Cylindrical 3D Printer Guide

Luke Vandenberghe

Last Updated: October 26, 2020

ANGL3D
PRINTING

You can contact me at `angl3d.printing@gmail.com`

Before Getting Started

This is a guide that will teach you how to convert a traditional Cartesian FDM 3D printer into a cylindrical 3D printer. This project involves the installation of an apparatus to replace the y axis of the 3D printer with a rotating cylinder as well as the use of custom slicing software. By the end, you will be able to print a couple models with a cylindrical 3D printer and slice 3D models with non-planar slicing techniques. This project was designed to work with the **Anet A8** with **Marlin firmware**. However, this project can likely be adapted to work on other Cartesian FDM 3D Printers through a few modifications in the slicing software and the cylindrical 3D printer apparatus. The slicer was programmed from scratch in **Python 3.7** using the **numpy-stl library** `https://pypi.org/project/numpy-stl/` and the **Panda3D Engine** `https://pypi.org/project/Panda3D/`. Review guide for more details of how to setup and use the slicer.

PLEASE BE AWARE: This project has many problems. The slicer has bugs and can be quite slow. This is a project that is very early in development. It was made by a student (me) who has never done this before. **USE AT YOUR OWN RISK**! I did this with the intent to make something cool and I did my best to make it so you can make something cool too.

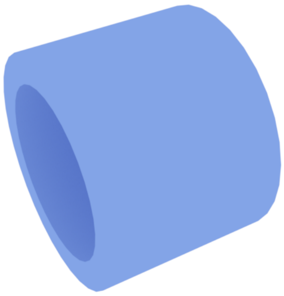
Introduction

There are many challenges to 3D printing geometries that posses curved features on mainstream Cartesian FDM 3D printers. For starters, it is difficult to achieve a high degree of accuracy while not also sacrificing speed. One factor that decreases the speed of the print is the fact that model itself must have a high polygon count to provide the required detail needed to produce an accurate approximation of the curve. With increased amount of position instructions, the printer is required to make larger amount of fine movements, which in turn limits the print speed. Another challenge is printing overhangs on 3D models with cylindrical features. There is often a need for support structure on models with cylindrical characteristics due to the fact that if the model has overhangs and possesses cylindrical symmetry, the overhangs will extend in multiple directions. With no initial orientation that can compensate for the multiple overhangs, support structure is required. The problem with support structure is that is costly, takes time to remove, and often ruins the surface quality of the model. A possible solution to these issues is the cylindrical 3D printer. A printer that can print a subset of 3D models with curved features at higher speeds without sacrificing print quality. It reduces the need for support structure and produces much smoother curved surface finishes.

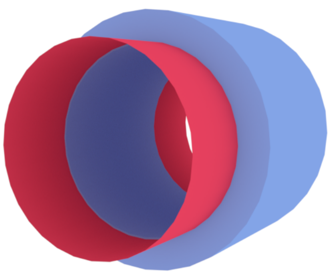
Concept

To convert a Cartesian FDM 3D printer to a cylindrical 3D printer you will have to replace the y-axis of your 3D printer with a rotating cylinder rather than a translating bed. The print head applies filament directly on top of the cylinder and moves along x-axis, while the cylinder rotates clockwise and counter-clockwise around the x-axis. By doing this, we can supply the printer regular instructions to print a flat layer like before however, the layer will be printed on a curved surface. The beauty of this is by telling the printer to move in a straight line, you can quickly print a “perfect”circle. The printer can be given instructions to print a simple filled in rectangle and it will print a “perfectly” curved cylindrical shell.

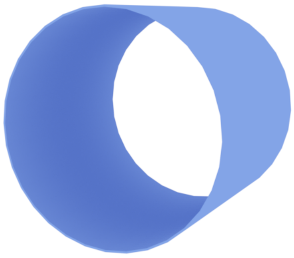
The planar slicing techniques used in the mainstream slicers are not suited for an add-on like this. This add-on requires a custom slicer. For the rest of this section, I will be taking you through the basic premise of how the models are broken up into layers in this new slicer. We will start with this cylinder with a hole in the middle or in other words, a bushing.



