.320
G1Z5.000
M03 S23000
G91 G28 X0 Y0 Z0
M30
```

This might look complicated at first, but it's quite easy to spot the part we're interested in once you get used to it. I've highlighted it in the code above. Here's a quick overview of what the highlighted section of code does:

1. ***G0Z5.000*** tells the machine to move the endmill to the safe height of 5 mm
2. ***G0X0.800Y19.526*** then tells it to move to the starting position of the first cut.
3. ***G1Z0.000*** drops the endmill to the cutting depth (in this case 0.1 mm).
4. The XY commands like ***X0.800Y19.526*** tell the endmill to do an absolute move by a set distance in each axis.
5. Again, ***G0Z5.000*** causes the endmill to retract to the safe height, ready to move and start the next cut. This

Because our board outline file only produced two cutting paths, it's pretty easy to tell what code we need to delete in order to remove the cut on the inside of the line. In the example above it's the code in yellow. If you highlight each block of code between ***G1Z5.000*** commands, typically the shorter of the two blocks will be the inside line, and the longer of the two blocks will be the outside cut we want to actually do.

If you need help telling what lines are what, I recommend copy-pasting your .fnc file into an online g-code simulator like <https://ncviewer.com/>. That way, you can make changes and see their effects quickly!

Go ahead and delete the code highlighted in yellow above and save it as a new .fnc file. To check we got it right, click the **Add CNC** button and choose the file we just saved. You should see that we have successfully removed the inside cut and just kept the outside one.

