

CNC Mini Mill Assembly Manual

Parts List:

The design is somewhat flexible in terms of size. The Z-size is currently limited to 27mm, but the X and Y axis can be whatever size you choose. Keep in mind that the 8mm rails are fairly light for this application, so I wouldn't recommend going much larger than 150mm in either axis.

For the 3D printed parts, go to www.github.com/t�og/cnc_mini_mill, download the .STL files, and print off every part. Some parts are required multiple times, so follow the printing guide. There's a lot to print!

The non-3D printed parts (Vitamins, as the RepRap community calls them) are as follows:

- A spindle (search eBay for 500W High Speed PCB Engraver Air Cooled Spindle Motor 52mm ER11 Router R1Q1)
- A base block that is 150mm longer than your desired length, 19mm wider than your desired width, and at least 50mm (2") thick. I used 8" C-channel with 1/4" spacers on each side, giving it a total width of 8.5" (215mm), giving it a working width of 7" (175mm).
- Two vertical risers
- 8mm rods: Two as long as the base, two as wide as the base PLUS the width of the vertical risers PLUS 30mm, and two 175mm long
- Twelve LM8UU bearings
- Two 8mm x 4mm lead Acme lead screws, 300mm long (each with an anti-backlash nut)
- One 8mm x 2mm lead Acme lead screw, 200mm long (with an anti-backlash nut)
- Six 608vv bearings
- Small assortment of 8mm or 5/16" washers (maybe 10 or so)
- GT2 Pulleys:

- Two 60-tooth 8mm-bore
- One 40-tooth 8mm-bore
- One 20-tooth 5mm-bore
- Two 16-tooth 5mm-bore
- GT2 Belts:
 - One 122mm
 - One 154mm
 - One 180mm
- Three NEMA 17 x 34mm stepper motors
- An assortment of M3 fasteners (up to 20mm long)
 - A quick count says about 80 screws and almost that many nuts, maybe I can eventually add a breakdown on the lengths
- 13 M6 screws and nuts
 - 7x 25mm long
 - 6x 20mm long (although you can just use 25mm long instead to keep the parts list simple)

Now here's the thing about the electronics: my setup works great for me, but it is very minimalistic. You should consider using a RAMPS or some other platform as well. In case you do want to go super cheap, I will describe how my platform is set up. You will need:

- An Arduino Uno or Nano
- Three A4988 stepper drivers (DRV8825 work as well)
- A 2004 LCD display
 - With an I2C adapter. I stole the adapter from the back of a 1602 display and soldered it onto my 2004 display
- A joystick
- An optoisolated relay
- A [power switch](#)
- An [outlet](#)

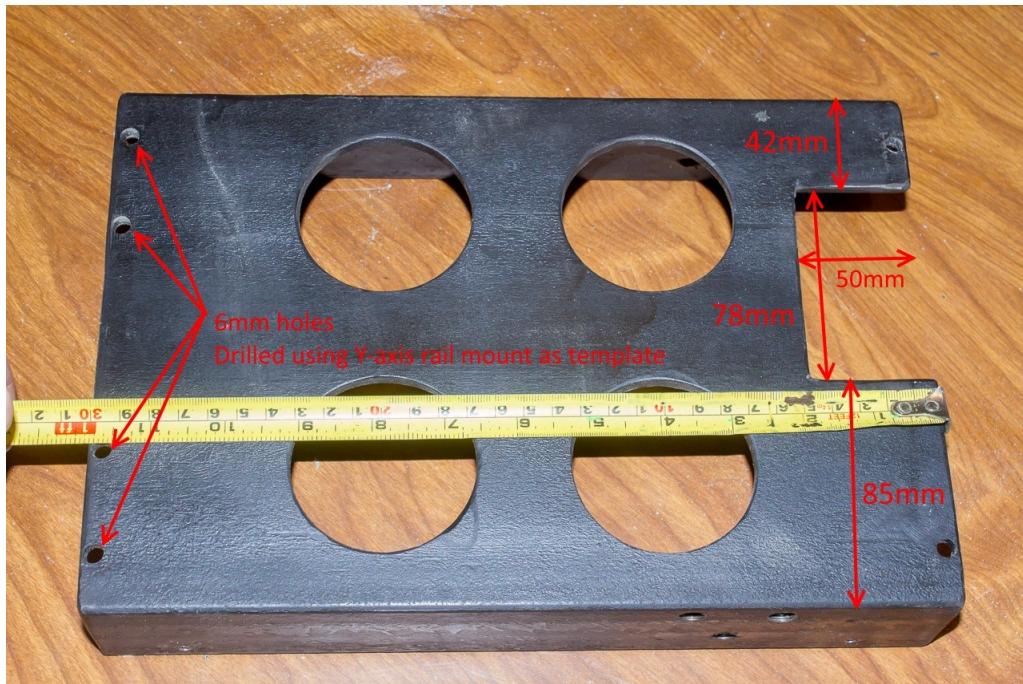
Tools Required:

Make sure you have the following tools ready:

- Metric allen wrench set
- Metric wrench set
- Soldering iron & solder
- Wire stripper
- Hacksaw
- Drill & drill bits
- Tape (masking, electrical, or duct)

Base:

Here's my 8" C-channel. I cut it 11.75" long. The large holes in the middle are to allow access to the carriage's underside (when it's all put together). The 6mm screw holes were drilled using the Y-axis rail mounts as a template (I'll get to that later). The rectangular cutout is for the Y-axis motor. You'll need to cut it out as well.

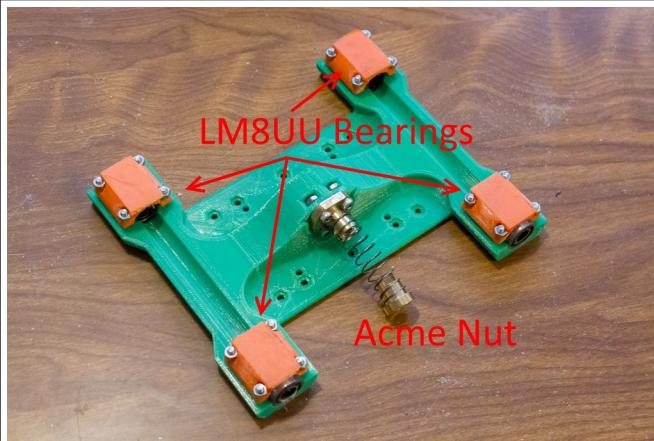


If you do find a section of 8 in C-channel for the base, print four feet and install them on the edges.



Table

Print the Table and 4x Y-axis Bearing Clamps. Assemble the table (the big green part in the photos below). You'll see how the Acme nut has been filed down on one side. You need to do that, it might be easier to do it after you screw it onto the table.



Y-Axis Rails

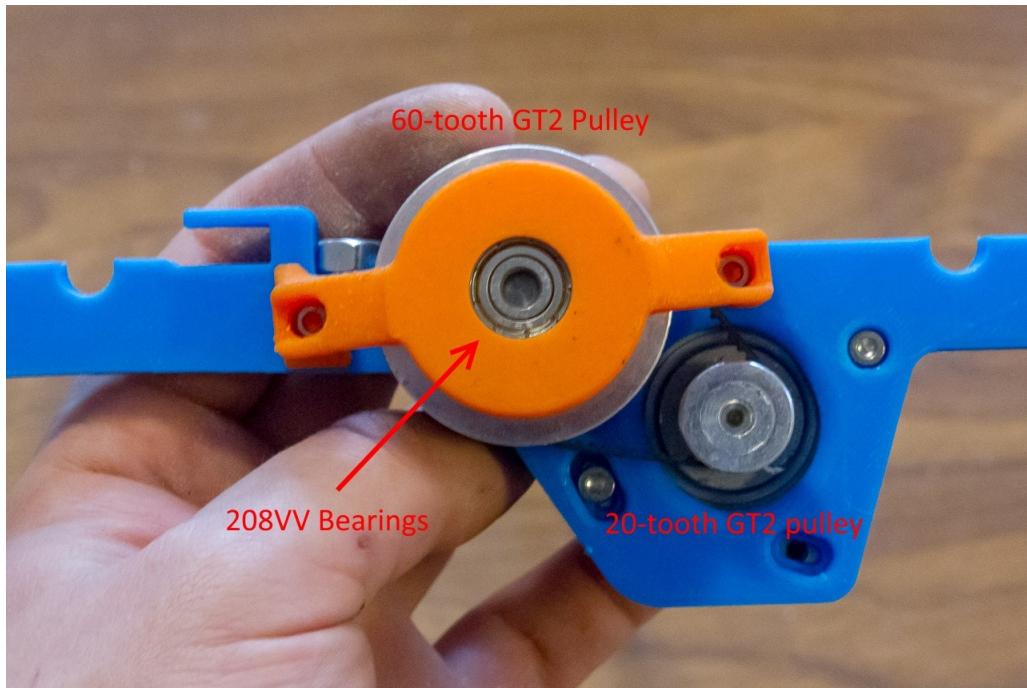
Note: Several parts shown in this section are different than the parts in other sections. I've been constantly improving the design, even while photographing the assembly. In particular, the belt and pulleys have been changed since some photos were taken.

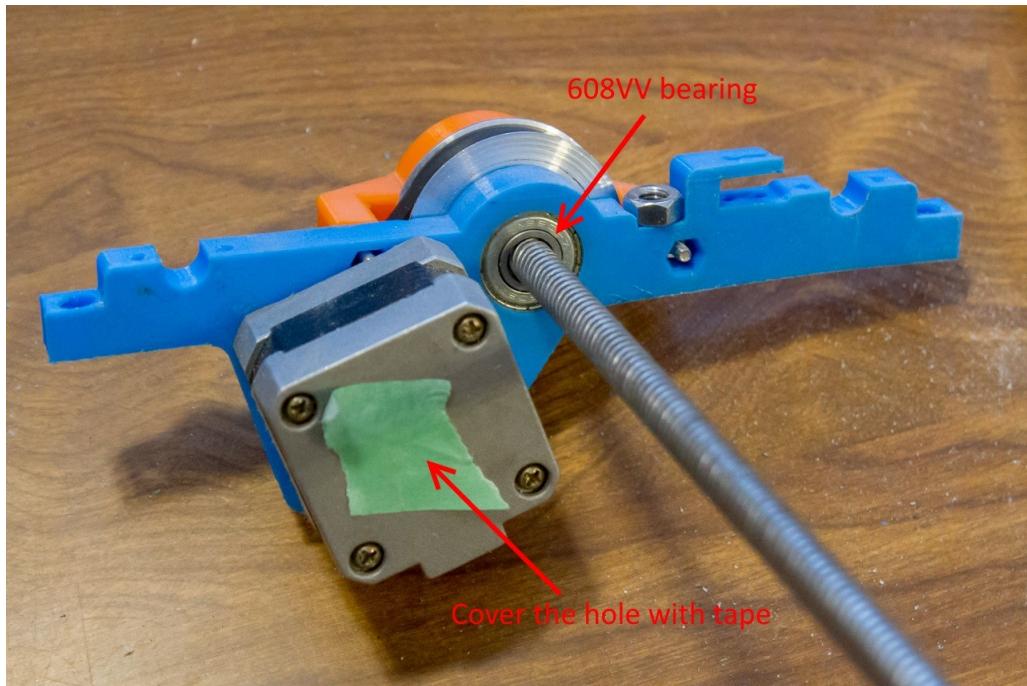
The 40-tooth GT2 pulley on the Y-axis will need to be modified a little bit. The belt guides are too large and will interfere with the table or workpiece. Grab a 5/16" or M8 screw and mount the pulley on the screw. Then put the screw in a drill and use some sandpaper, a file, or a grinder to turn the sides down until they're just tall enough to guide the belt. Below are my before/after photos, one of the belt guides fell right off! That's ok though, I don't expect the belt to fall off the pulley because of that.



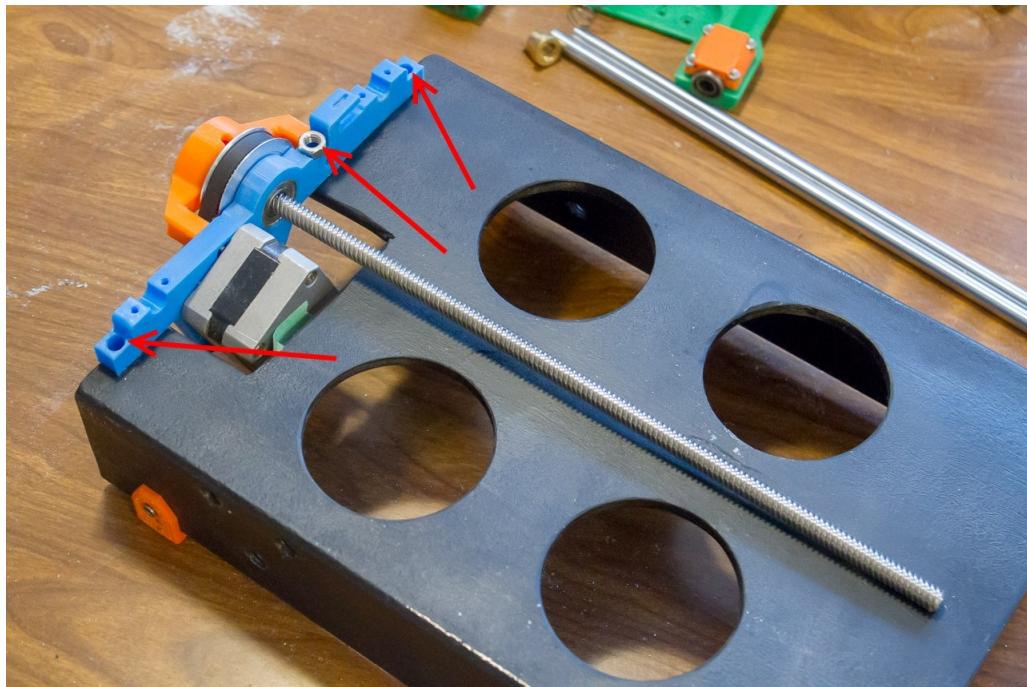
An Acme lead screw will attach to the mount through a 608VV bearing. Use some washers to space the 40-tooth GT2 pulley, and use the clamp with another 608VV bearing installed to secure the lead screw. The motor gets a 16-tooth GT2 pulley and a 122mm belt goes between the motor and the lead screw. Cover the shaft hole on the rear side of the motor with some tape!

NOTE: These are photos of an earlier mount which used a 60-tooth and 20-tooth GT2 pulley. Need to replace with updated photos.



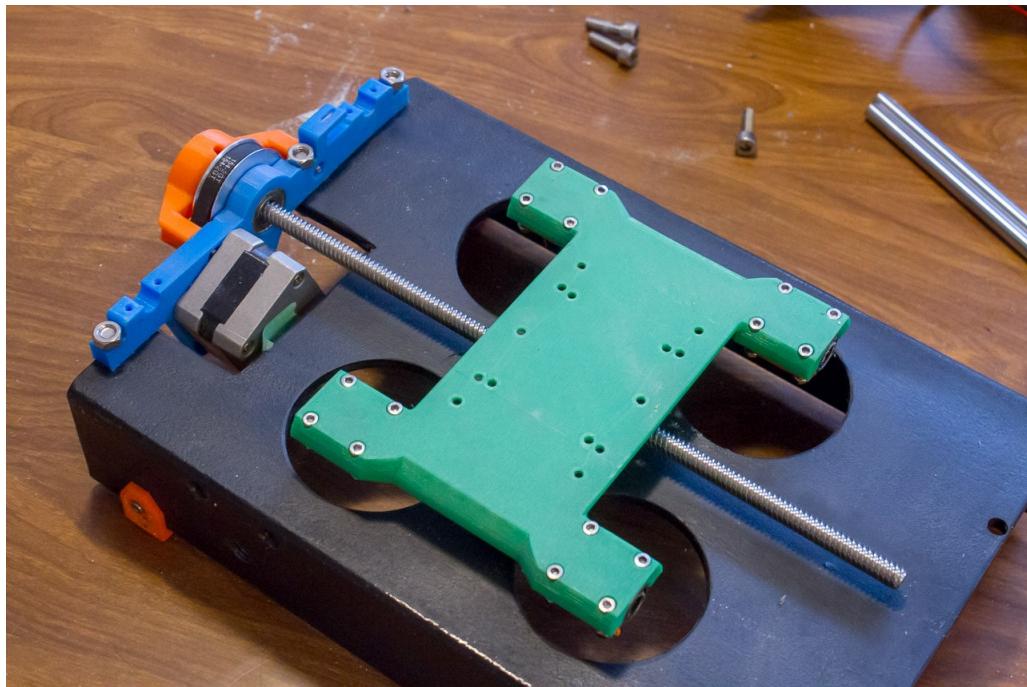


Position the Y-axis rear rail mount on the base. Drill the three 6mm mounting holes in the base using the mount as a template. I would suggest starting the holes with the template, then remove the template to finish them. That way you won't melt the template by the heat of the drilling process. If you don't have a metric drill bit set, use a 15/64" bit (it's like 5.95mm) and work it around a bit to slightly increase the diameter.