Instead of making slice through the bushing with a plane like a typical slicer, I can make a cylindrical slice through the bushing like so and find where the curved surface intersects the bushing and make a slice.

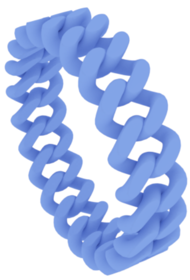


With that slice I can obtain a layer that makes up the model. This is good however, we have a problem. the layer is not flat. In order to make it flat, I have to make a cut and unwrap the layer.

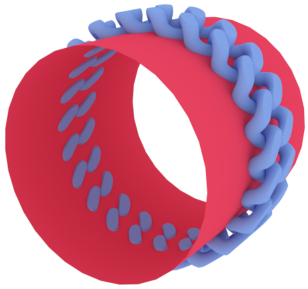


Performing this, we now get a flat rectangular layer. The left end of the rectangle will be defined as y=0 and the right end will be defined as the total change of y needed to complete one full revolution with the cylinder. This is obtained through a calibration step that I will get into later in the guide.

This technique does not just work with bushings, it can work with a whole variety of items such as bracelets.



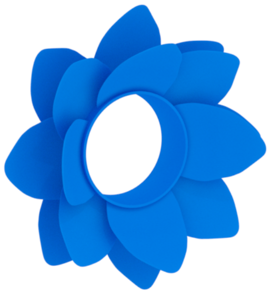
Like before, let's make a cylindrical slice through the model and obtain a layer.



We can then take this layer and unwrap it to be on a plane.

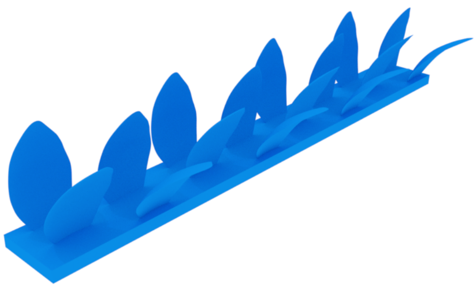
Just like before, but now with just a more complex geometry, we can supply the printer a series of these layers and acquire a bracelet using the cylindrical 3D printer.

Now, let's take a look at how to take advantage of the cylindrical 3D printer to print a model without support structure with this 3D model of a lotus.

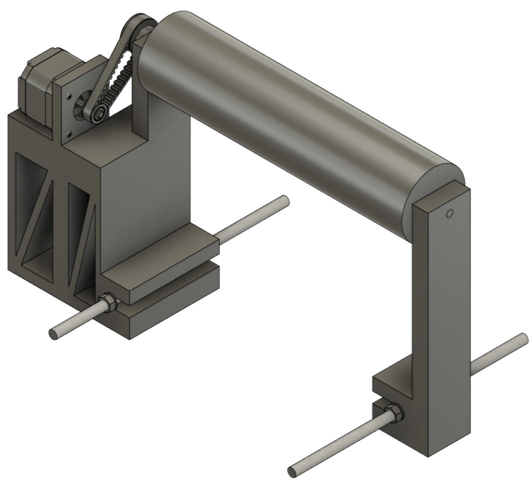


Printing this model all at once with support structure on a Cartesian 3D printer would be a nightmare. It is surrounded by petals protruding in a large variety of directions and elevations. This model would be covered in support material that would be difficult to remove without breaking any fragile petals. Generally, a model like this would have to be broken up in multiple pieces to print. However, with the cylindrical 3D printer this model can be printed fully without support material.

By unwrapping the full model using the same concept that we have been exploring, we can see why this is possible. There are no major overhangs in this unwrapped model. This model is essentially what is being printed just on a curved surface. Each flat layer that makes up this model is being printed on top of the cylinder. By printing radially outward each petal can be printed without support structure.