We still have a problem though – right now if we tried to do the board online cut, it would try and cut all the way through the board in one pass and break the endmill. To fix this, re-open the .fnc file we just made in text editor.

We already identified the code that cuts the correct shape, and we know that the command ***G1Z0.100*** will control how deep the cut is. So, we can **copy and paste the code for the shape, and just adjust the cut depth each time**. I like to cut about 0.6mm per pass, and to do a first pass that just hits the copper and no more (it gives a slightly neater edge on the final copper cut). Boards are about 1.6 mm thick, so overall I tend to do my cuts as **4 passes: -0.1 mm, -0.7 mm, -1.3 mm and then -1.9 mm** to make sure I get all the way through thicker boards. In the end your program will look something like the code below. Note that lines starting with an open bracket are comments in G-Code. I've stripped out the XY moves to keep the code smaller here.

```
(some header code
(1st pass
G0X13.500Y20.320
G1Z0.100
(a bunch of XY moves)
G1Z5.000
(2nd pass
G0X13.500Y20.320
G1Z-0.600
(a bunch of XY moves)
G1Z5.000
(3rd pass
G0X13.500Y20.320
G1Z-1.200
(a bunch of XY moves)
G1Z5.000
(4th pass
```

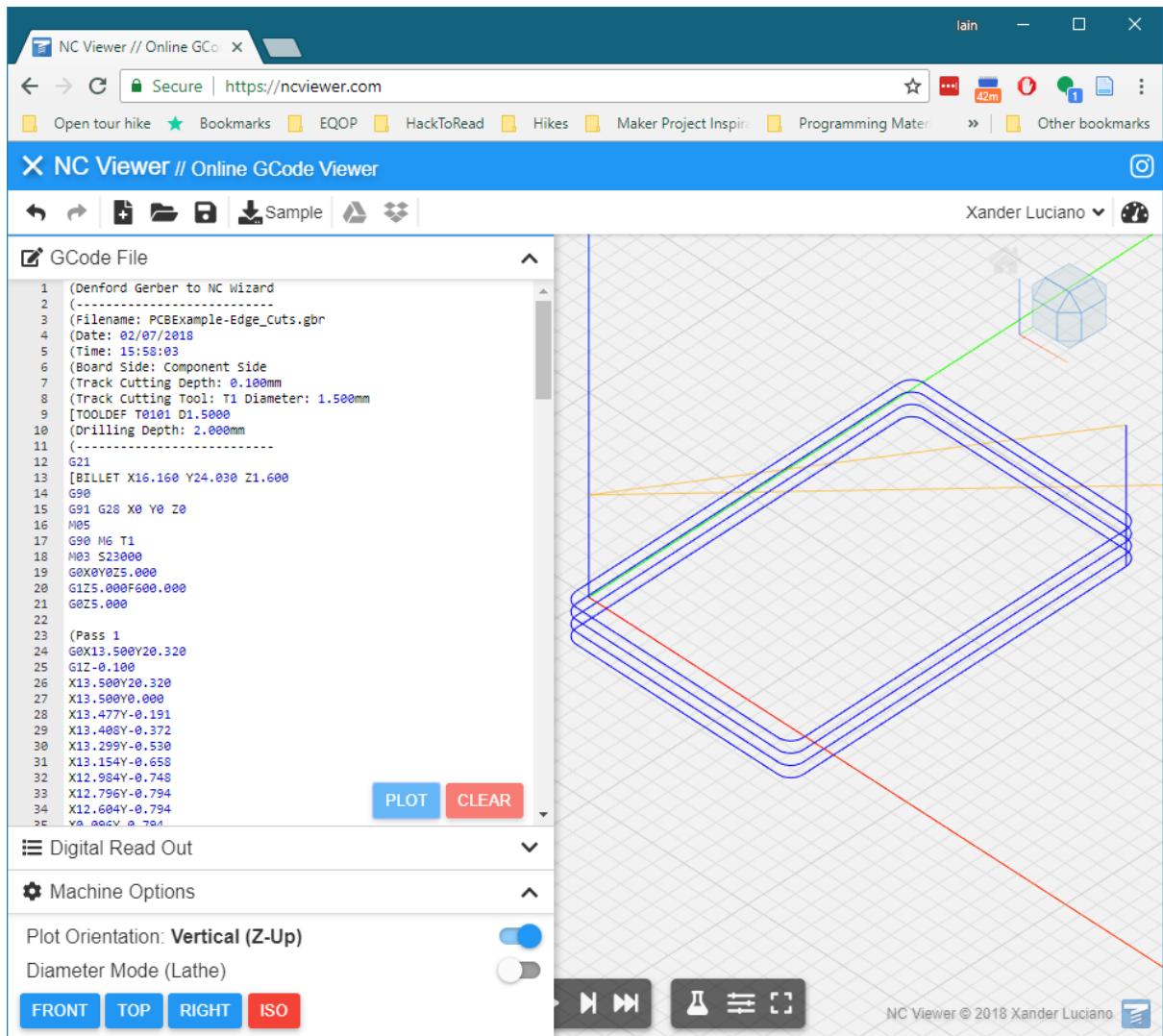
```

G0X13.500Y20.320
G1Z1.900
(a bunch of XY moves)
G1Z5.000
M03 S23000
G91 G28 X0 Y0 Z0
M30

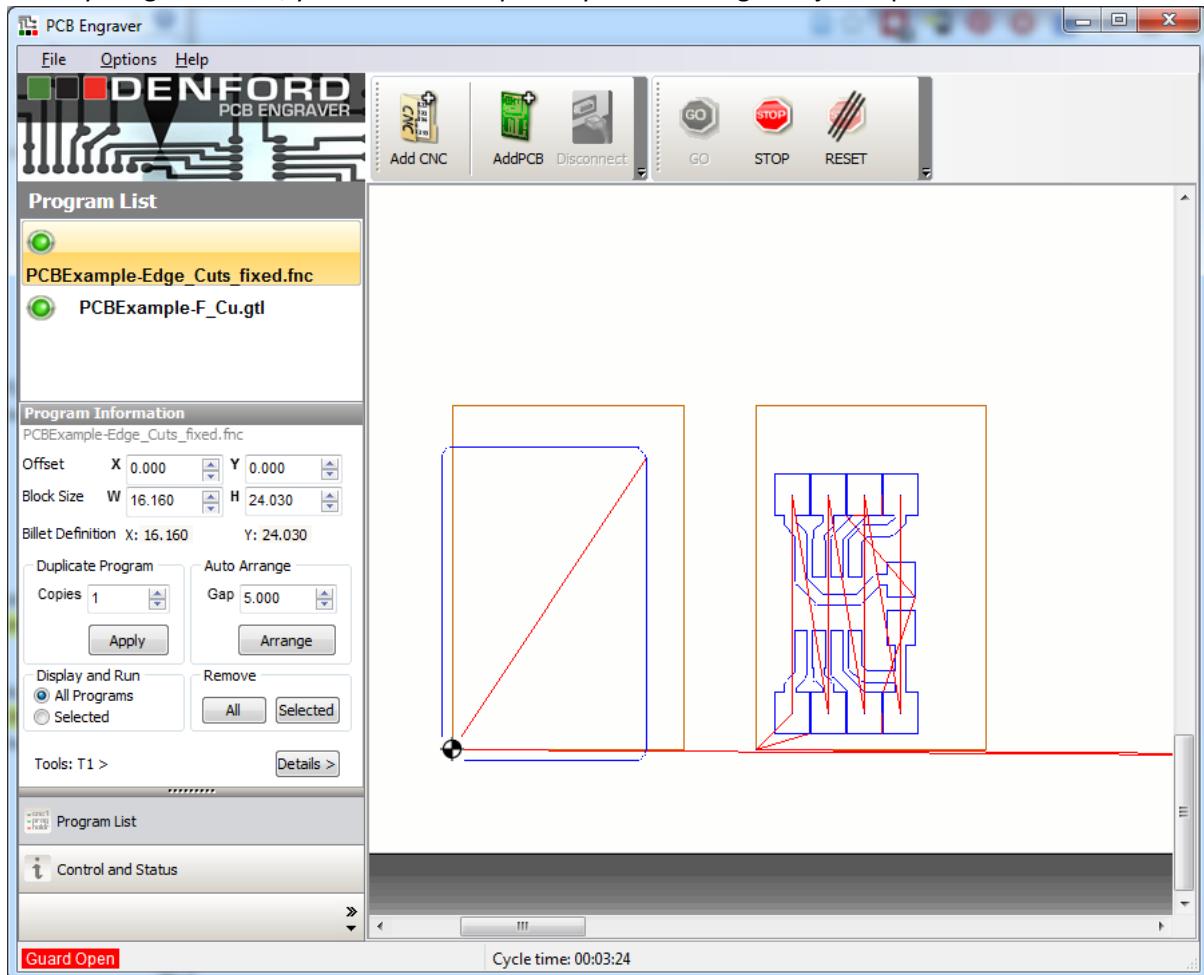
```