Screw the Y-axis rear mount using M6 screws and nuts. If you are using the 8" C-channel, the middle screw will need to be 25mm long and the other two 20mm long - but you can always use longer. I decided to use socket-head screws here, but regular hex head will be fine.

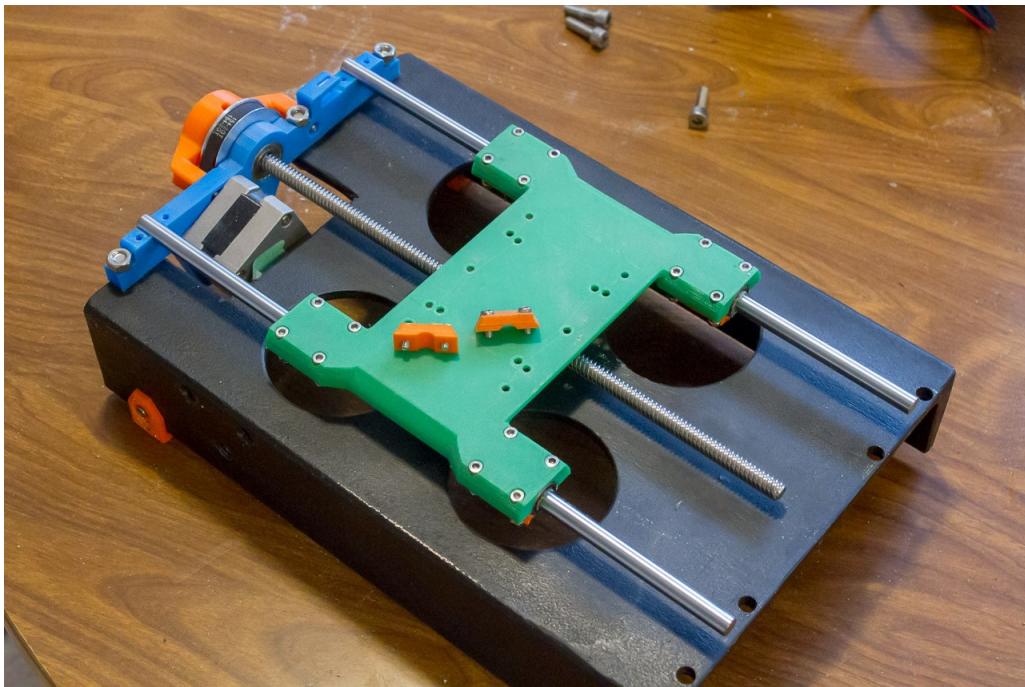
Now screw the assembled table onto the Acme lead screw.



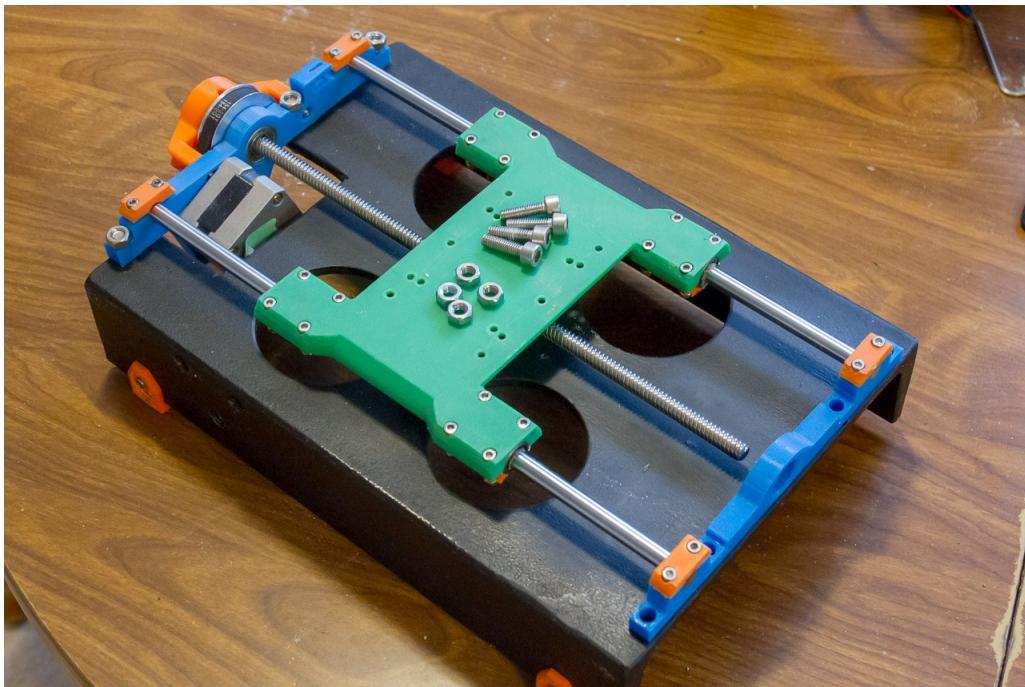
Print the Y-axis front mount and 4x Rail Clamps. Grab two of the 8mm rails. Screw the rail clamps onto the mount using 16mm M3 screws, but don't tighten them all the way. NOTE that the front mount shown below is a deprecated design - the hole in the middle doesn't exist on the current design.



Slide the 8mm rails through the table and onto the rear mount. The other two rail clamps that you printed will go on the rear mount.



Screw the rail clamps on the rear mount. Slide the front mount onto the rails. When you have it positioned, drill the 6mm holes for the front mount.

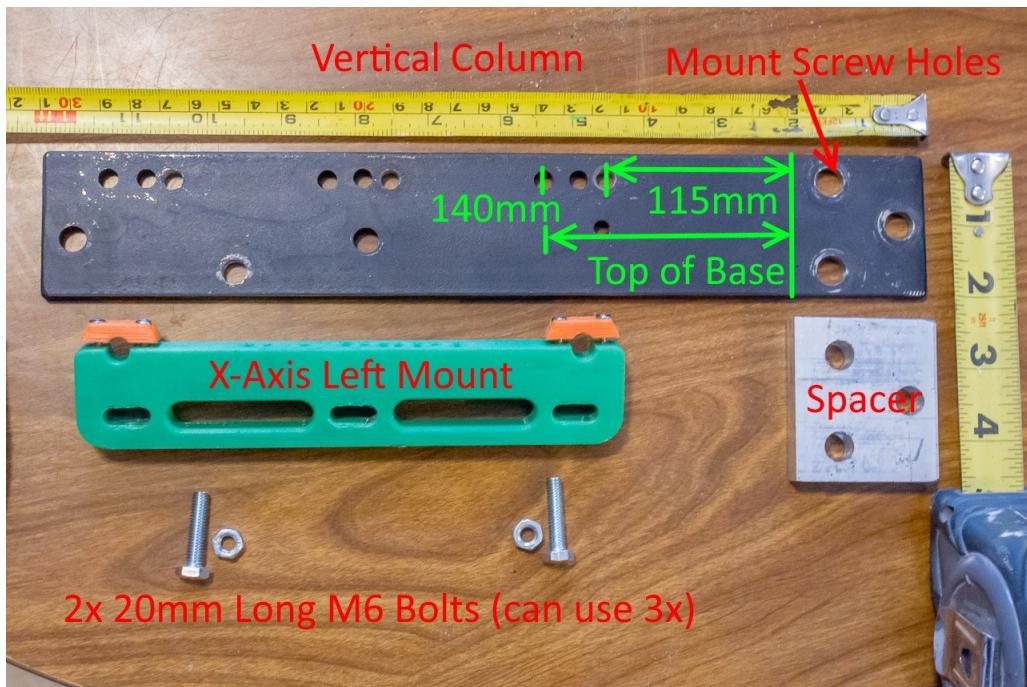


Screw the front mount down. The Y-axis is finished!

Vertical Columns

I used 2" x 1/4" flat iron for the vertical columns. They were cut about 310 cm (12") long, which means they rise about 255 cm (10") above the top of the base. I added two 1/2" thick pieces of aluminum as spacers between the vertical columns and the base, but that is completely optional. It just makes the work area another inch wider.

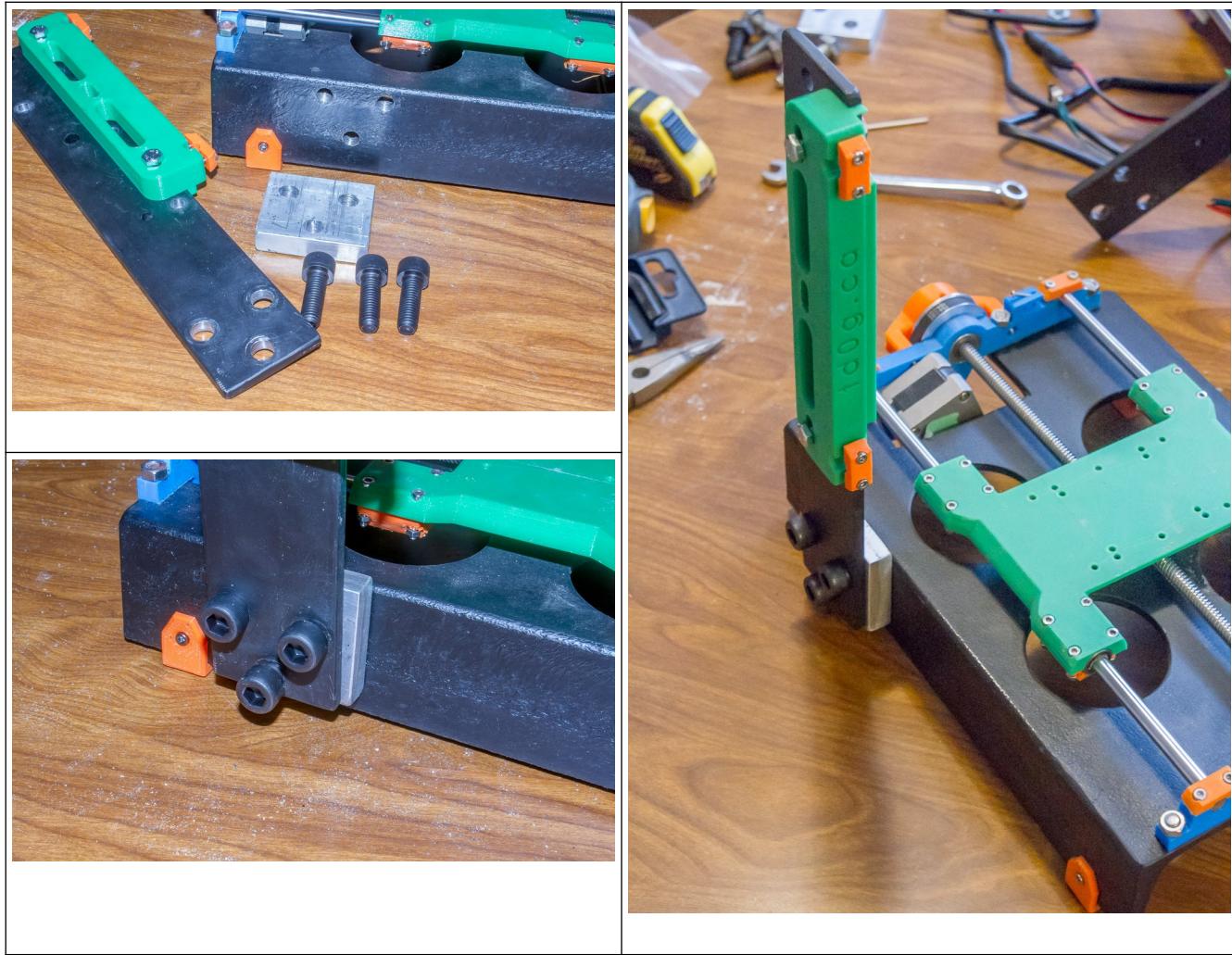
Print the X-Axis Left Mount and 2x Rail Clamps. The mount is secured to the column using two or three 20mm long M6 bolts and nuts. You can use the mount as a template for marking or for drilling the holes. The holes on the mount is slotted so you can easily level the gantry later.



Once the holes are drilled, fasten the rail mount to the column using two or three M6 screws. Screw the rail clamps to the rail mount using 16mm M3 screws and nuts. The nuts slide into slots which can't be seen on the photos here (older version shown).



Now you need to fasten the column to the base. I opted to use M10 fasteners. First, I drilled three holes in the column. Then I clamped the column onto the base, and drilled those three holes through the base. Then I tapped the holes in the base (to save some work, you could just drill them out to 10mm and use nuts), and I drilled the holes in the column to 10mm.



Left side done!

Set up the right side column so that it mounts to the base (same steps as the last paragraph). make sure that it's squared off with the left side column!

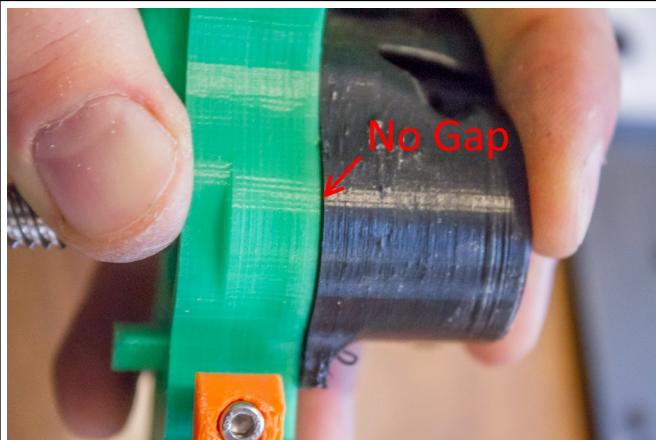
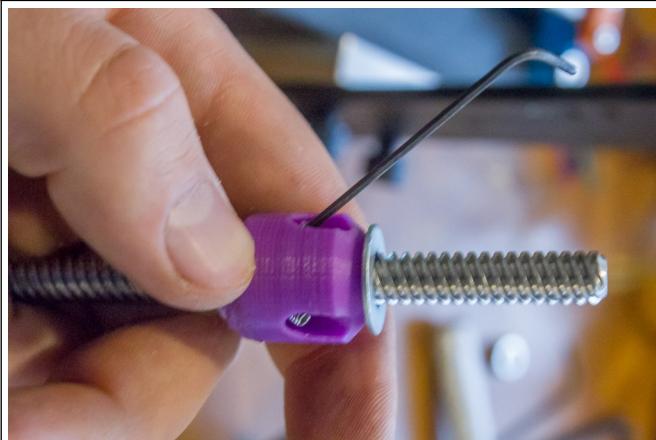
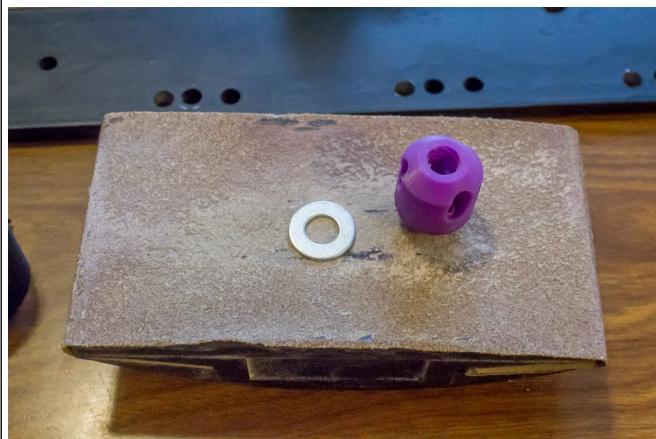
Print the X-Rail Right Side Mount, the X-Rail Bearing Clamp, and 2x Rail Clamps. Screw the rail clamps onto the side mount. Grab two 608VV bearings, press one into the side mount and one into the bearing clamp.