Setup Cylindrical 3D Printer Apparatus



To view how the apparatus is assembled, how it installed on the build plate, and the parts needed to construct it open the **assembly drawing pdf**. Some of the parts can be bought in a package at the amazon link `https://www.amazon.ca/Houkr-Aluminum-Bearing-Synchronous-Printer/dp/B07KBFWD6W/ref=sr_1_3?dchild=1&keywords=stepper+motor+with+belt&qid=1593804382&sr=8-3`. Also, part of the apparatus can be 3D printed and the STL files for the printable parts can be found at the **cylindrical 3d printer apparatus folder**. A couple of points for the installation of the apparatus that are not mentioned on the assembly drawing:

1. Make sure to raise the **Z Endstop** so the print head is level with the top of the cylinder. If you are using an Anet A8 I made a custom z endstop raise you can 3D print in the STL file folder to more easily raise the end stop.

2. I recommend taping a removable sheet on the cylinder to make it easier to take the model off of the cylinder afterwards.

3. The rod and cylinder have to rotate as one unit. I used glue on both ends.

4. I used the pre-existing y-axis stepper motor and moved it on to the apparatus but you can get another stepper motor and wire it to replace the y-axis slot on the mainboard.

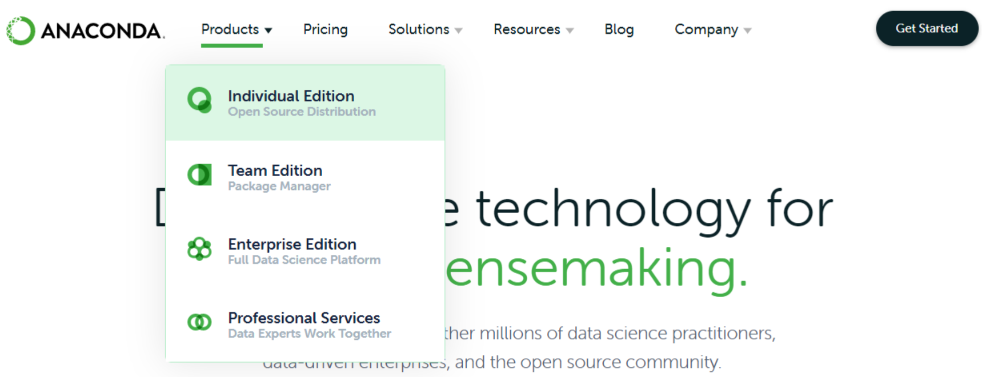
5. I used a metal cooking pin for the cylinder.

At the end of document there are some pictures of the cylindrical 3D printer apparatus I built that you can use as a reference.

Setup Slicer

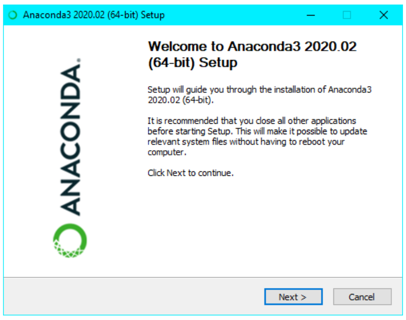
This is a setup tutorial for **Windows** users. We will be installing **Spyder** which is a cross-platform Python IDE. The slicer was written using **Python 3.7**.

We will first need to download **Anaconda**. Here is a link to the Anaconda website (`https://www.anaconda.com/`) . When you get to the site, go to the **Products** tab and select **Individual Edition**.



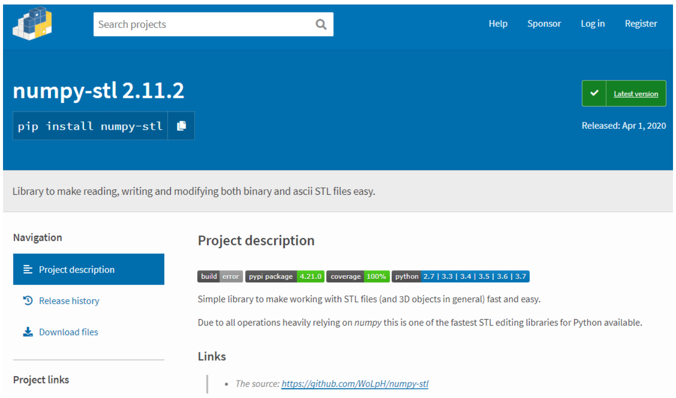
Scroll to the bottom of the page and select a Anaconda installer. Make sure to select an installer under **Python 3.7**.