If you want to take a look at an example of an original edge cut file and a fixed one, take a look at the files “PCBExample-Edge\_Cuts\_original.fnc” and “PCBExample-Edge\_Cuts\_fixed.fnc”.

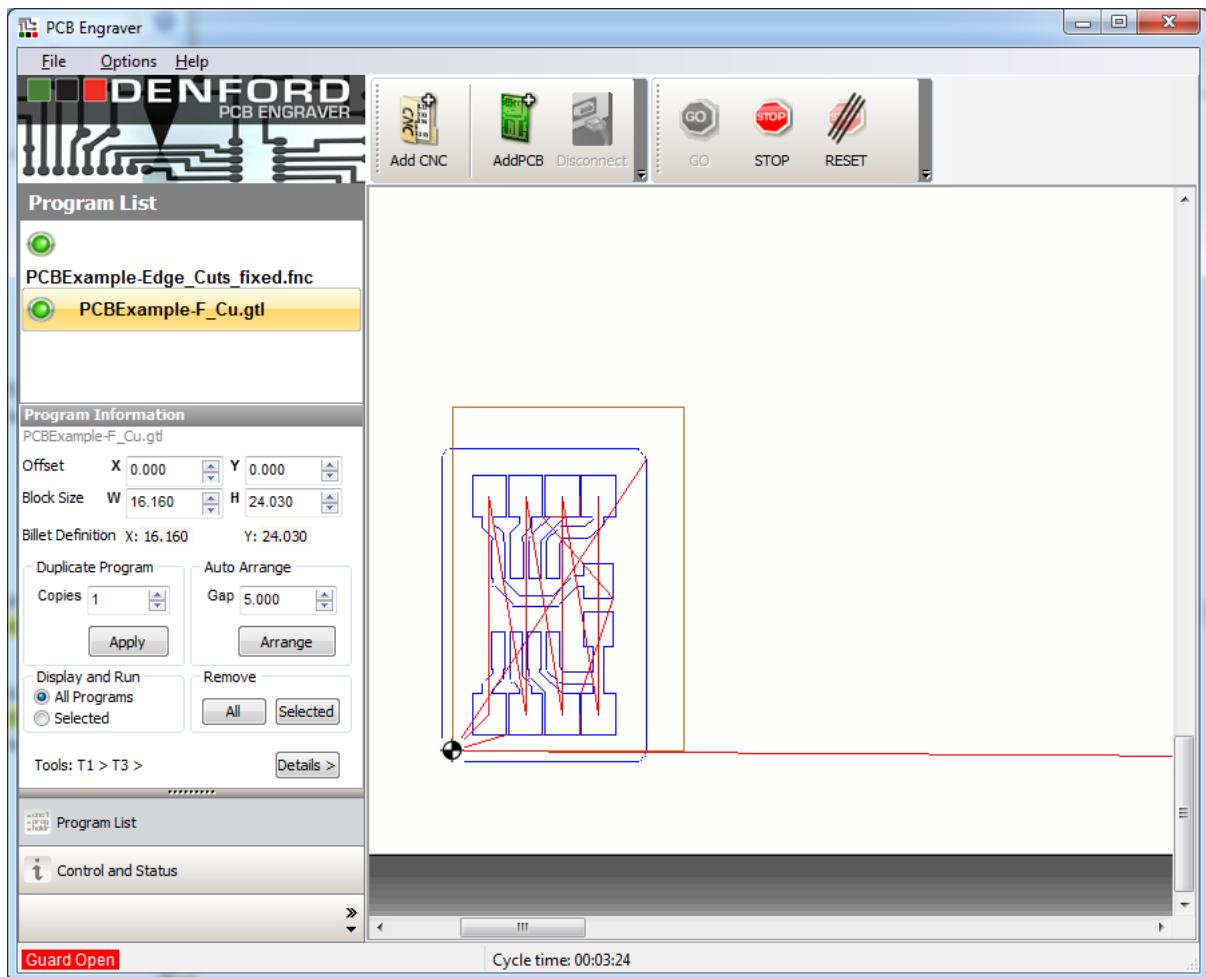
If you used a g-code simulator, you should see the multiple steps of the cut.



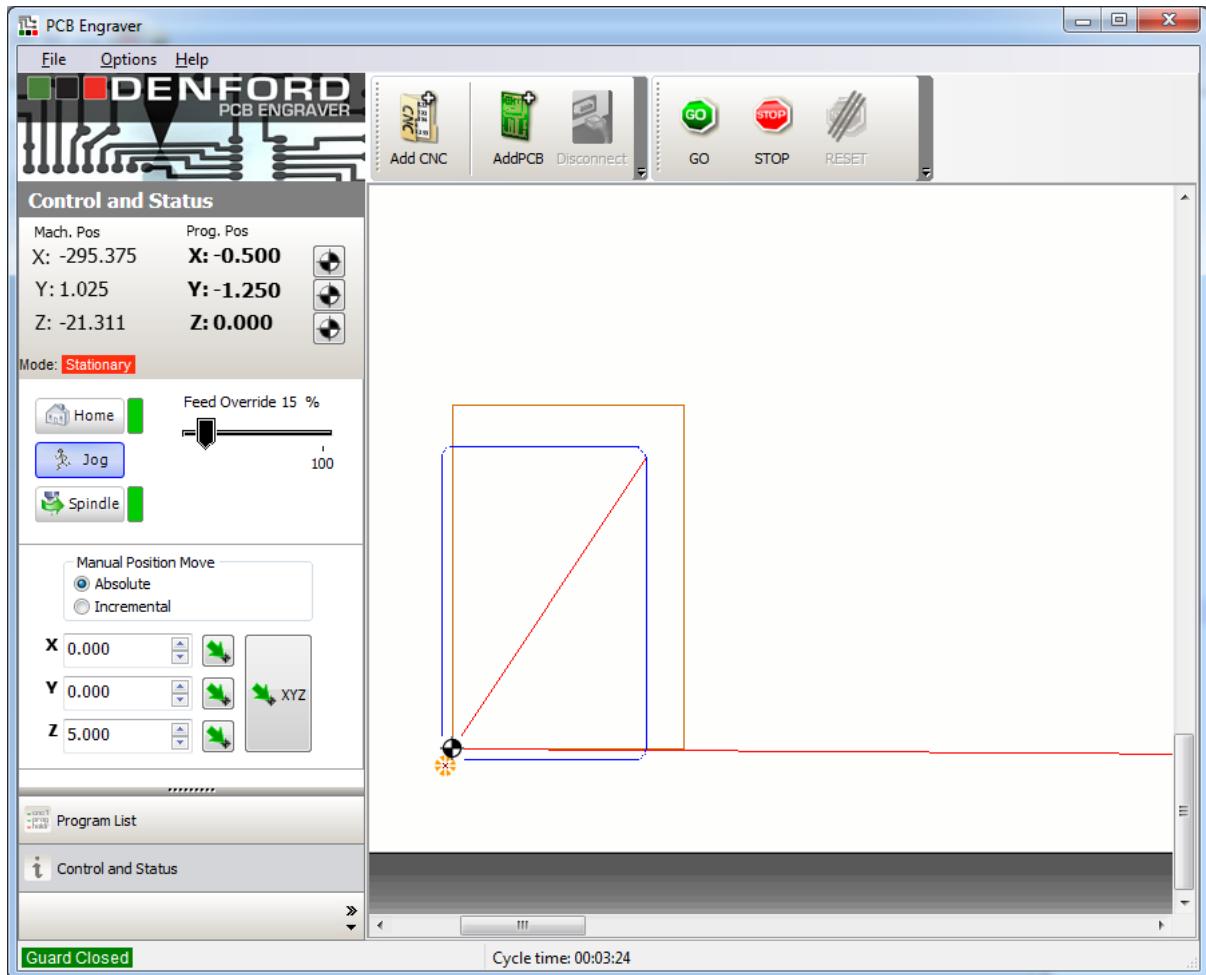
If everything went well, you should end up with your fixed edge cut job imported as shown below.



To align the two jobs, simply select each in turn and make sure that the **Offset** setting is at 0 for the two jobs. You should end up with the two jobs on top of each other as shown.



The last thing to do is in the **Program List** tab on the left hand side of the screen, switch the **Display and Run** option from **All Programs** to **Selected**. Make sure you have your edge cut job selected as shown below.