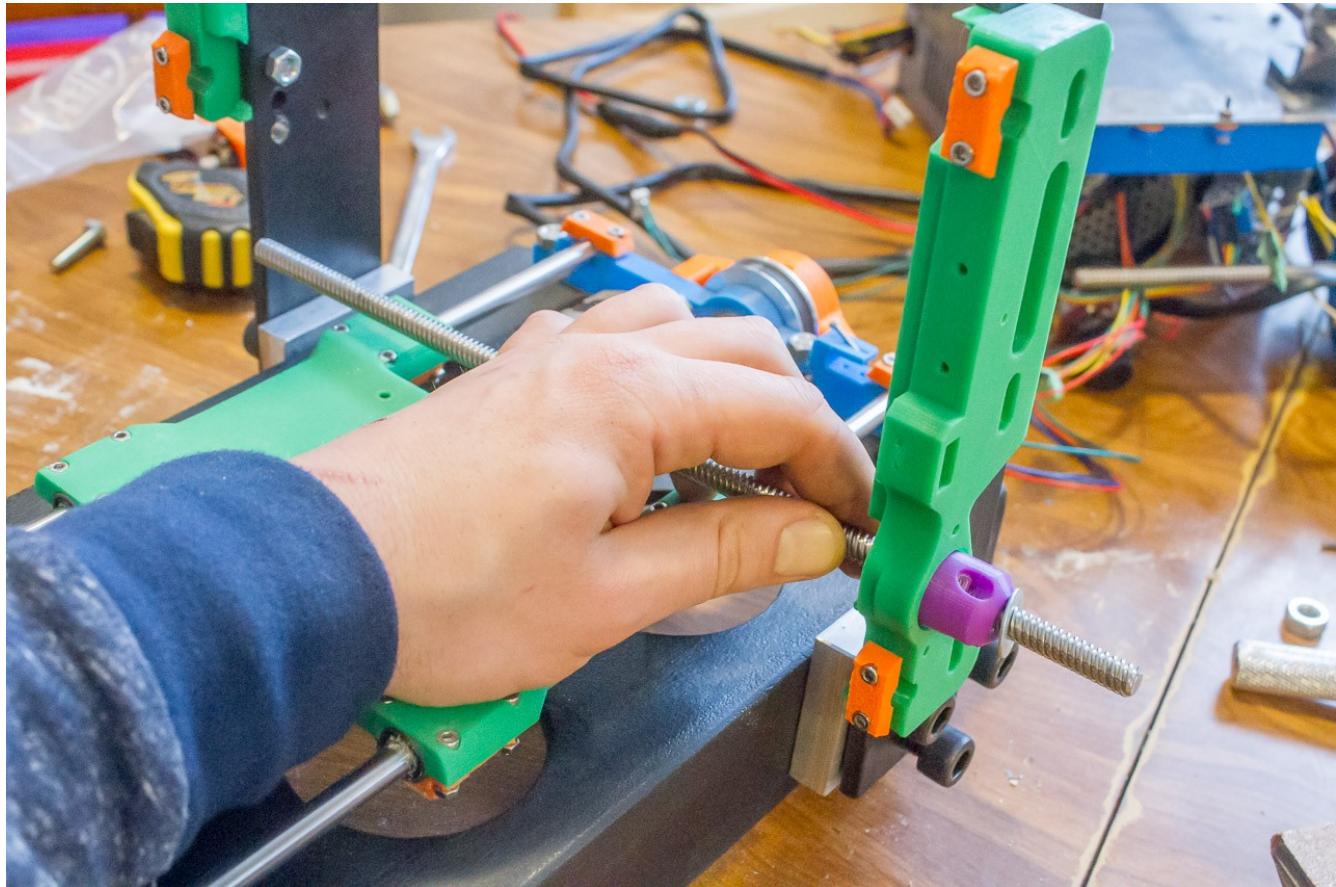




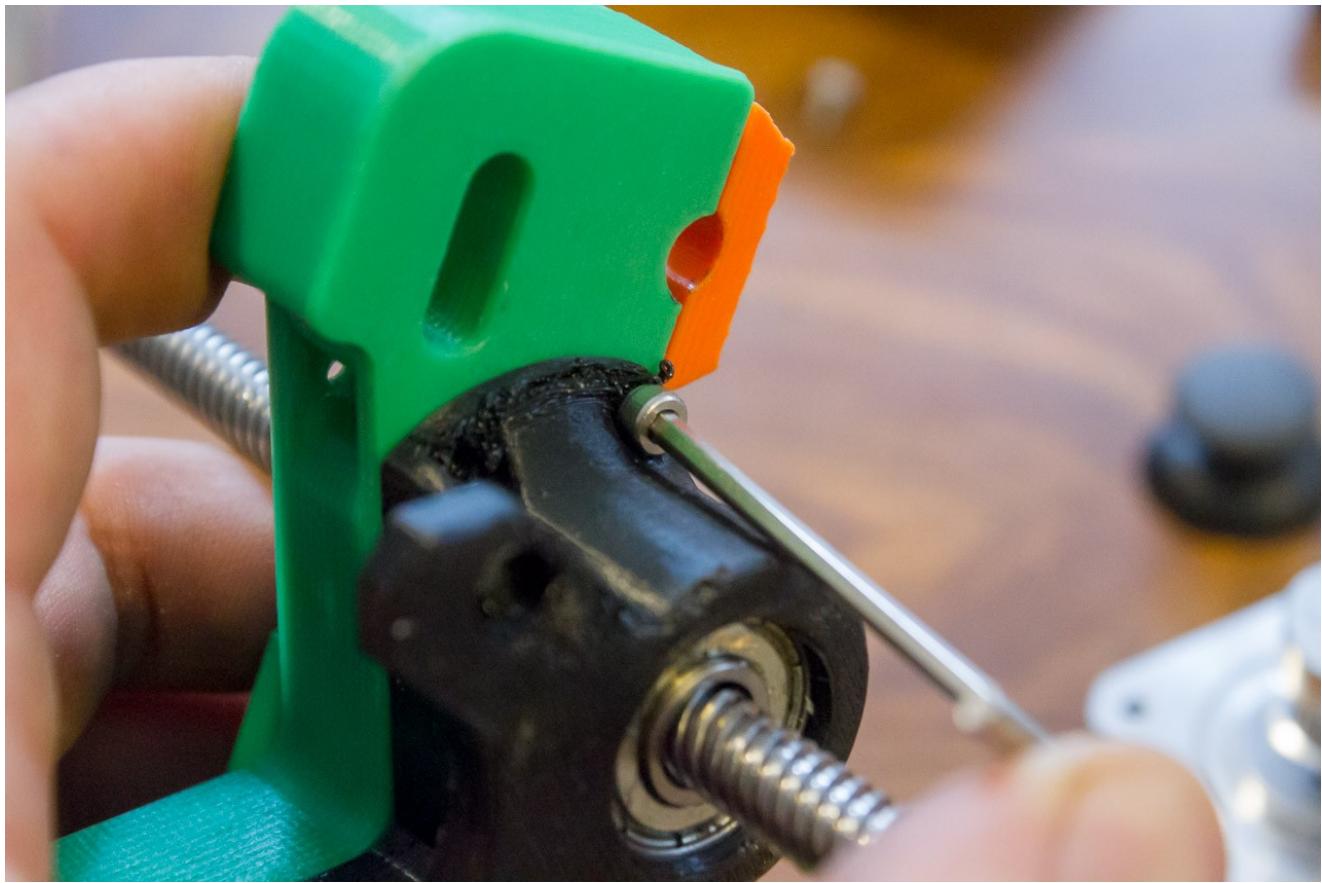
Print the X-Rail Acme Nut. Thread it onto the Acme lead screw. Test it inside the rail mount and bearing clamp - there should be no side-to-side play. If there is, you'll need to add a washer and sand the printed nut down until it JUST fits. The bearing clamp should fit against the side mount without any gap

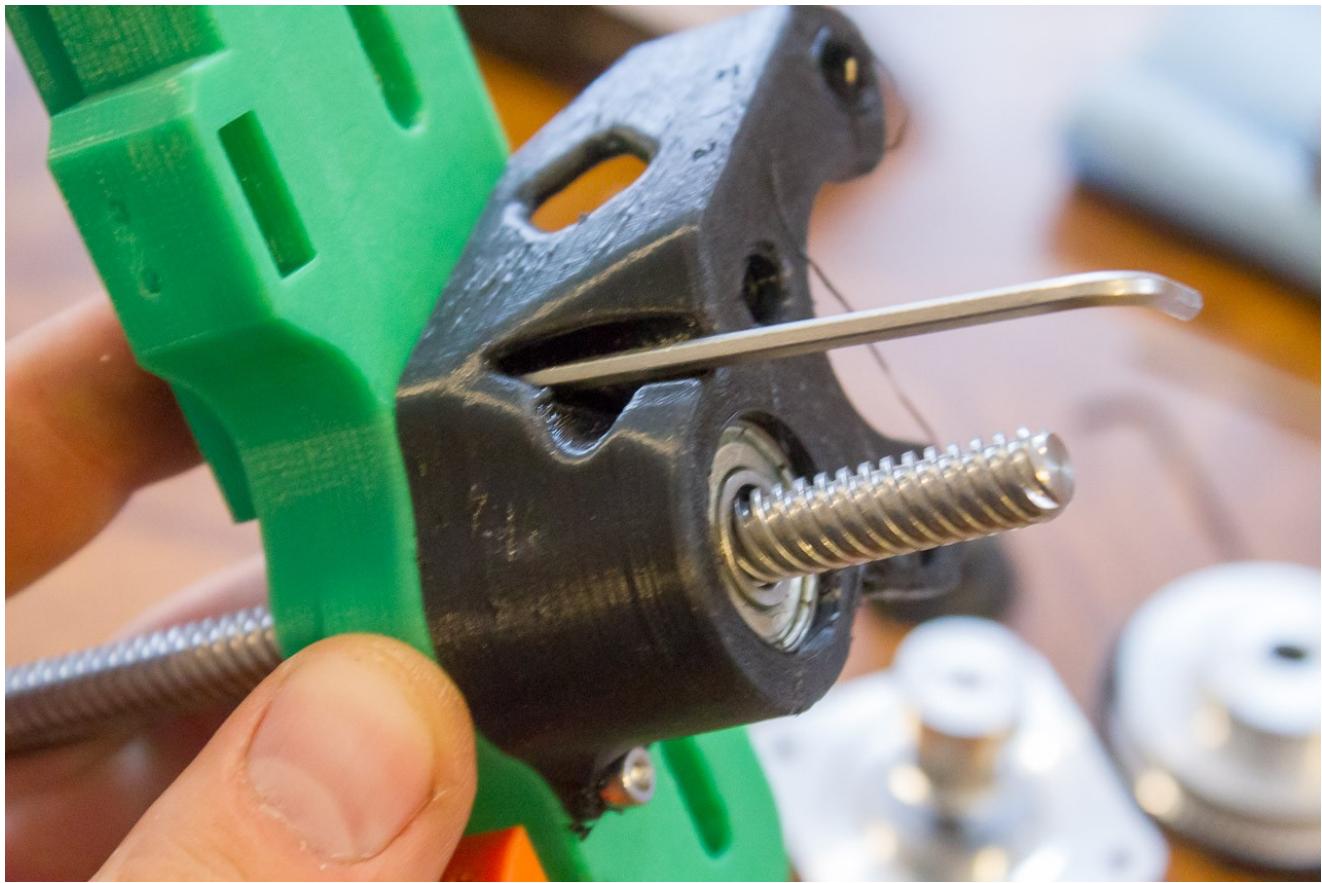


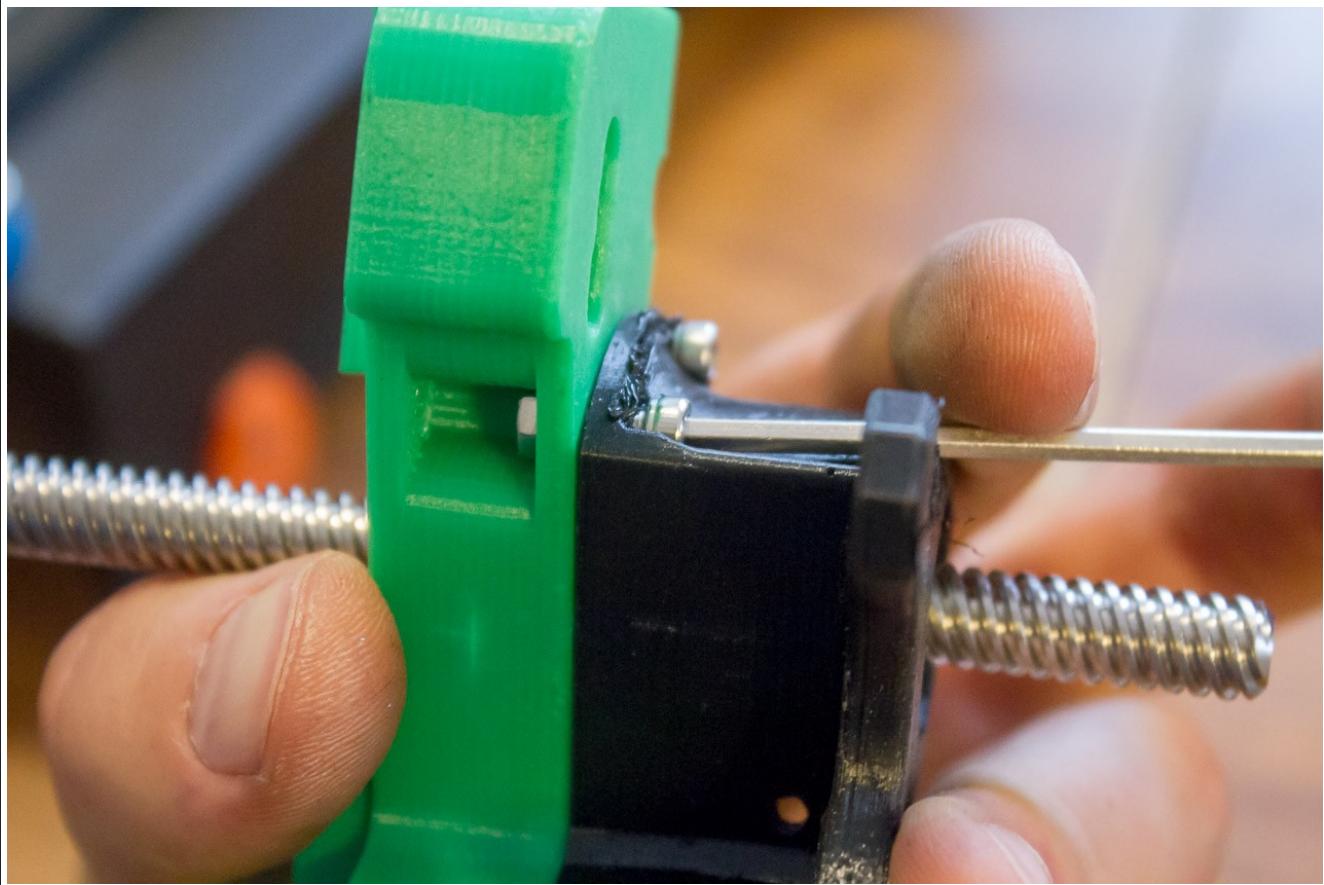
Once that's sorted, mount the vertical column on the base. Adjust the nut so that the screw comes close to, but doesn't touch, the left side vertical column. Once the nut is in the right spot, use some grub screws to tighten the nut on the threaded rod.



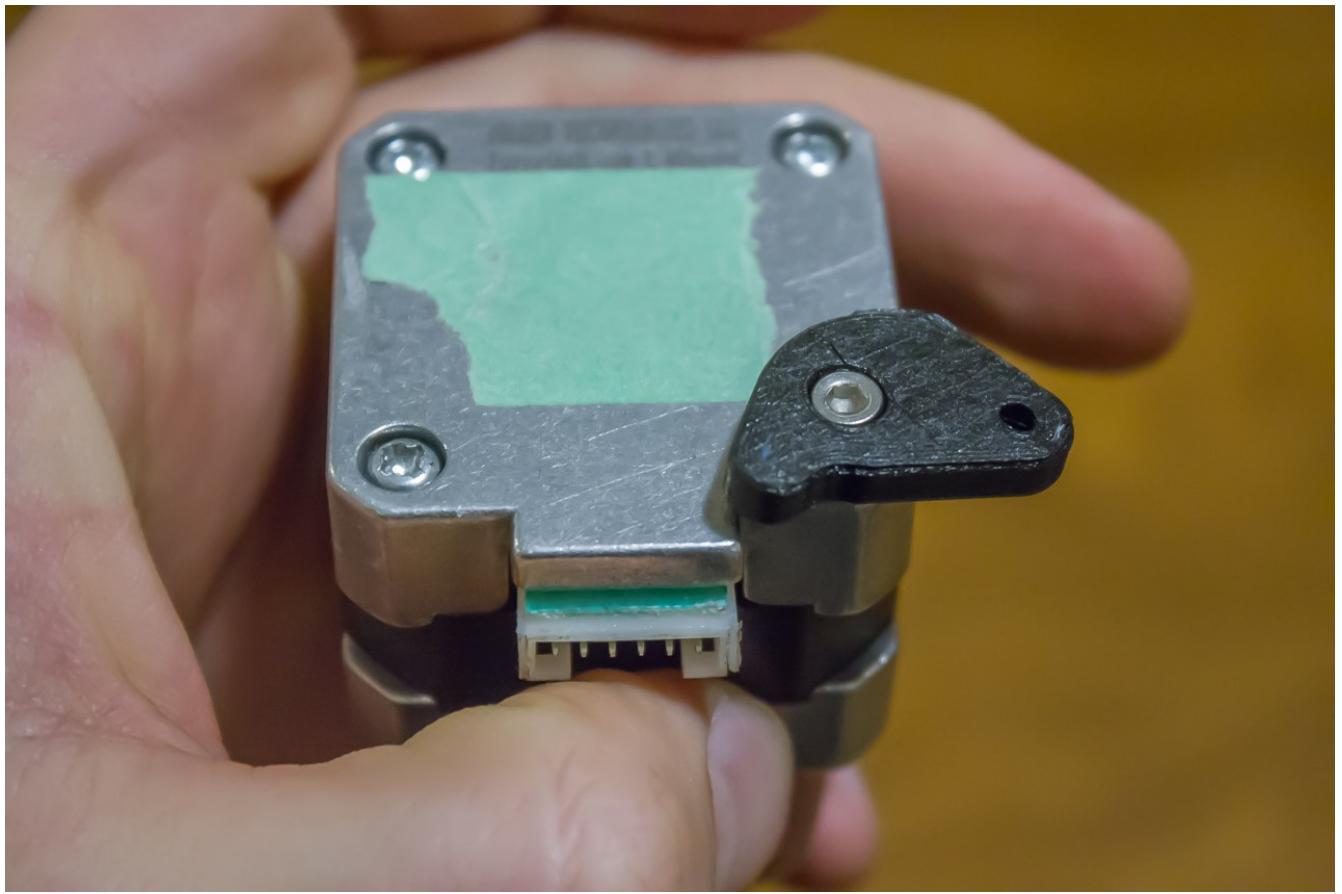
Fasten the bearing clamp onto the side mount using three 18mm M3 screws. The length is important!