Once the installer is downloaded open it and this tab should appear. Go through the Anaconda installation process and once it is finished installing continue to the next step.

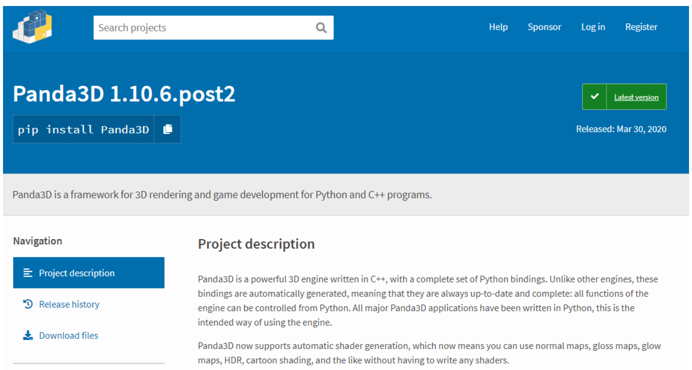


We now need to download the **numpy-stl** library and the **Panda3D** engine which can be done fairly easily using pip install. If you want more information about either of these, here are the two links to the sites. You may also need to install **PyQt5** if you do not have it.

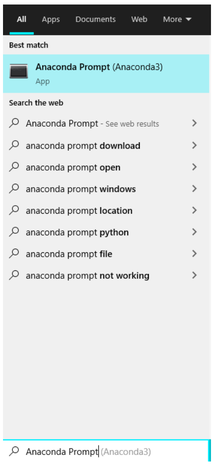
`https://pypi.org/project/numpy-stl/`



`https://pypi.org/project/Panda3D/`



Navigate to the **Ananconda Prompt**

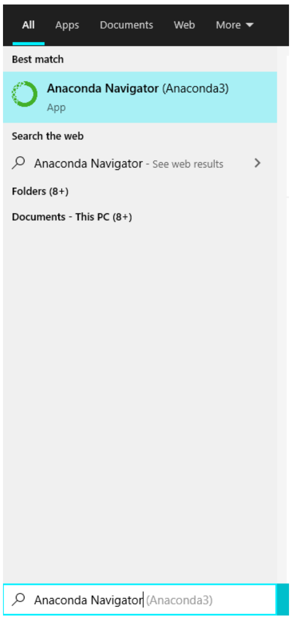


and type in these two commands.

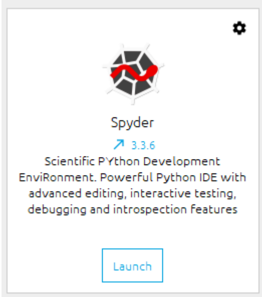
```
>pip install numpy-stl
```

```
>pip install Panda3D
```