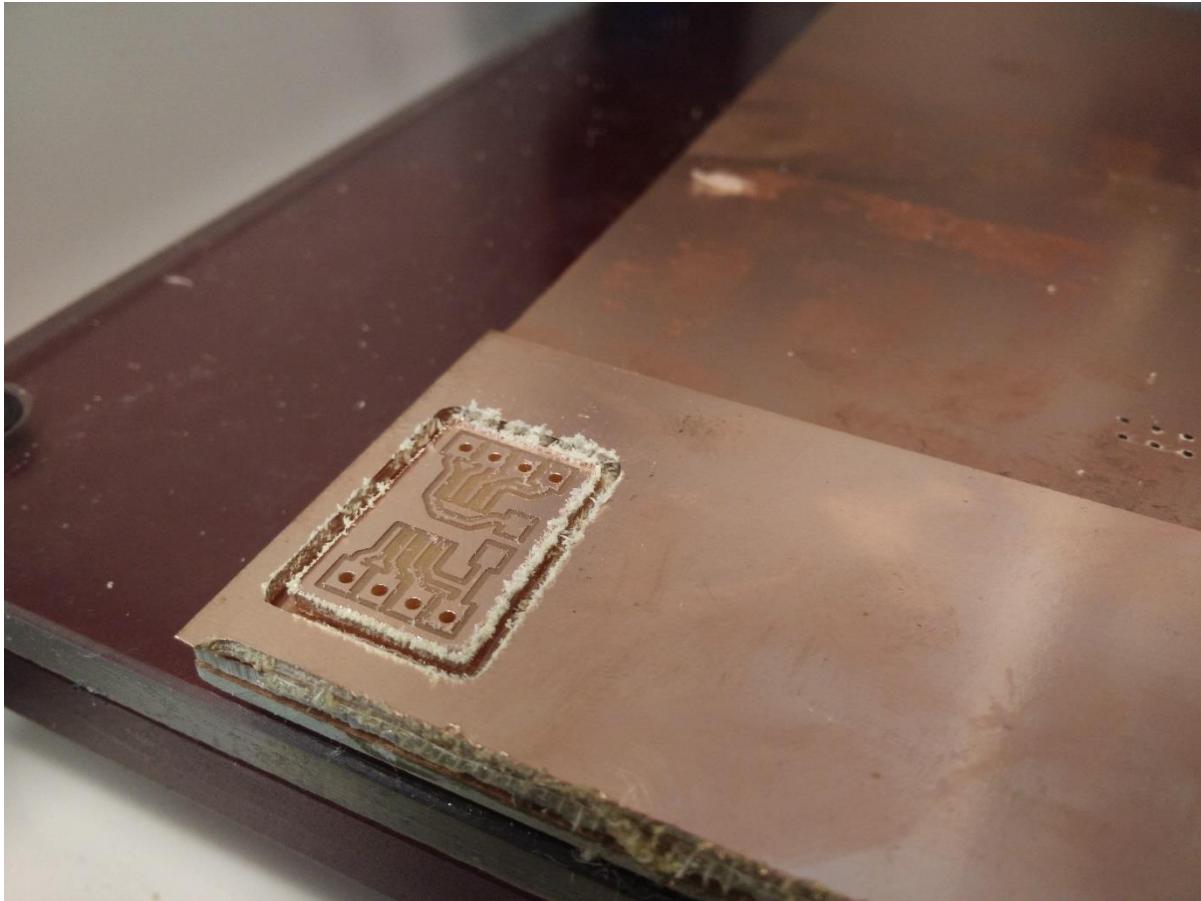


When you're ready, turn on the vacuum and hit **GO**. Keep an eye on the machine to make sure it's doing what we expect. We hand edited the G-Code so there is a good chance we made mistakes! If it looks like something is wrong, hit the emergency stop.

When the job is finished, you should have one nicely cut out PCB! The next job is to clean up the PCB, ready for soldering.

## Cleaning the PCB

Depending on how new the endmill was, our PCB might have a lot of burrs (sticking up bits of copper that haven't quite detached from the traces). I was using a new endmill when I wrote this so the burring on the traces wasn't minimal. This is what my board looked like before I removed it from the engraver.