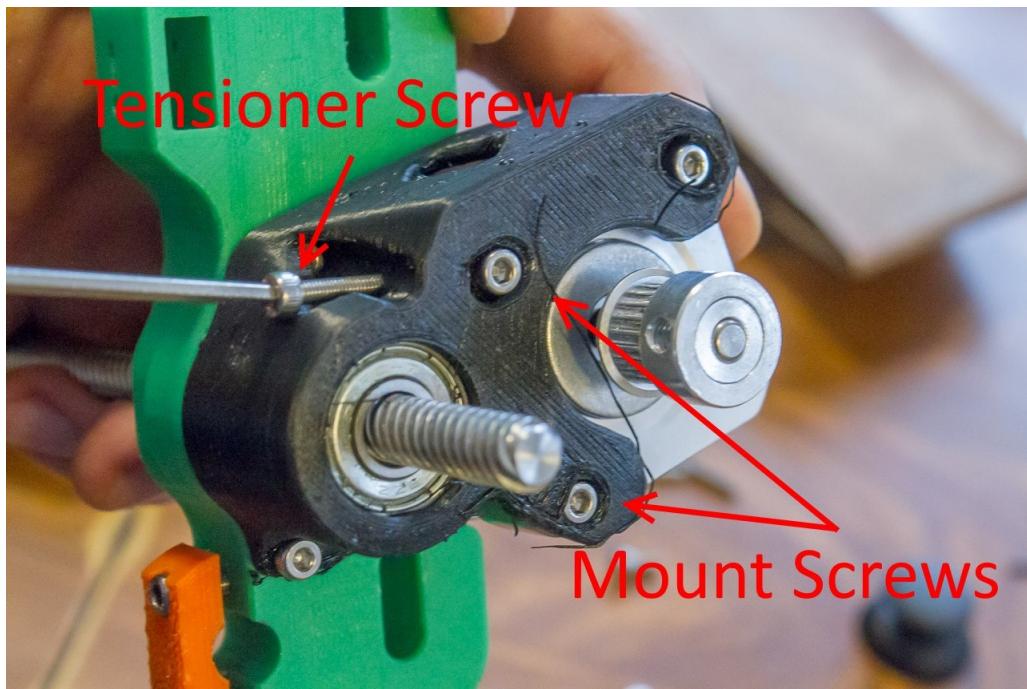




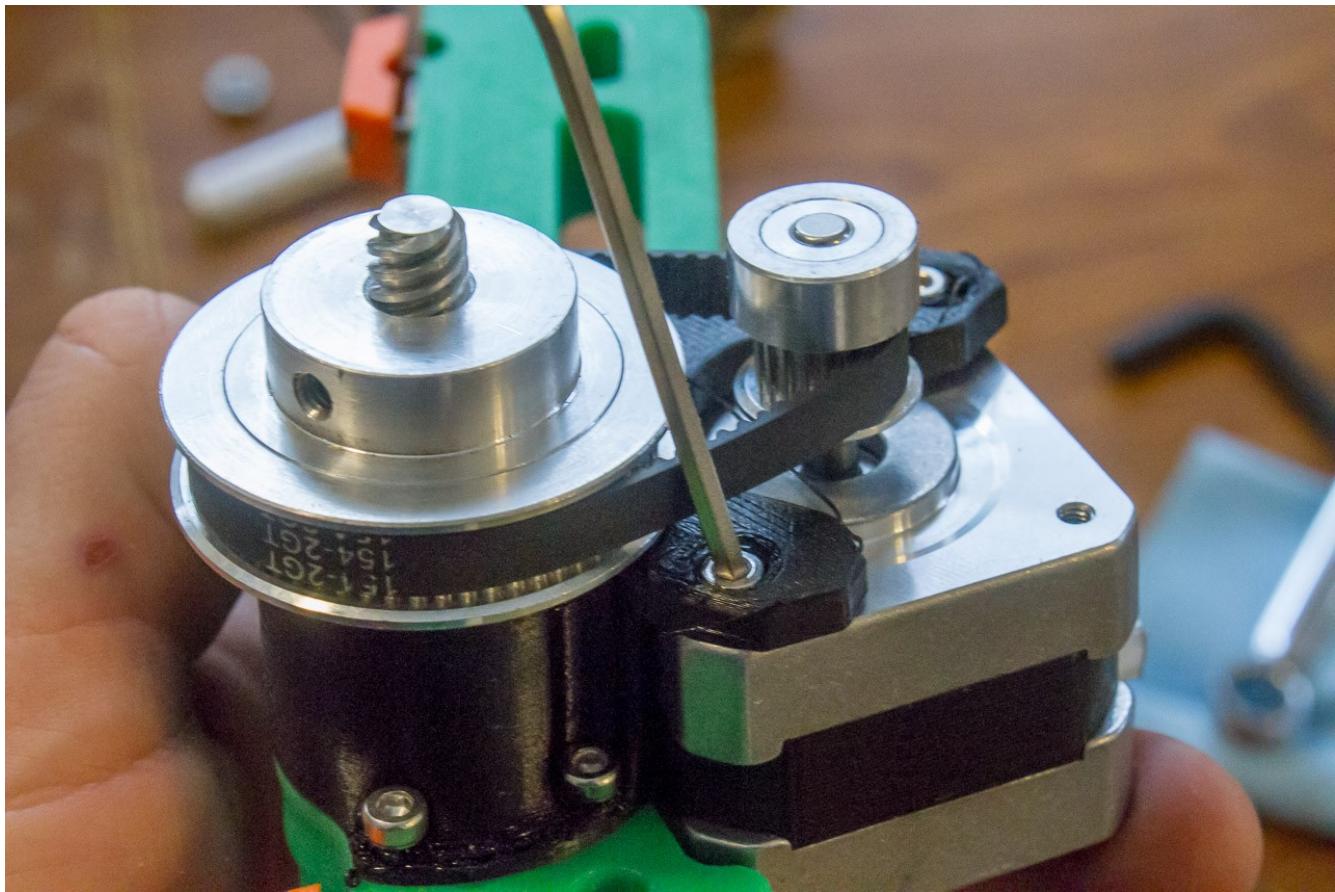
Cover the rear hole on the 34mm NEMA 17 motor with some tape. Remove one of the countersink screws and replace it with a socket-head screw. Put the small motor brace on the screw head as shown below.



Install the motor onto the bearing clamp. The motor brace uses one of the bearing clamp mounting screws you tightened earlier. You will need to loosen that screw first before sliding the motor into place, then tighten it again once the motor is positioned. Add the tensioner screw but don't put it in all the way yet.



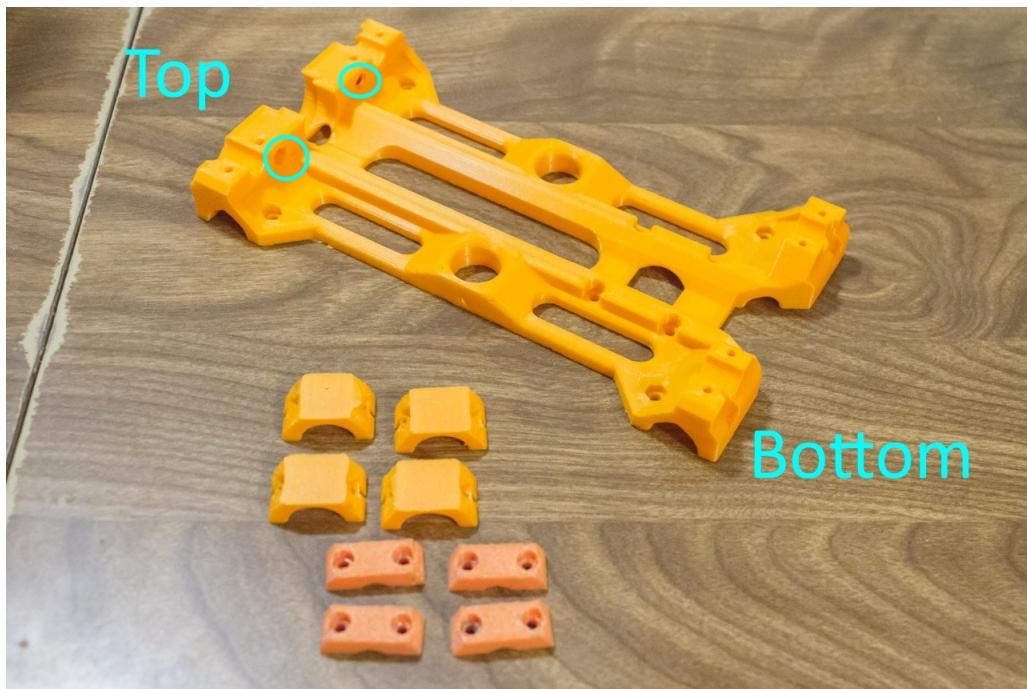
Install a 20-tooth GT2 pulley on the motor and a 60-tooth GT2 pulley on the lead screw. Put a 154mm belt on and tighten the motor using the tensioner screw.



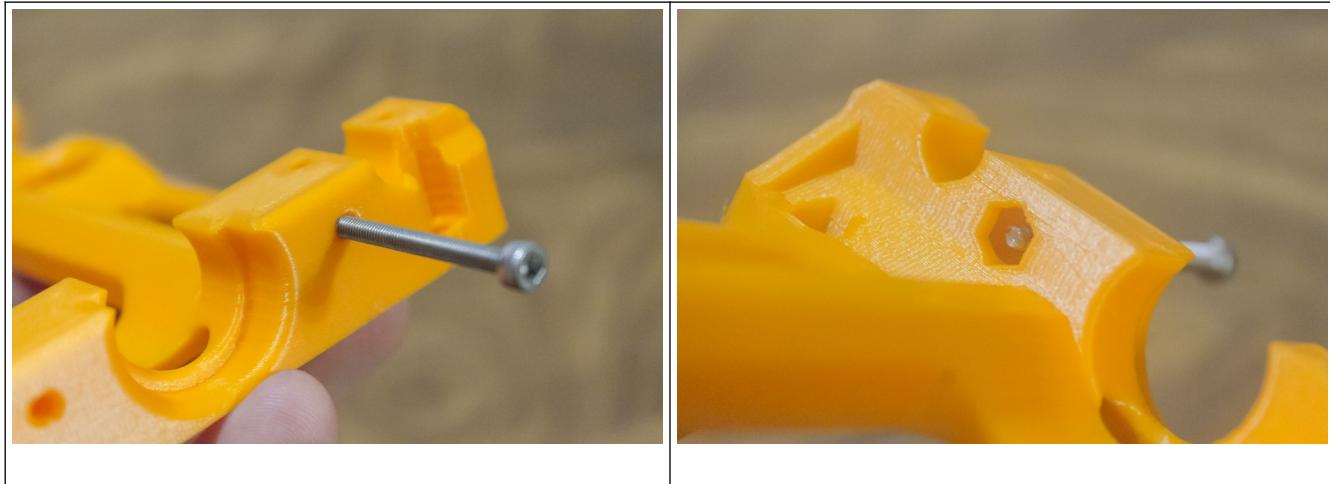
Gantry & Spindle Mount

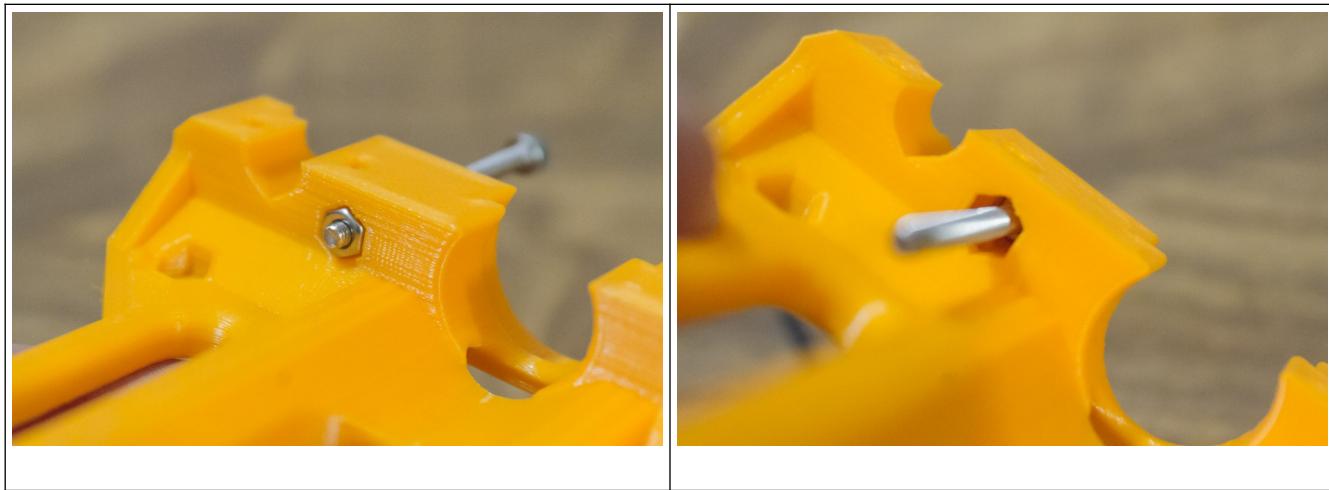
Note: Several parts shown in this section are different than the parts in other sections. I've been constantly improving the design, even while photographing the assembly. The Gantry is particularly different, and at the orange gantry was the more recent design.

Print off all the parts for the gantry and spindle mount: the Gantry, Z-Axis Motor Mount, X-Axis Nut Mount the Z-Axis Carriage, 4x Rail Clamps, and 4x Y-Axis Bearing Clamps. Here's a picture of the gantry, bearing clamps, and rail clamps.



Start by inserting an M3 nut in each of the circled holes. Use a screw through the top to pull the nut in, and tighten it so the nut is secured inside the hole. Then remove the screw. This will be important for later.



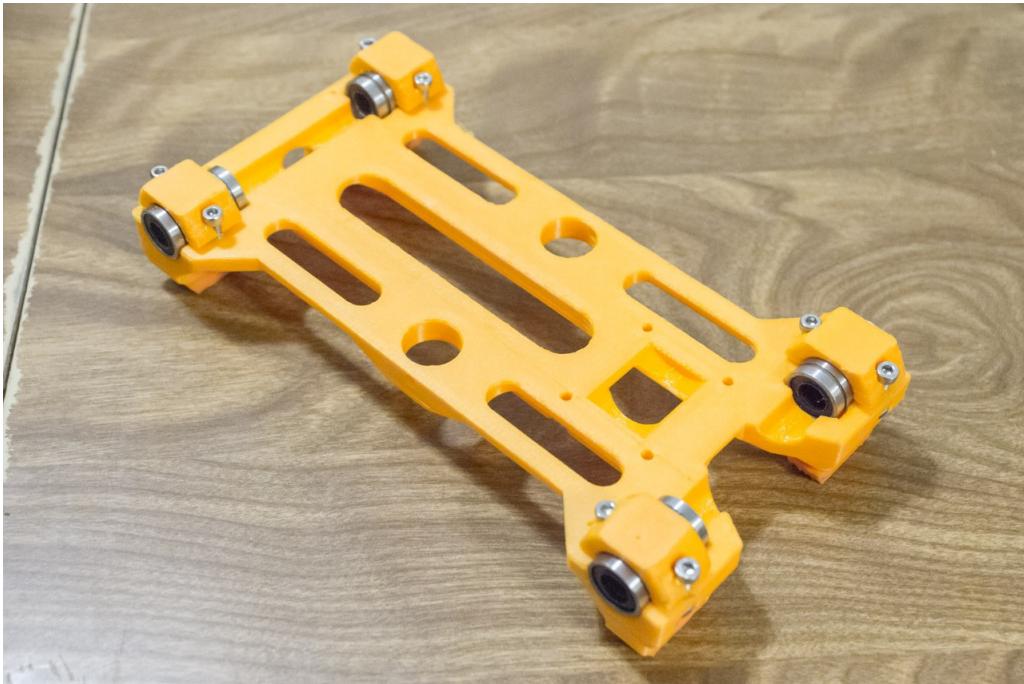


Now screw on the rail clamps. Each M3 screw gets a nut.