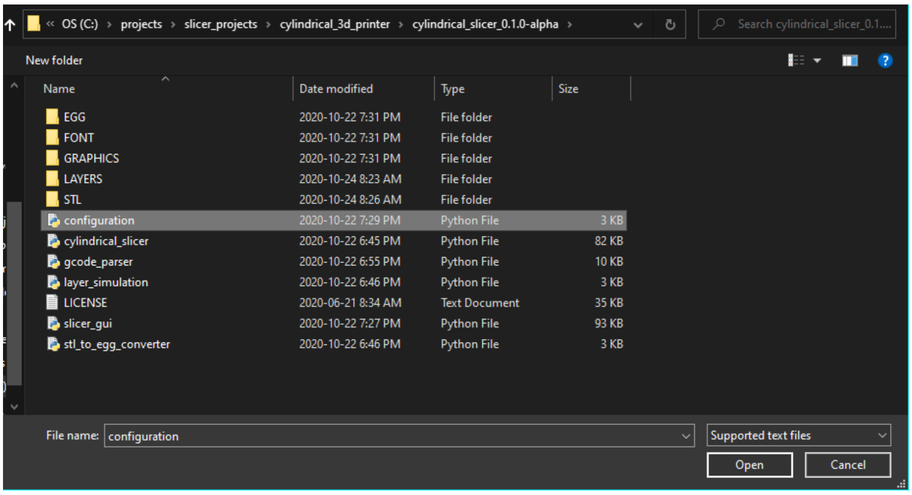
Next, Download the the source code containing the slicing software if you have not already then find the **Anaconda Navigator**. Once you find it, start it up.



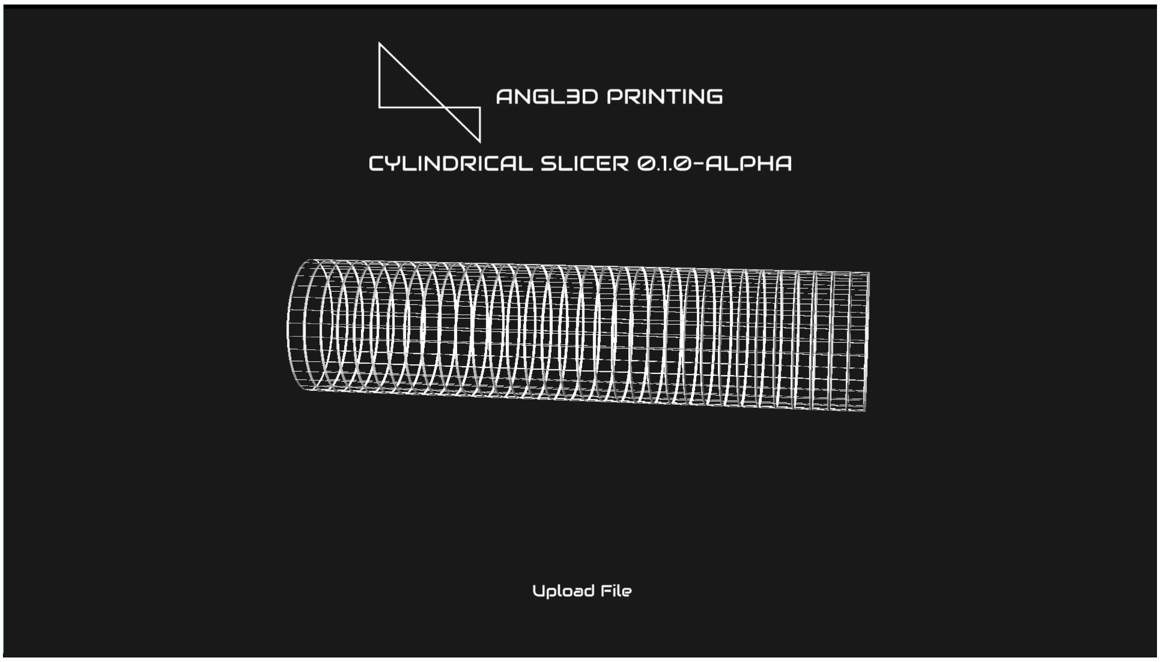
You will see a couple of different applications on the **Anaconda Navigator** home screen. Locate the application labeled **Spyder** and launch it.



After the Python IDE starts up, at the top left corner select **File** and then **Open**. Find the folder containing the slicer software that you unpacked earlier and open **configuration.py**.