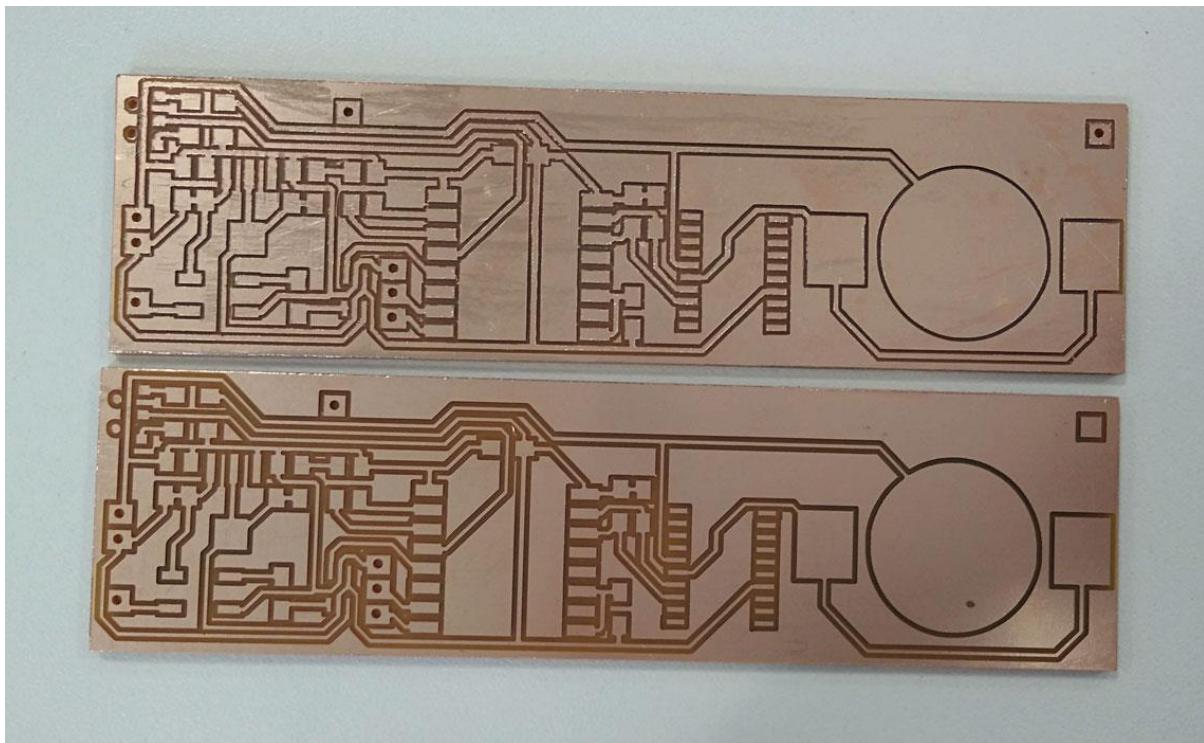


You can see little flecks of copper around the traces, and what looks like a mess around the board outline (it's actually just from the double sided tape). I used the metal spatula thing for getting 3D prints off the 3D printer.



I then use the flat edge of the spatula to scrape down the edges of the board to remove the worst of the burrs and stuck tape, and then gently draw it across the top surface of the PCB to remove any big burs on the traces. If you were unfortunate enough to be using a really old or damaged endmill, you might even want to use some high-grit wet sandpaper to clean up the top copper. It's also worth doing a good visual inspection to make sure all your traces are properly cleared. You can use a craft knife to clean up any dodgy looking sections.

The picture below shows an example of the same board cut with an old endmill (top) and a brand new endmill (bottom).



The top board was using an old bit and had a **LOT** of burrs. It would have needed a significant amount of work to clean it up. In comparison, the bottom board was cut using a new endmill. It needed almost no post processing at all, and has a super crisp cut.

It's pretty hard to know how a 0.4mm endmill is fairing just by looking at it. I think the easiest way to spot a mill at the end of it's life is based on:

- **Track width vs clearance width:** I use 0.4mm tracks as a standard, and a new 0.4mm bit should give a clearance of 0.4mm. As the mill wears out, it's cutting width decreases. On the bottom board, you can tell that the traces and cuts are about the same width. On the top board, the traces are significantly wider.
- **Burrs:** Although some burrs are to be expected, if you feel like you would have to clean up a lot of traces with a craft knife then there is a good chance the cutter is past it (or you have your settings totally wrong!).

Once you're happy, go wash the board in hot soapy water and give it a good scrub with an old toothbrush. This helps clear out any little bits of copper caught between traces, and removes the oils from your fingers. If you don't wash the board, the oil from your skin reacts with the copper over time. This makes the copper very difficult to solder on, and can even cause so much damage that eventually traces get worn away. Just take a look at the sacrificial copper in the machine – you can usually see dark fingerprint marks all over it!

And that, finally, is us done! Go solder your masterpiece!