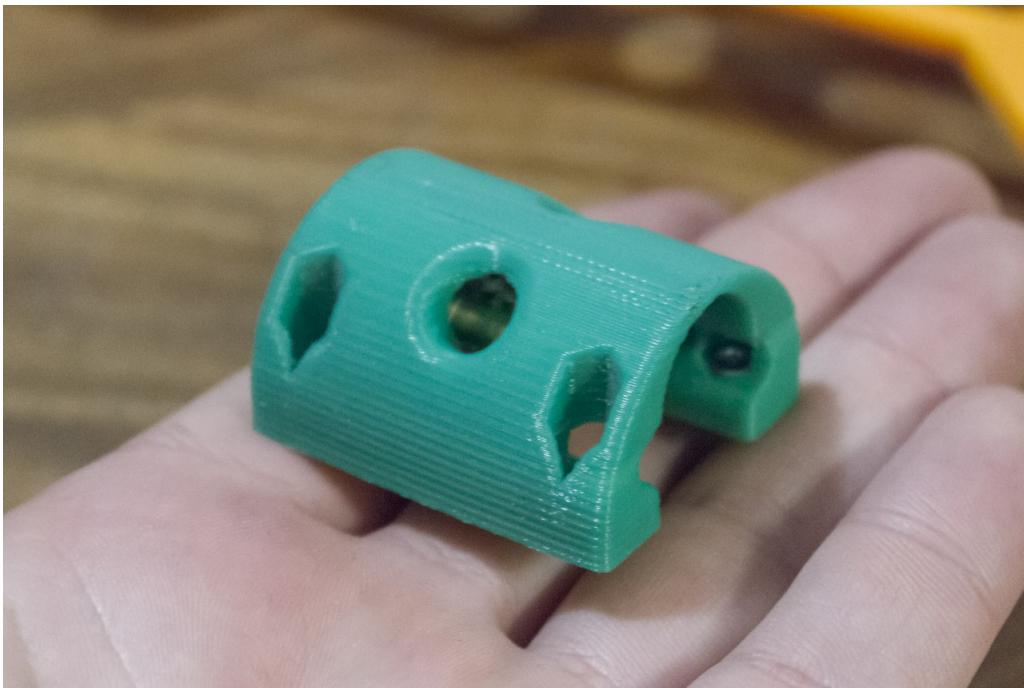


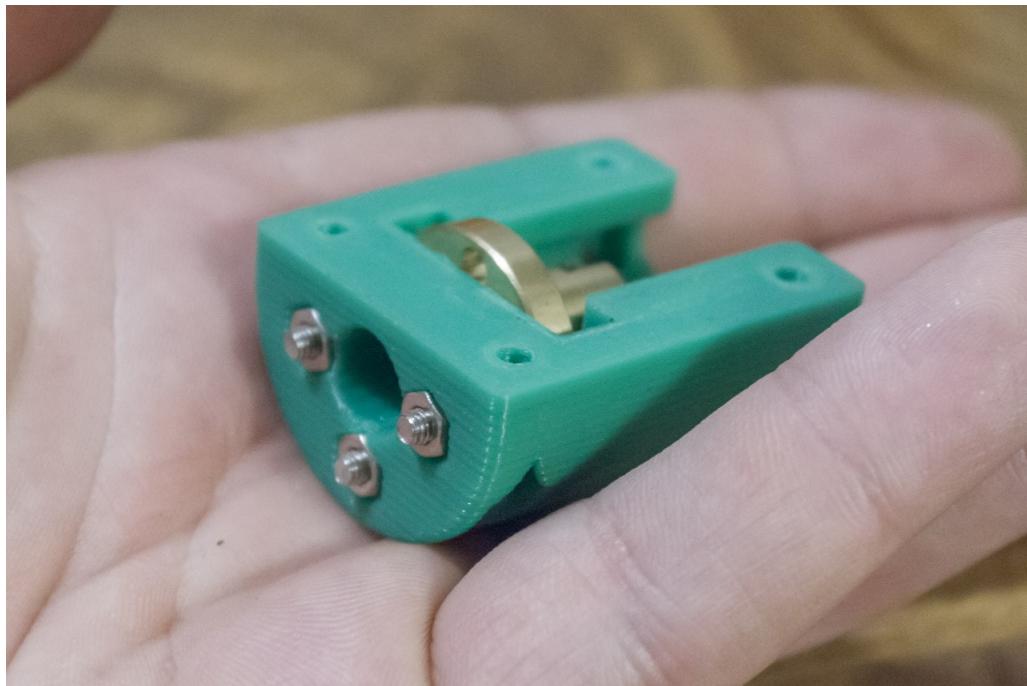
Install all four LM8UU linear bearings.



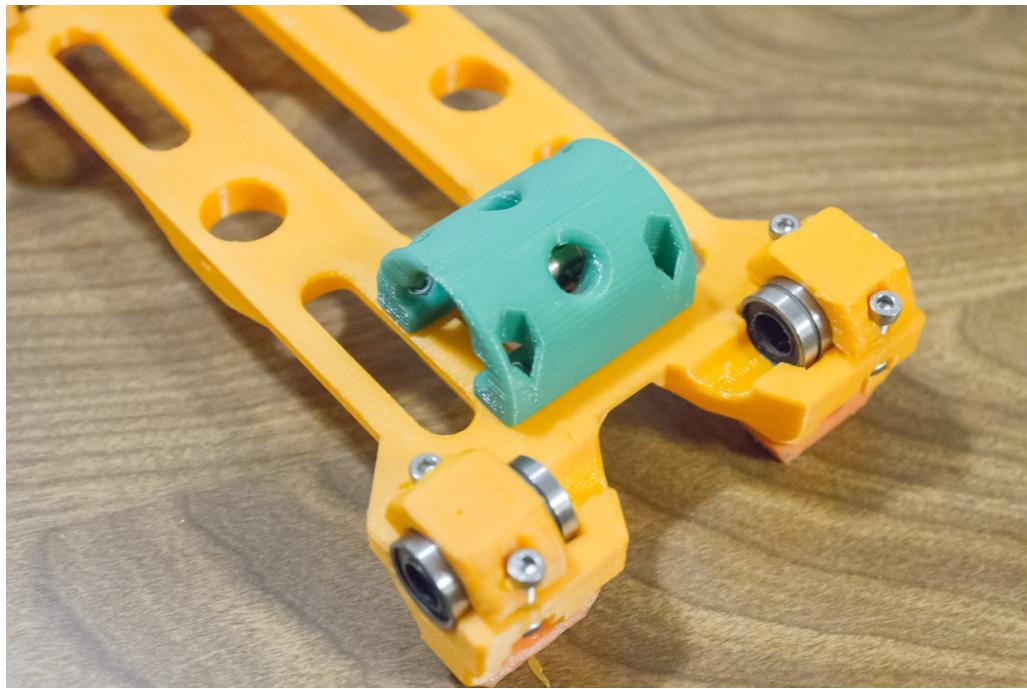


Take the X-axis nut mount and install the Acme lead screw nut inside it.



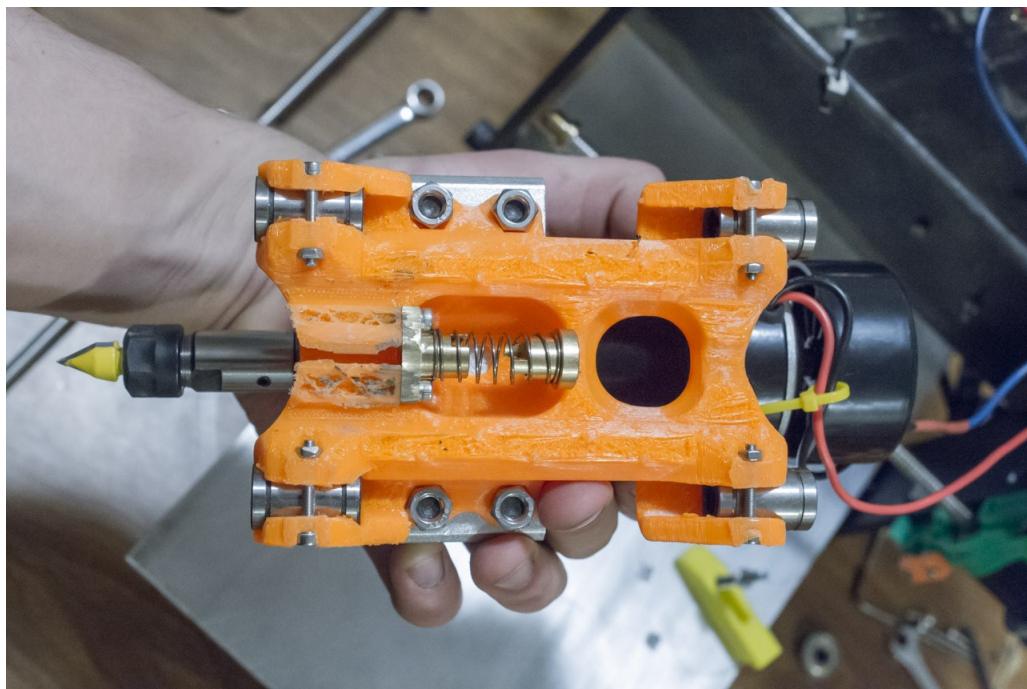


Screw the nut mount onto the rear side of the gantry. When I installed the spring-loaded anti-backlash nut, it turned out to be easier if the nut mount was installed the other way around. So please note that it is installed backward in this photo (the open end should be to the right).

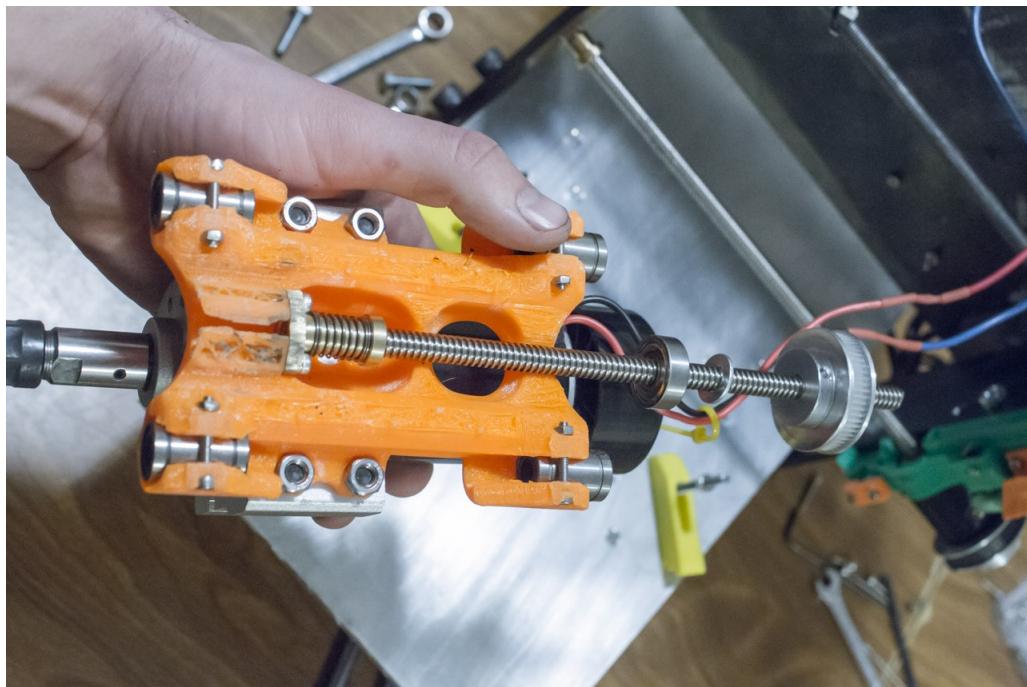


Here's a photo of the spindle mount. I've already assembled this one several months ago, so there are no photos of the assembly process. You'll note that this one is in rough shape, I had to change the design a bit to make everything fit. The available model will work just fine without the mods.

Here's the assembly process. First, trim all the little bits on the bottom that are only used to make printing easier. Then slide four LM8UU bearings in and secure them with screws. Next install the Acme lead screw nut. Finally, mount the spindle on with M6 screws.



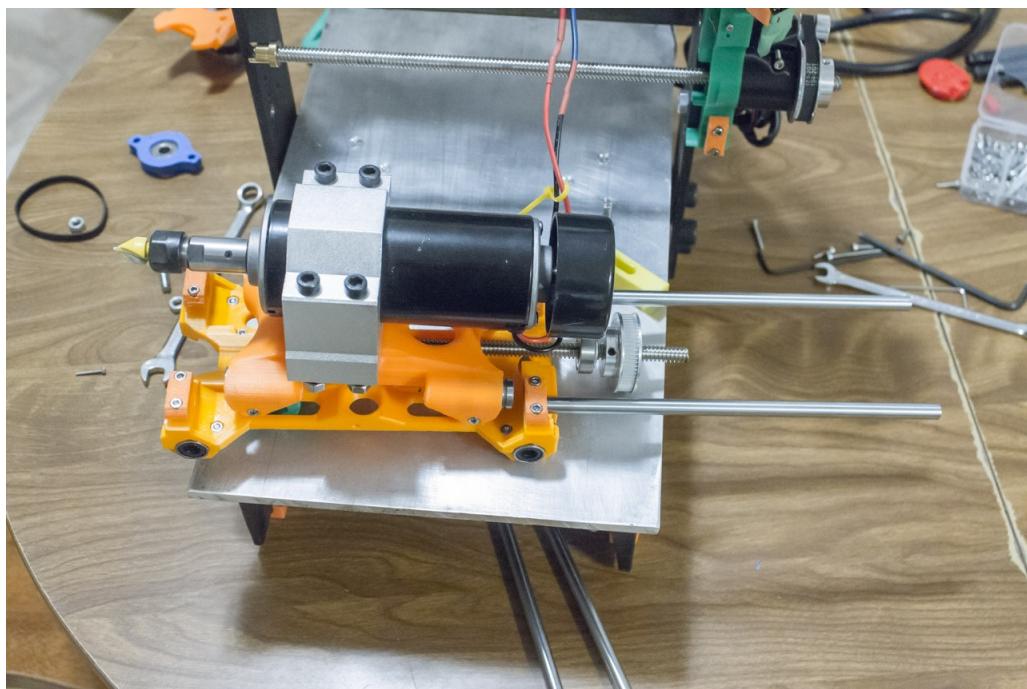
Screw the Acme lead screw into the nut. I already had the 60-tooth GT2 pulley, 608VV bearing, and spacers on the lead screw here. You can put them on later.



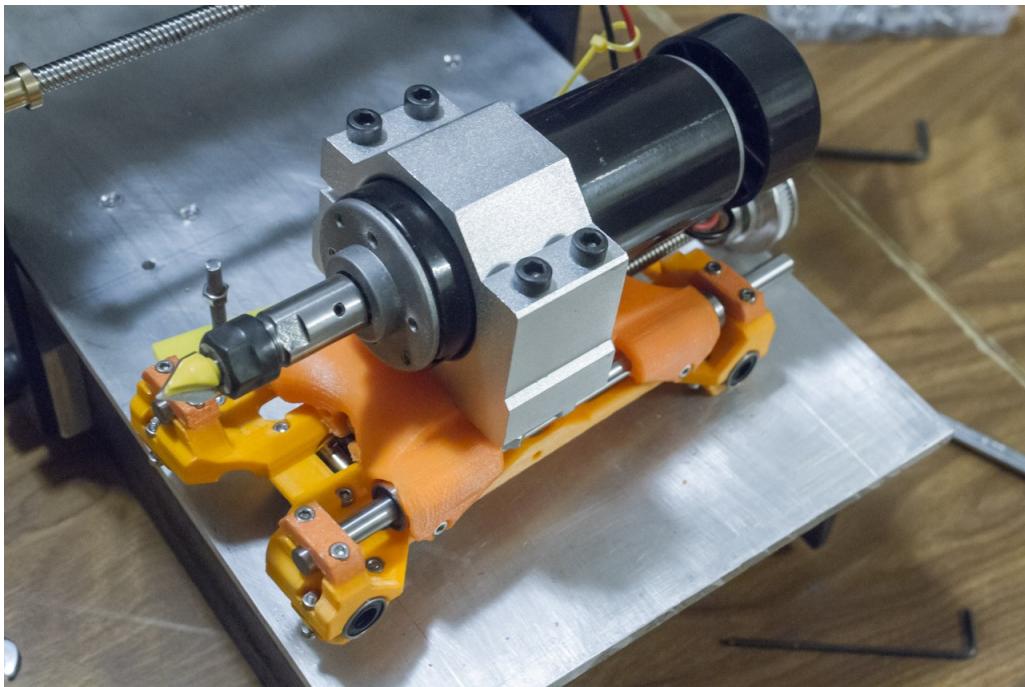
Ok, back to the gantry. Get both Z-axis rails started in the top of the gantry.