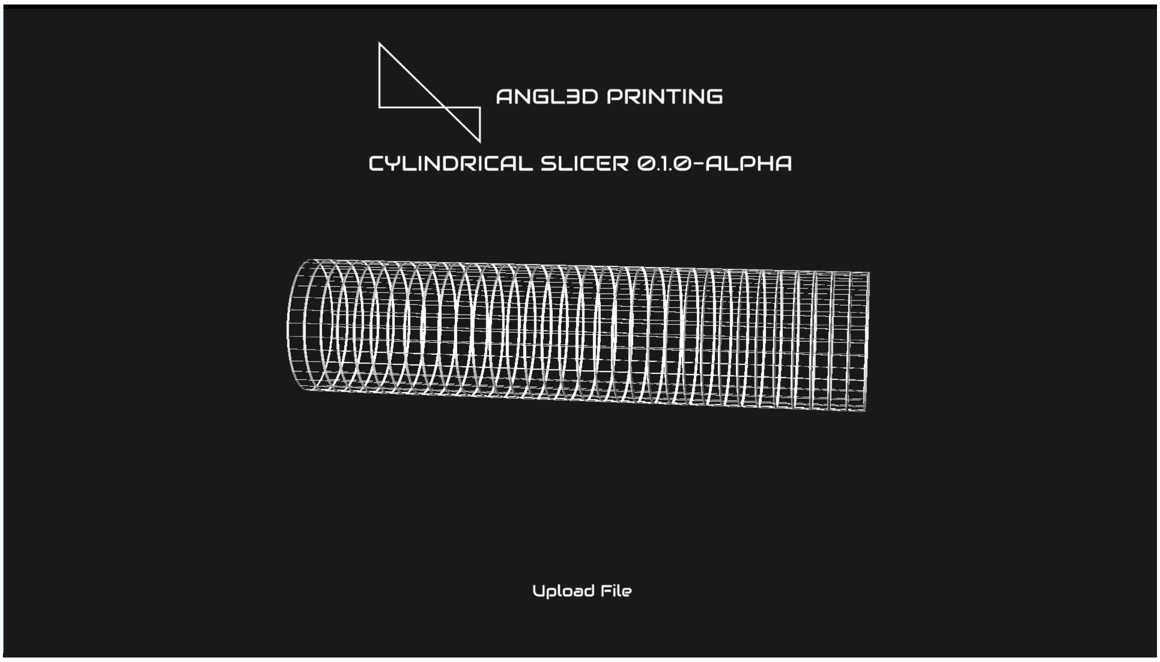
Once **configuration.py** is open in Spyder, click the **green play button** on the tool bar **or** press **F5** on your keyboard to run the file. The start screen of the slicer should appear on your monitor.



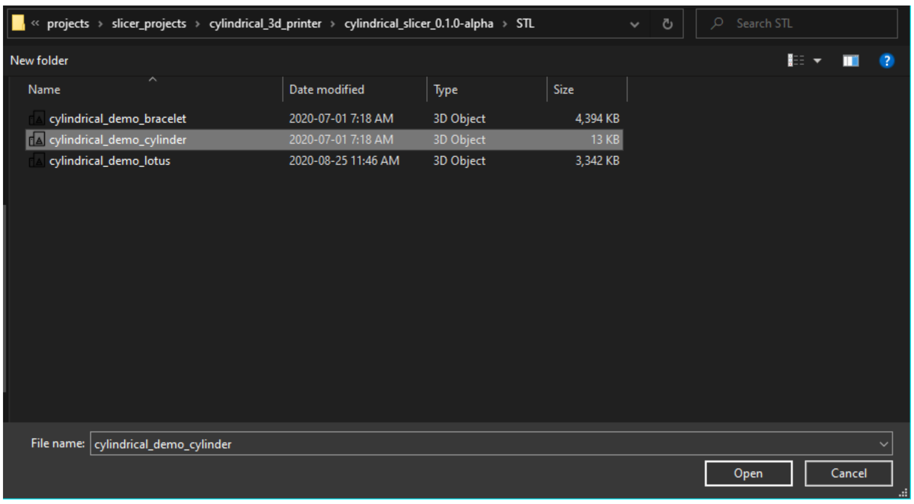
Congratulations! You got the cylindrical 3D printer slicer up and running. In the next section, I will be giving you a little tour of the application.