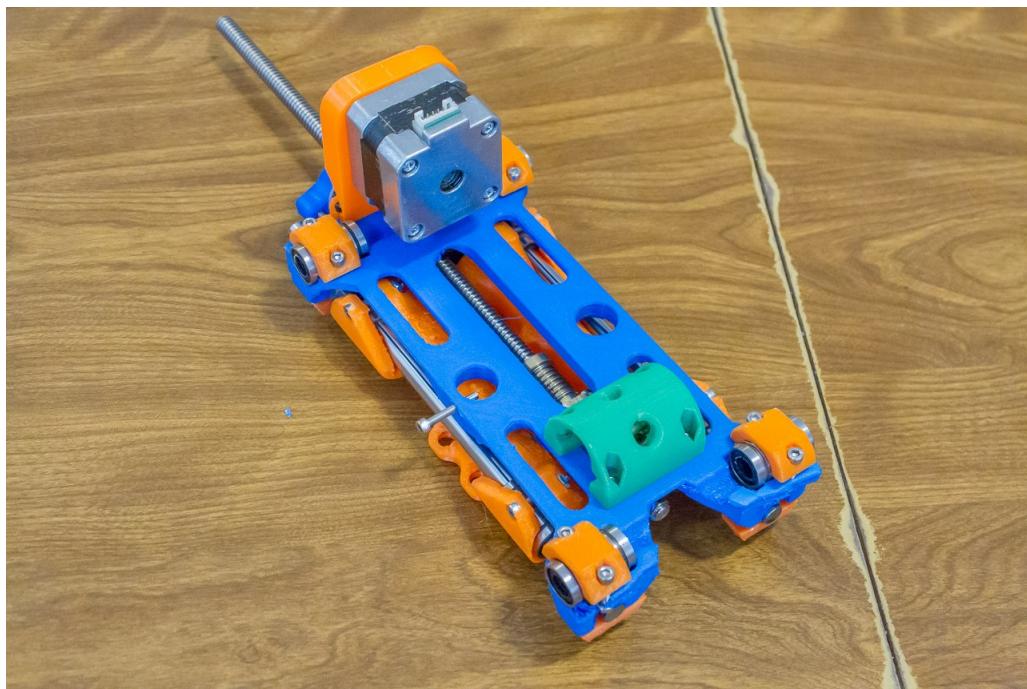
Set the spindle mount on top of the gantry.



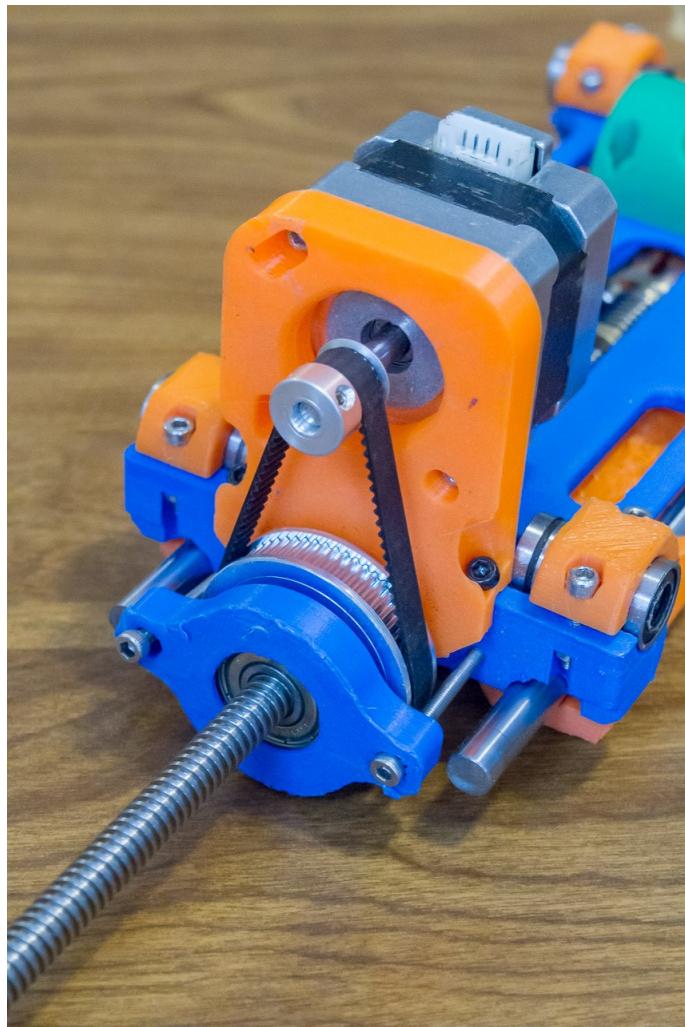
Push the rails down through the linear bearings and into the bottom clamps. Push the 608VV bearing down into the gantry.



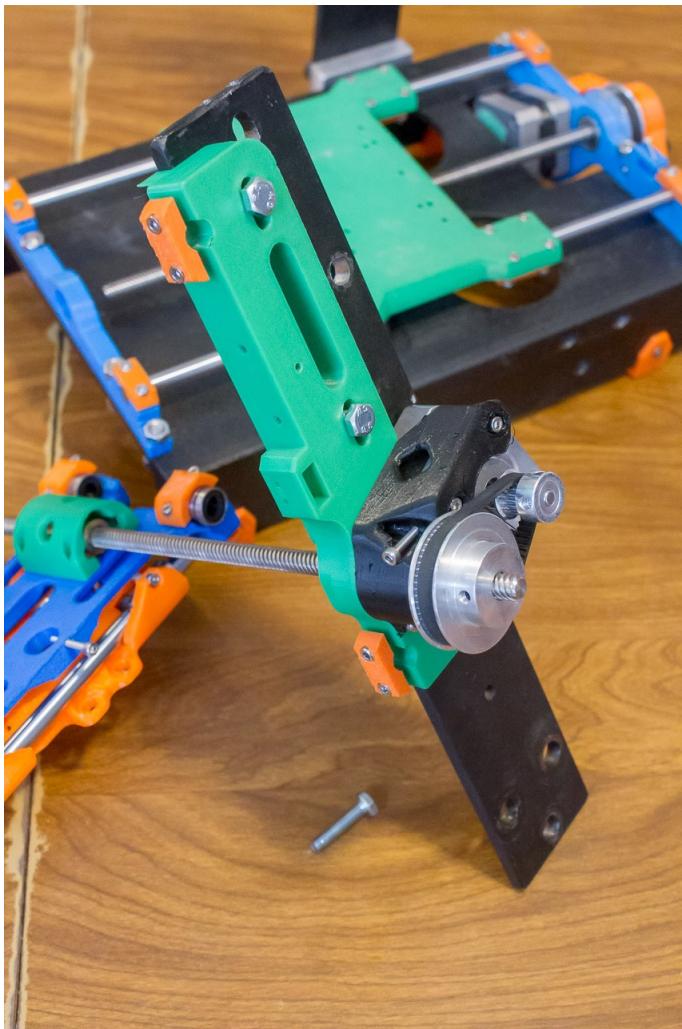
Install the motor mount on the top of the gantry. Install the motor on the motor mount. Slide a 16-tooth GT2 pulley on the motor and a 60-tooth GT2 pulley on the lead screw (if it's not already there) and put the 180mm belt on.



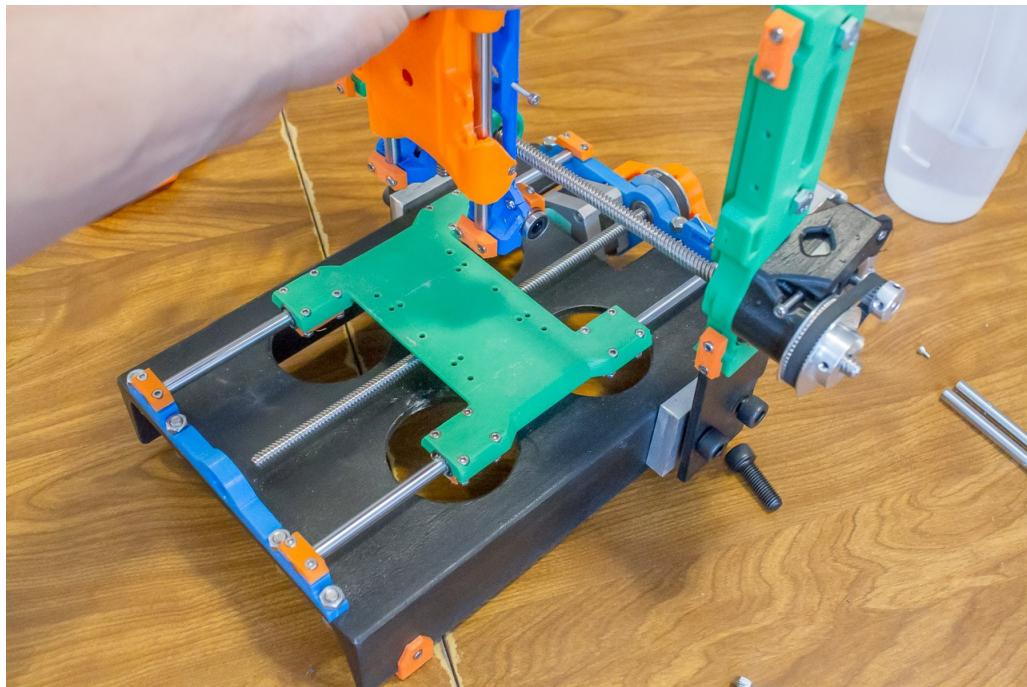
Insert a 608VV bearing in the Z-axis bearing clamp. Slide the bearing and clamp down over the Z-axis lead screw. Using long screws, secure the bearing clamp down using the nuts that you inserted in the gantry at the start.



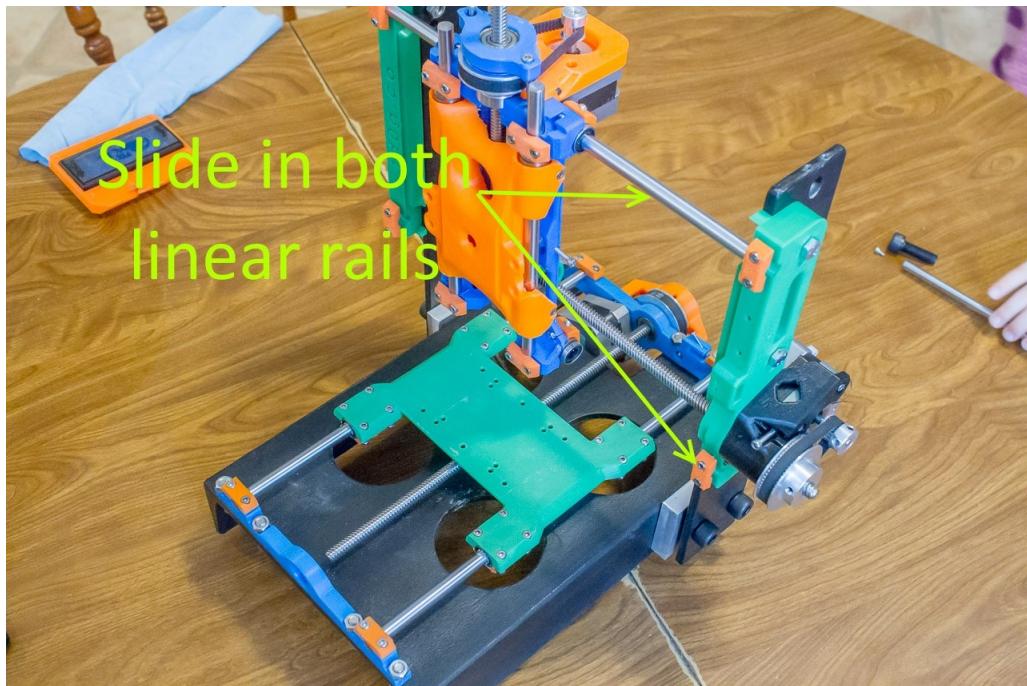
Time to put the gantry onto the mill. Screw the X-axis lead screw into the gantry's lead screw nut.



Install the right vertical column onto the base.

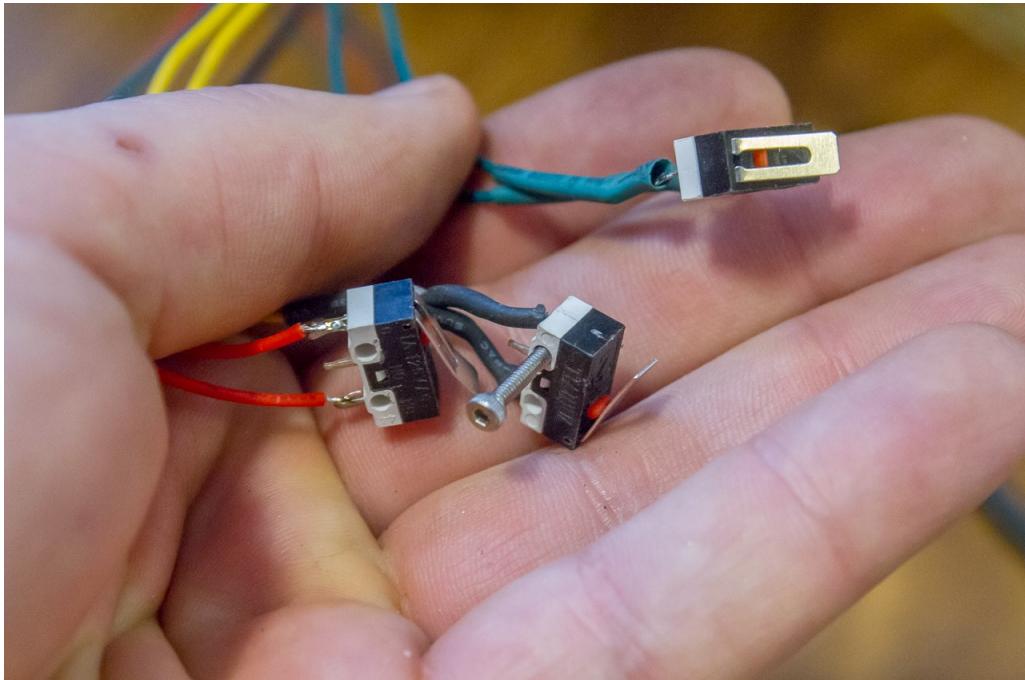


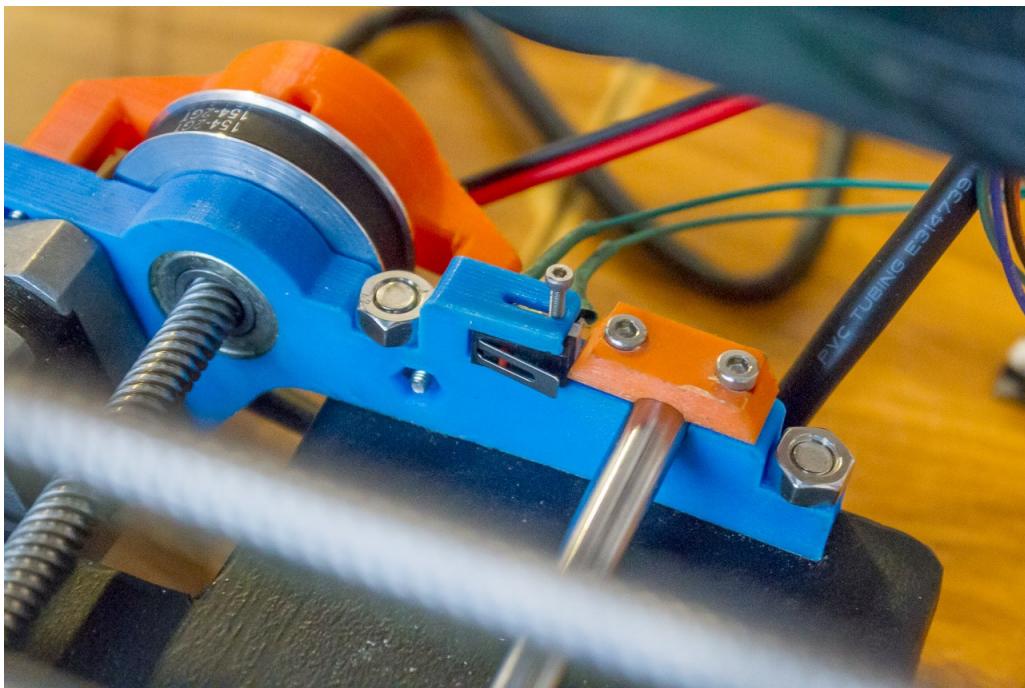
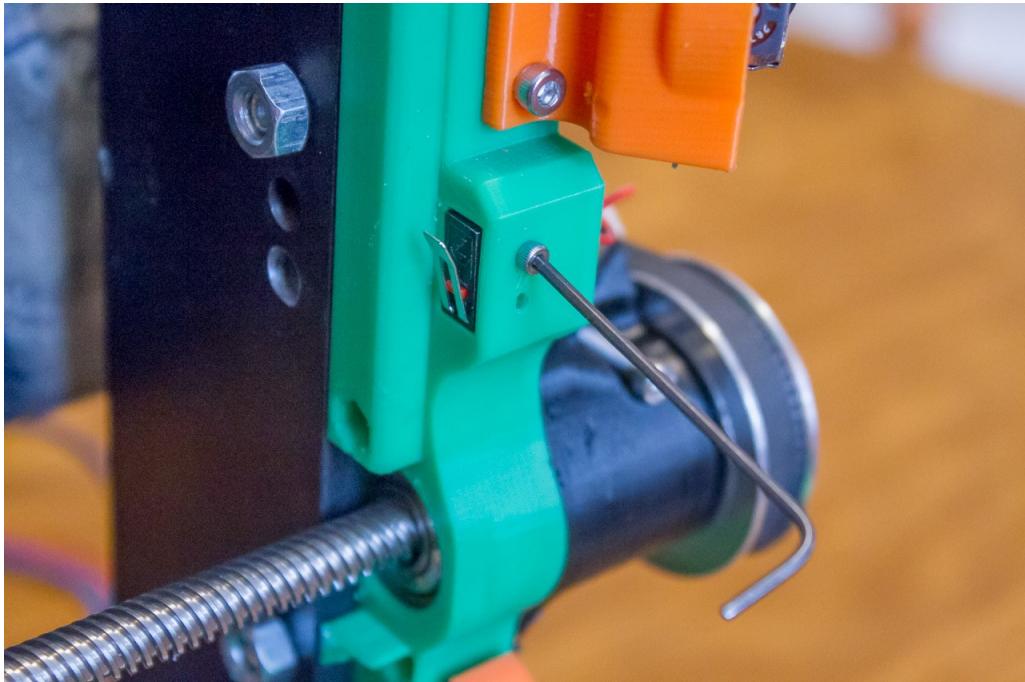
Slide the X-axis linear rails through the rail mounts and the linear bearings.