Tour of Slicer

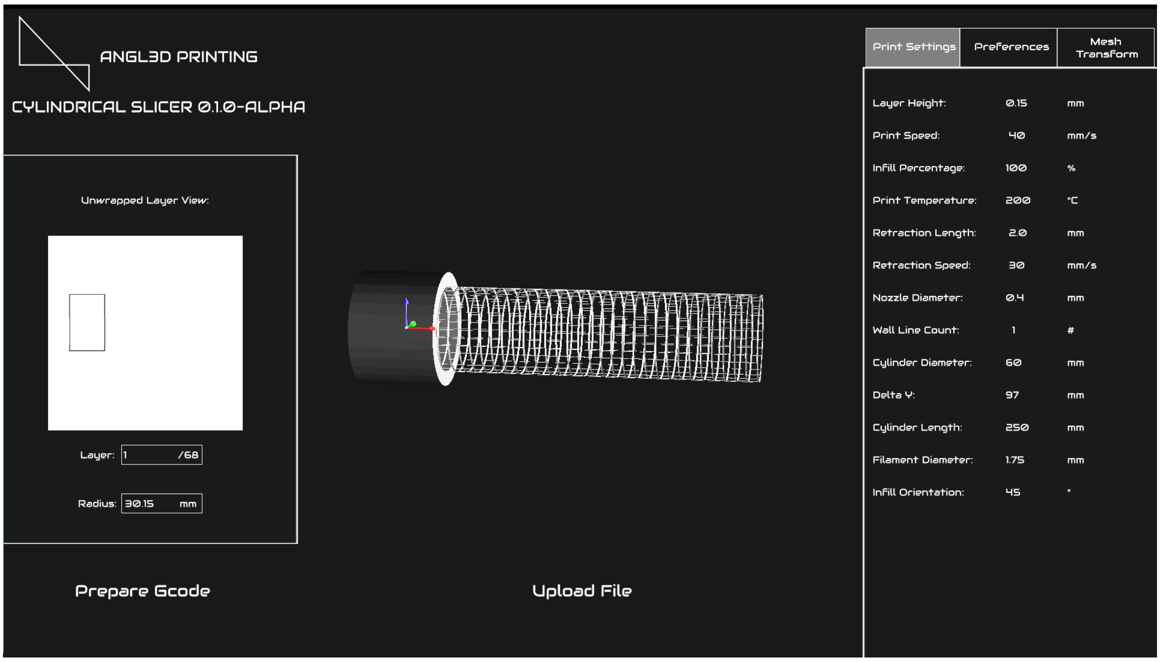
Note before starting: Please be patient with the slicer, it can take a while to load models. Also when in the slicer, depending on how detailed the model is it may take a while to perform actions. Just wait a little bit for the data to process before assuming the program crashed. Alright, let's start by selecting the **Upload File** button located at the bottom of the screen.



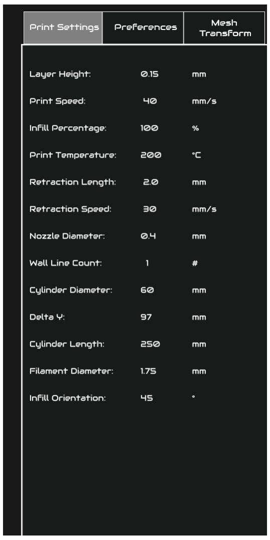
Navigate to **cylindrical_3d_printer_slicer** folder and inside the **STL** folder select **cylindrical_demo_cylinder.stl**. In this slicer you can only upload one model to the program. This slicer does not support multiple models. Also there is no option to delete the model. If you want to upload and slice a different model you have to restart the program.



Click **Open** and this screen should show up. Now that we uploaded our file to the slicer, let's explore the different features that this slicer has to offer. (You can click and drag the middle mouse button to rotate the view and use the scroll wheel to zoom in and out).