Endstops

At least three microswitches are used for the endstops. At the time of writing, there were only install points for one endstop on each axis (Max X, Min Y, and Max Z). I intend to add install points for the other endstop on each axis, so each will have both min and max endstops. Use M2 screws to secure the endstops on the rail mounts as shown below.

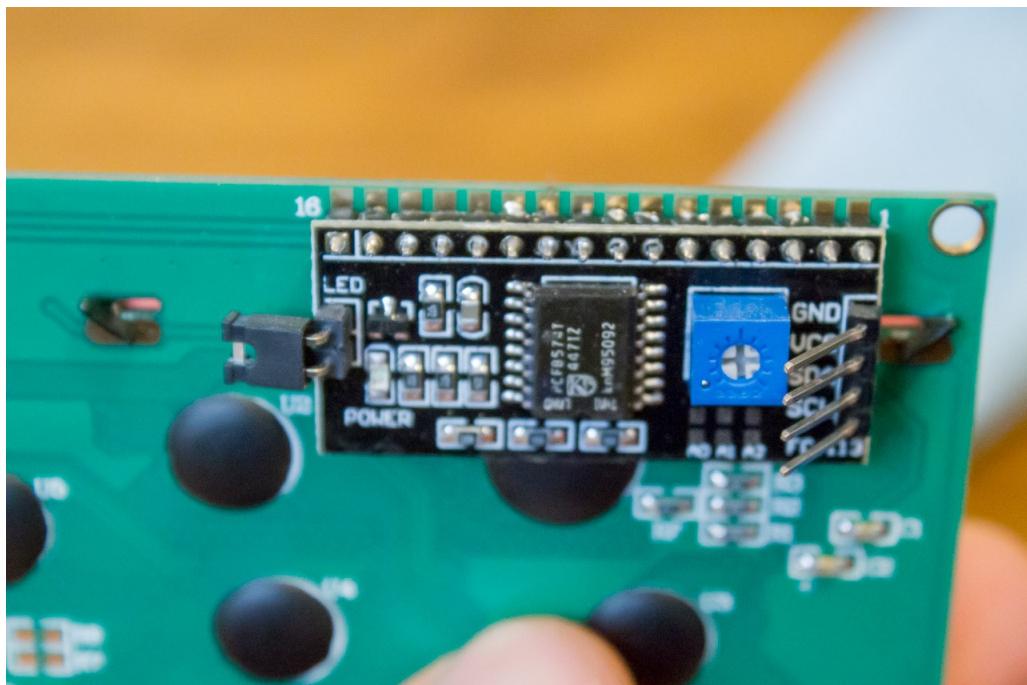




Display & SD Reader (OPTIONAL)

So now here's some of the optional peripherals. I made a mount for the 2004 LCD display on top of the right vertical column. There's also an SD card reader mount that fits on the back of the LCD mount.

I used an I₂C LCD module to reduce the number of data pins needed to 2. It was hard to find a 2004 LCD with the I₂C module, so I stole one from a 1602 LCD and installed it on the 2004 LCD.



Print the LCD Mount. Install the display inside the mount.



The mount has two 3mm holes. Position it on the top of the 1/4" vertical column and use the holes as a template to drill and tap the M3-threaded holes. I already did this a while ago, so no photos of the process.





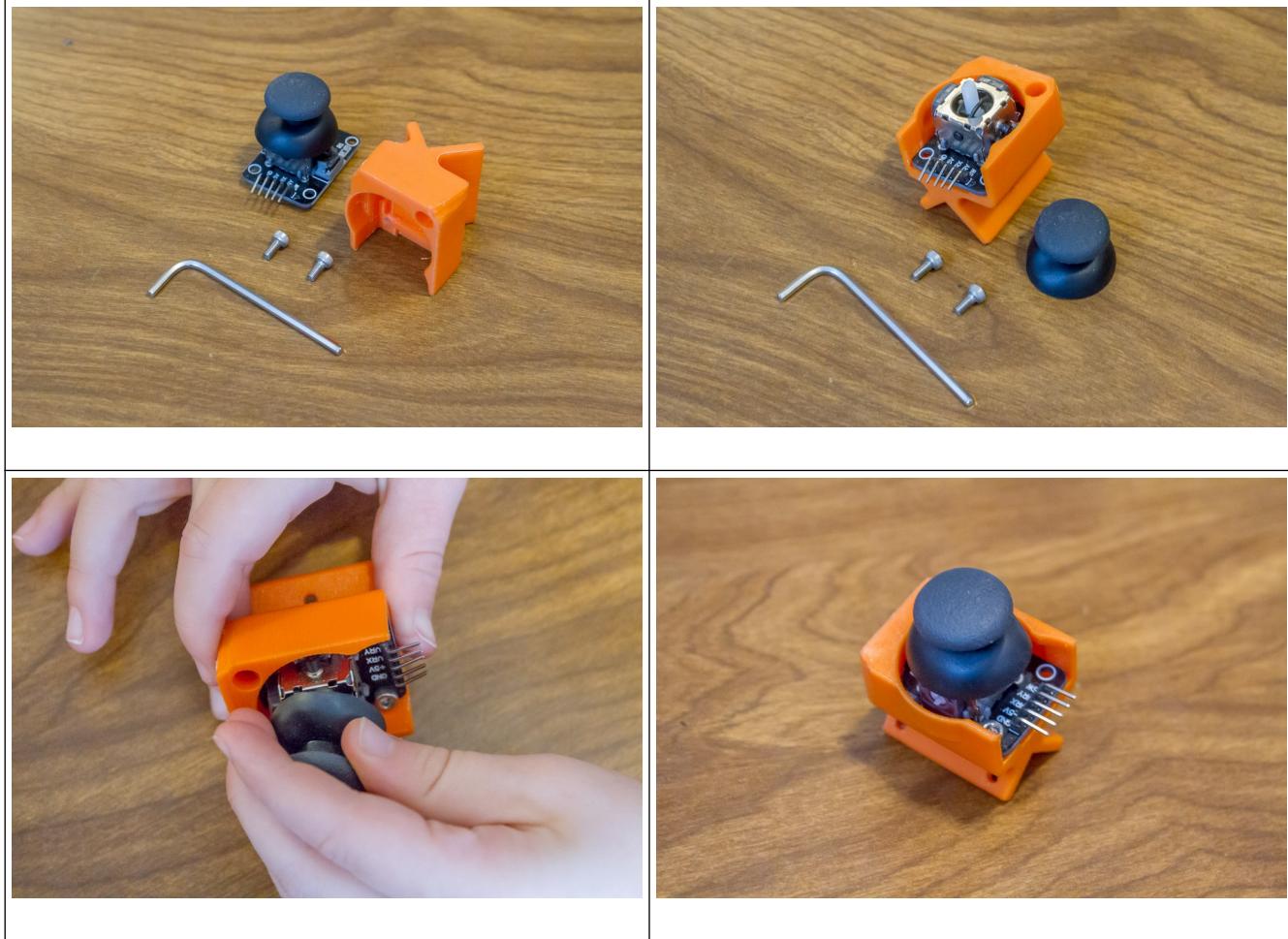
The SD reader mount just shares two of the LCD mount screws. Pretty easy to install.

NOTE: Picture of SD reader

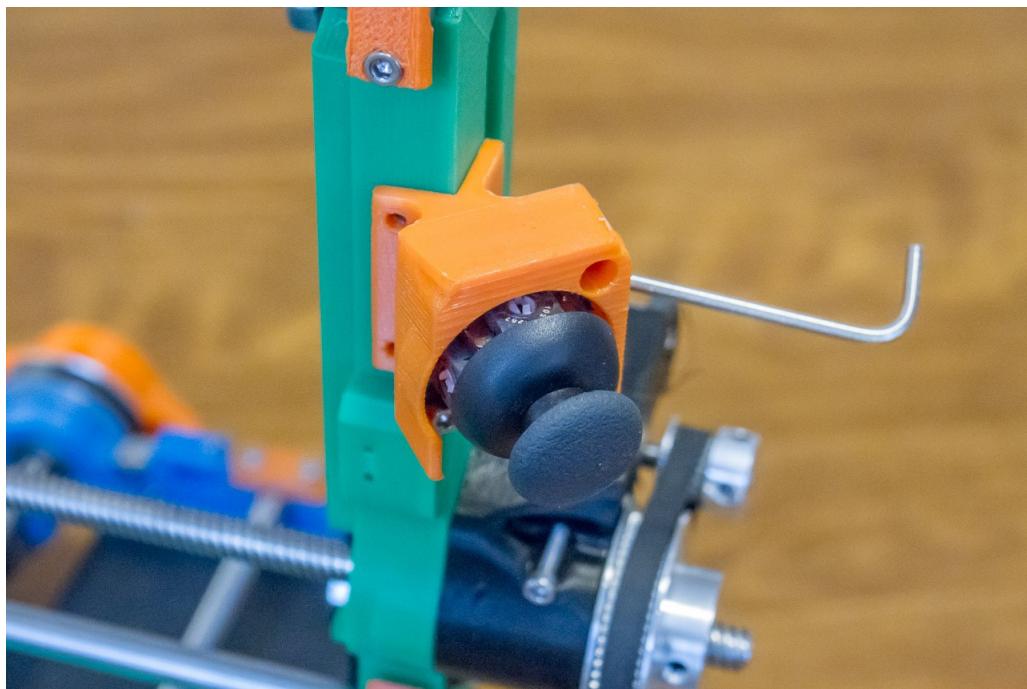
Control Joystick (OPTIONAL)

Having very little experience with CNC machining, I wasn't sure how user controls should work. Looking back, a keypad would probably have been better than an analog joystick... But here we are, the joystick is sufficient, and I haven't found the time to overhaul the user control system.

Print the Joystick Mount. Install the joystick using a pair of short M3 screws.

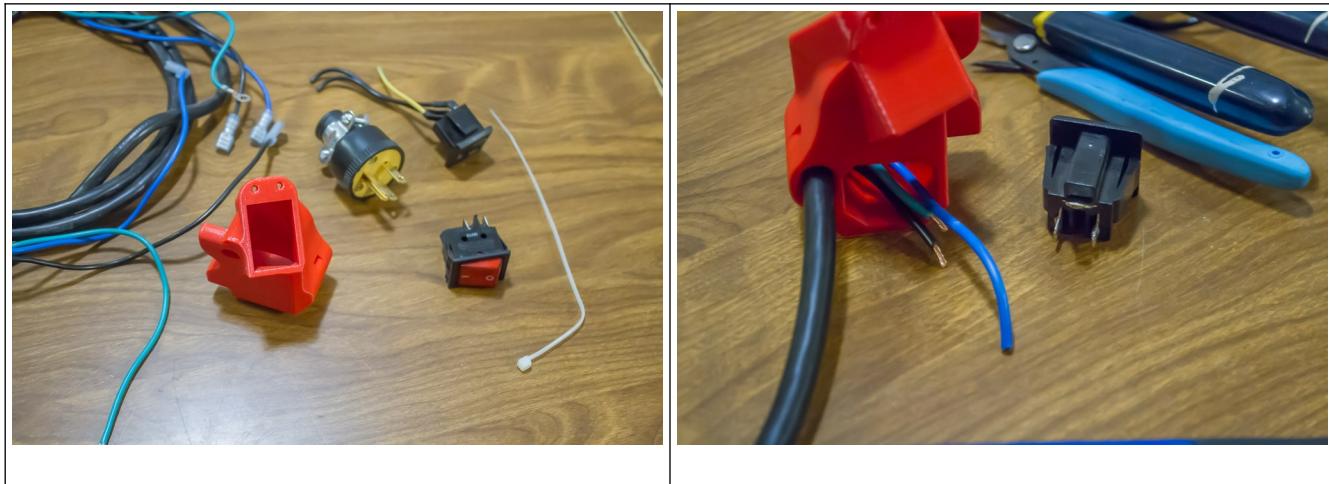


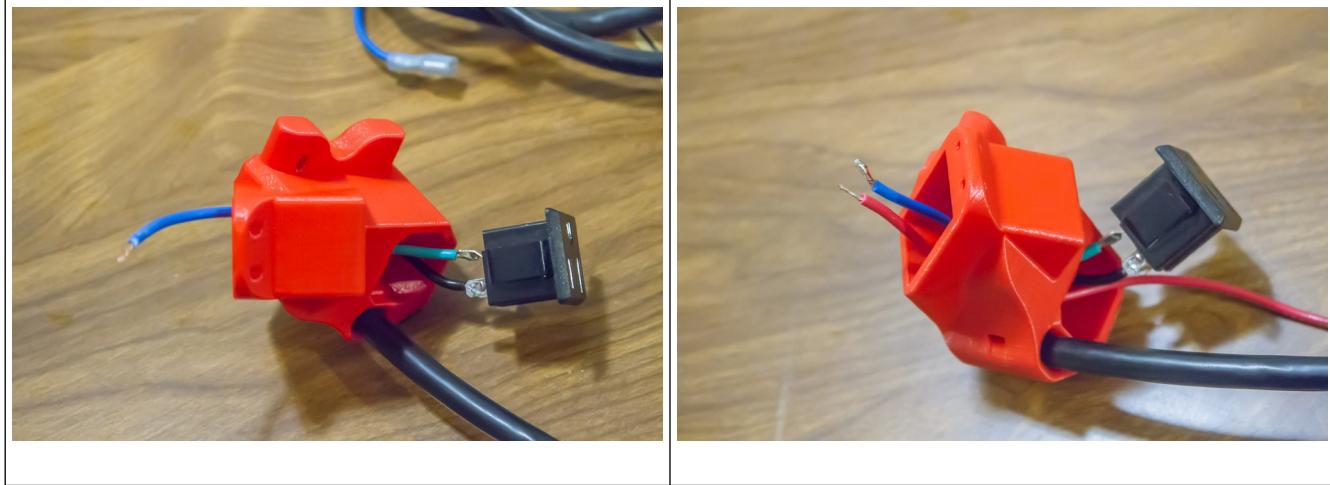
Install the joystick on the right X-axis rail mount using three short M3 screws.



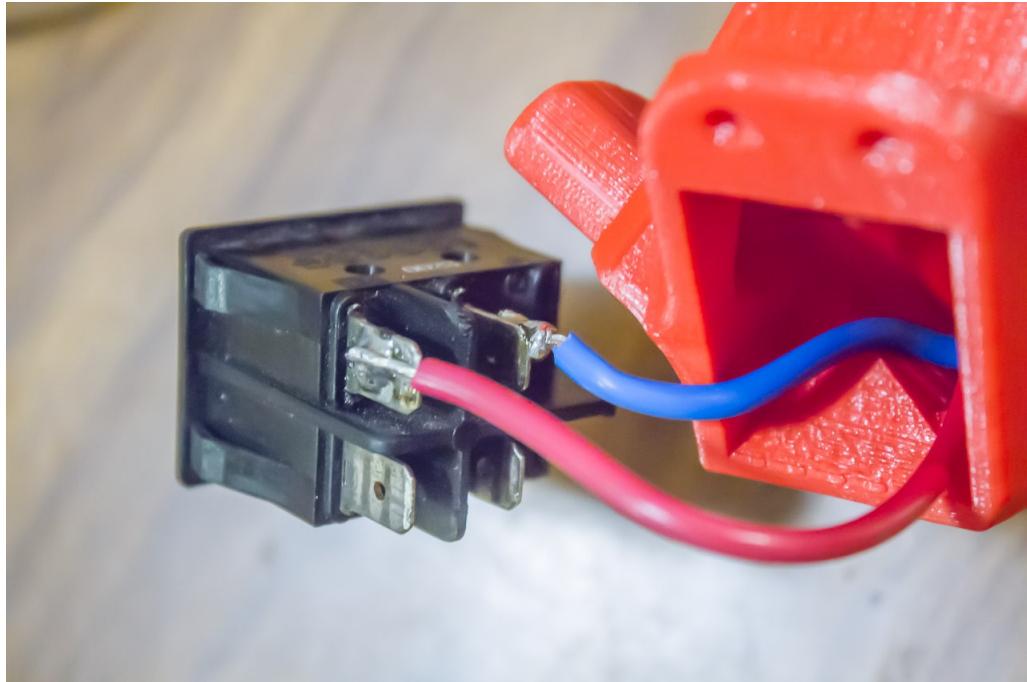
Power Distribution & E-Stop (OPTIONAL)

You can manage the power however you wish, but here's a recommended solution. Print the Power Box and E-Stop Button. Get the power rocker switch and outlet ready, as well as a section of extension cord. Thread the power cord into the distribution box and solder the neutral and ground onto the outlet.





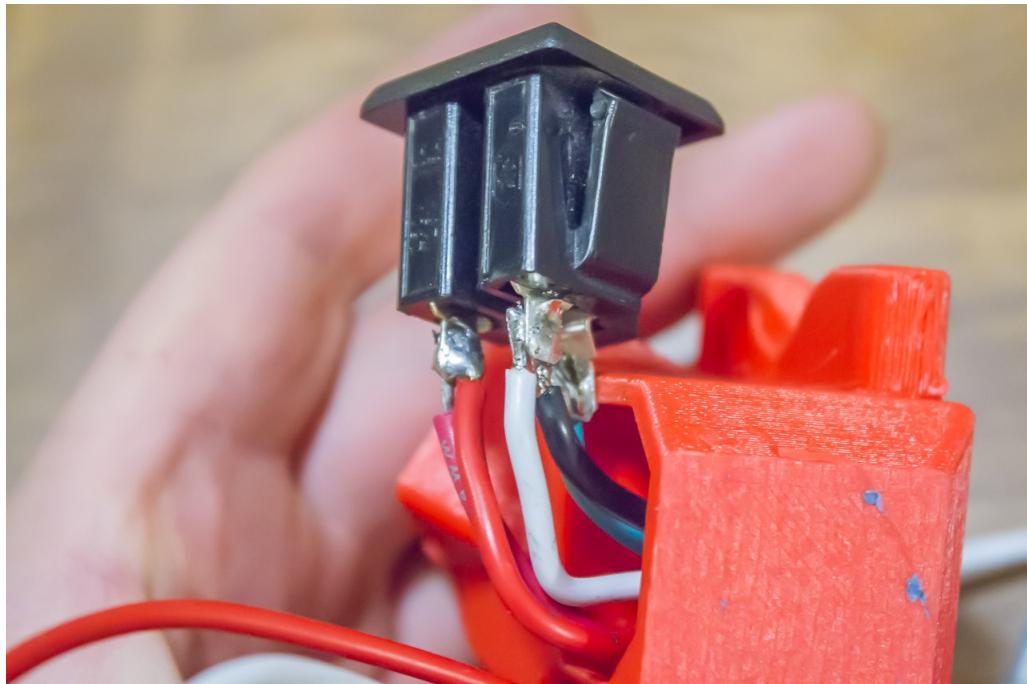
Solder the positive from the power cord to the switch, then solder a wire to the other side of the switch.



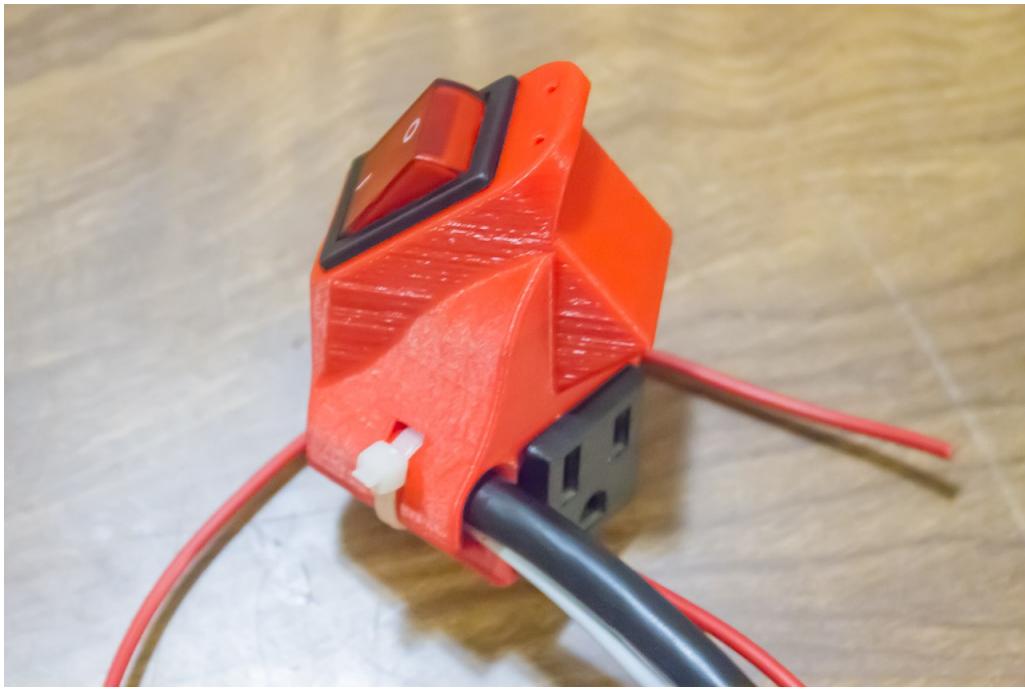
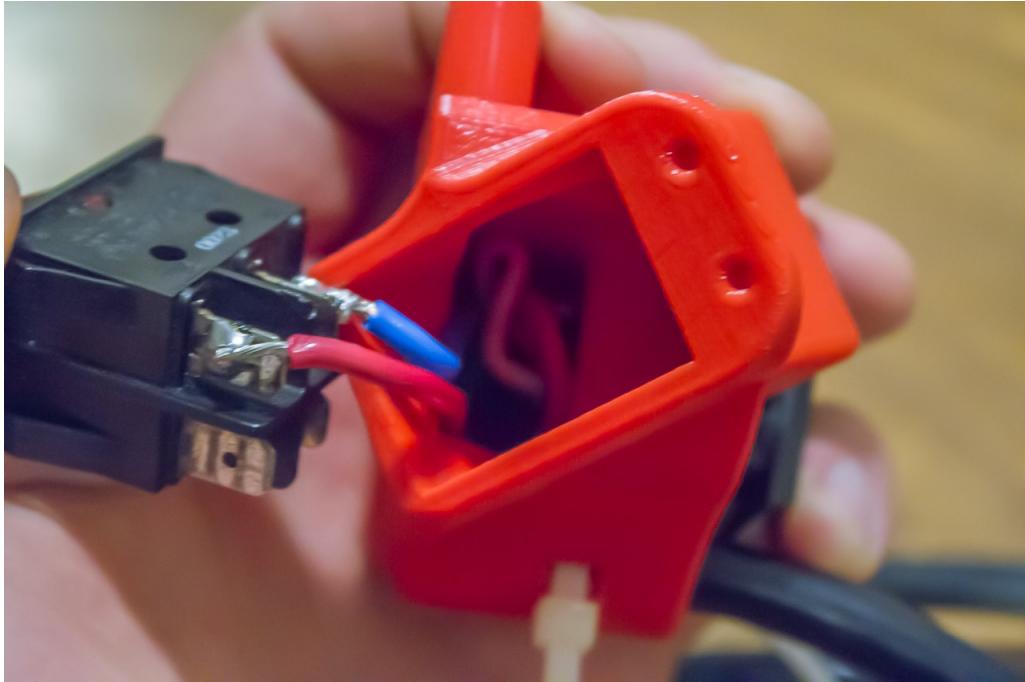
Insert the power switch into the box. Now get a pair of 16-gauge wires long enough to reach from the right vertical column to the spindle power supply. Run those wires into the box next to the input power cable. Use a zip-tie on the side to secure all the wires.



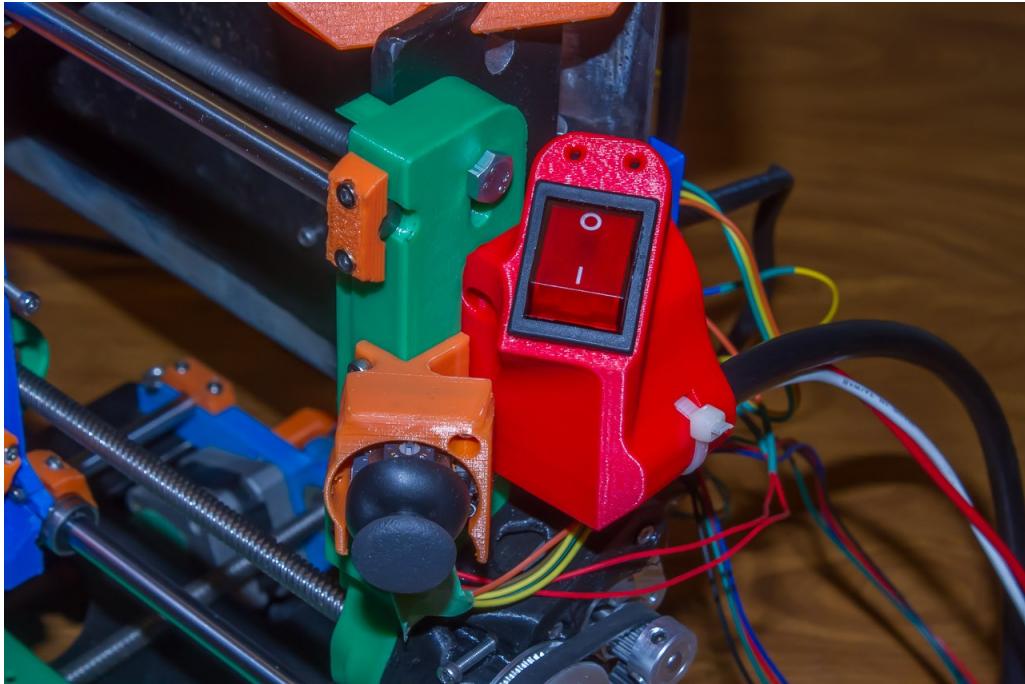
Solder the spindle power wires to the outlet. Solder the switched power wire to the power side of the outlet.



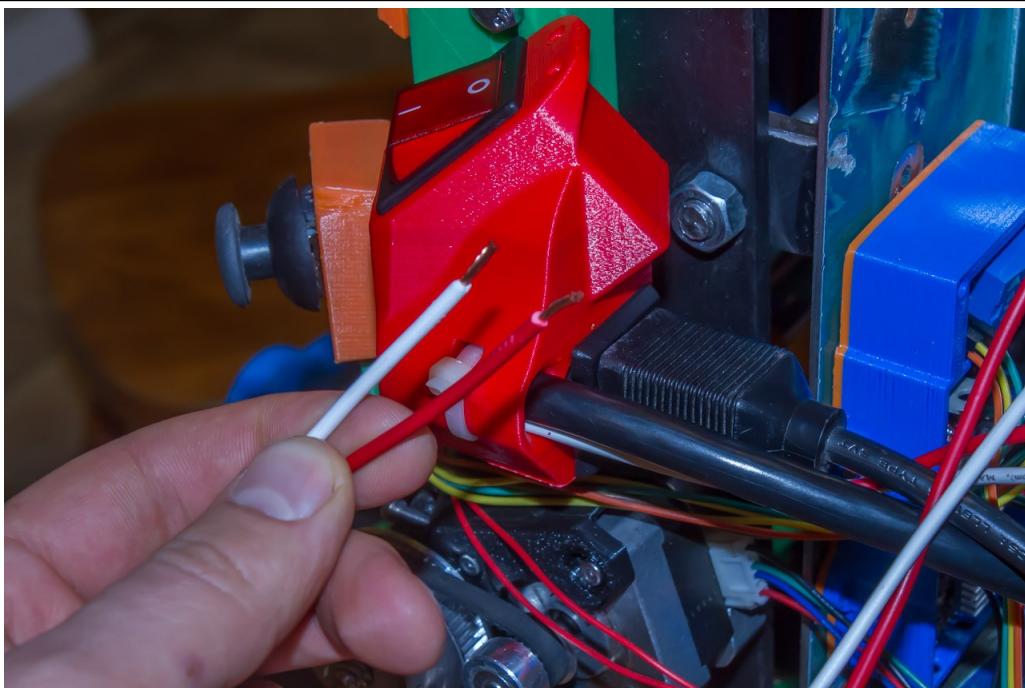
Push the outlet into the hole. I had to take the switch out to adjust the wires before the outlet would go in.

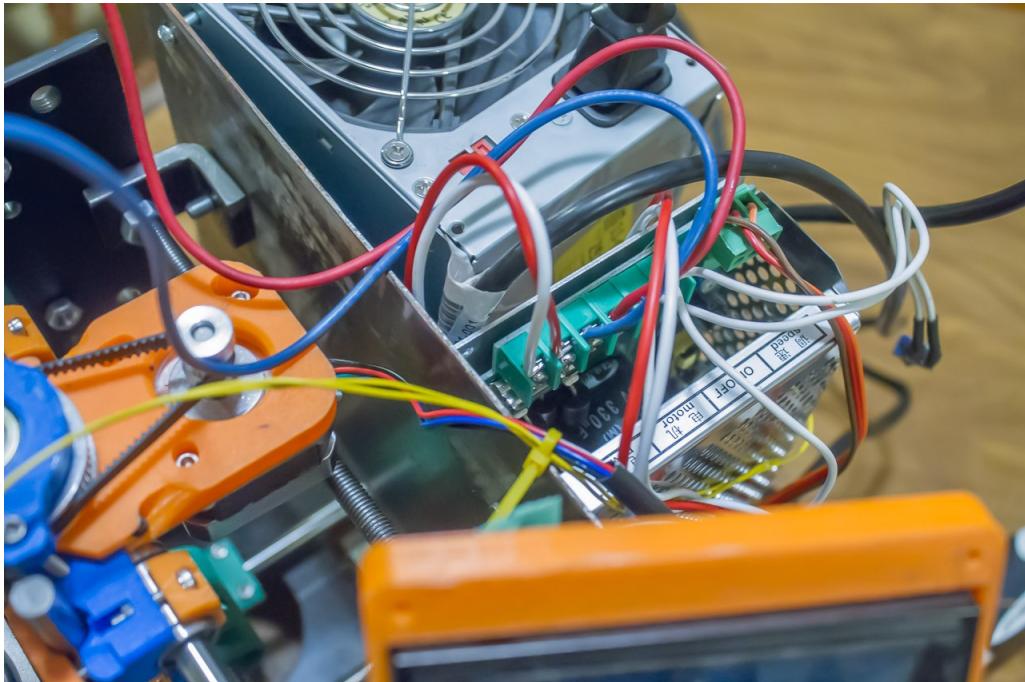


Slide the power distribution box into the side of the right rail mount.



Connect the spindle power wires to the spindle power supply.





Loosely attach the E-stop button to the power distribution box.

Electronics:

NOTE: I intend to share the control board schematics eventually. Time is a scarce commodity!

The mill can be run using a RAMPS board, just like a typical 3D printer. I'm using a custom control board with an ATMega328. The 328 is economical and easy to replace if the magic smoke ever escapes! However, it takes a few trick to get a 328 to run three stepper motors, endstops, the spindle, a user input system, the LCD, and an SD card reader.

I have the current firmware available on github at github.com/td0g/cnc_mini_mill. It barely fits onto the ATMega, what with the Liquid Crystal, I2C, SD, SPI, EEPROM, and Accelstepper libraries included! The firmware is mature enough that I can use the mill without a PC. It can run from the SD card and can even probe a surface grid and save the results to the SD card. More on probing and levelling below.

The user interface is a 2004 LCD display with an I₂C module. This requires only two pins from the controller. Unfortunately this means that any data sent to the LCD takes much longer, but this is accounted for in the code. Each time the display updates while the motor is moving, only one character is printed at a time, which takes about 300 microseconds. Then the controller will go back to driving the motors before printing another character.

An analog joystick is used for the user input. The controller performs a non-blocking analog - digital conversion, which means that it can drive the motors and control the display while the joystick is being polled. This results in a very smooth operation.

The endstops are connected to a single analog pin, which is polled in the same way as the joystick. Since the ATmega328P can perform analog - digital conversions at about 10 kHz (without adjusting the ADC Prescaler), the joystick axis and endstops are each polled at about 3 kHz. Currently there are only three endstops (max X, min Y, max Z) but I would like to add three more so there are min/max endstops on each axis. The min and max endstops would be wired in parallel, so the controller would have to intelligently decide which one was pressed based on the direction the axis was moving (IE. if gantry is moving left, then the Min endstop would be pressed).

The electronics are mounted on a plate behind the gantry. I added some rubber isolators to prevent the plate from vibrating.

Levelling:

The controller has two leads with alligator clips. The clips go on a PCB board and the cutting tool (while the spindle is off!) and can probe the surface of the PCB. The firmware allows the user to select an area and probe the surface. The two-dimension probe depth array are saved to a file on the SD card.

I wrote a VBScript program which can take the array and apply the data to a gcode file. It divides each movement in the gcode into shorter segments, and the Z position is adjusted based on the probe data. It uses a bilinear interpolation model, which isn't the best but is functional. I started working on a Python script

to do the same, but with a cubic spline model instead. Need more time to complete that.

Last modified: 2019-05-05