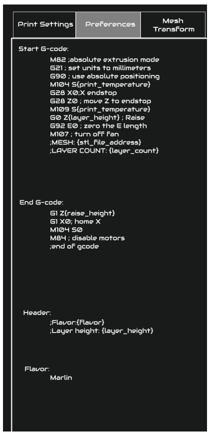
If you take a look at the **Print Settings** tab you will recognize that most of these adjustable parameters are general settings that you typically see in a traditional slicer. You have probably also noticed that there are some new settings such as **Cylinder Diameter**, **Cylinder Length** and **Delta Y**. Cylinder Diameter is pretty self explanatory as it is the diameter of cylinder you install on the apparatus. Just so you are aware the cylinder must have a diameter of 60mm in order for the demo models to work. Delta Y is obtained through calibration of your printer. Delta Y is the change in the y position of your printer after one full revolution of the cylinder. You can modify all these parameters in the configuration file. Make sure to restart the slicer after making changes to the configuration file for them to take effect. The dimensions of the cylinder seen in the slicer will actively change when you change the parameters.



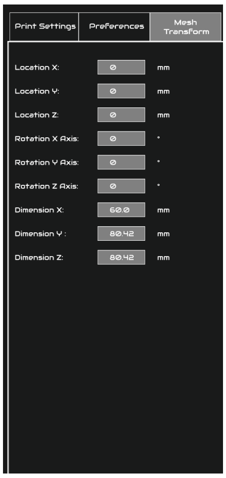
To get Delta Y, at your printer mark down the initial y position and the increase y incrementally until the cylinder has made one full revolution.

Mark down the final y position and take the difference to get your delta y. In my case it was 97 mm but yours may be different.

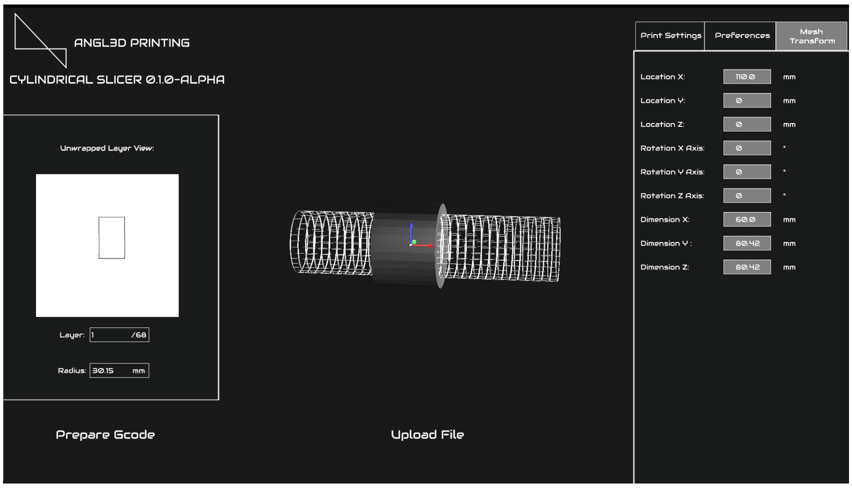
Moving to the **Preference** tab you will see all the start and end Gcodes. You can edit the Gcode strings directly in configuration.py. Also, here is a link `https://marlinfw.org/meta/gcode/` to get information about the Marlin Gcode.



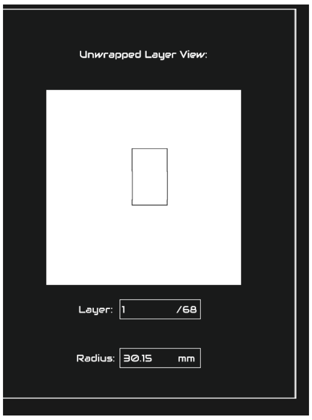
Next head to the **Mesh Transform** tab. Here you can move and rotate the model around on the x, y and z axis. You can center the model by double clicking on the **Location X** button, typing 110, and then pressing enter.



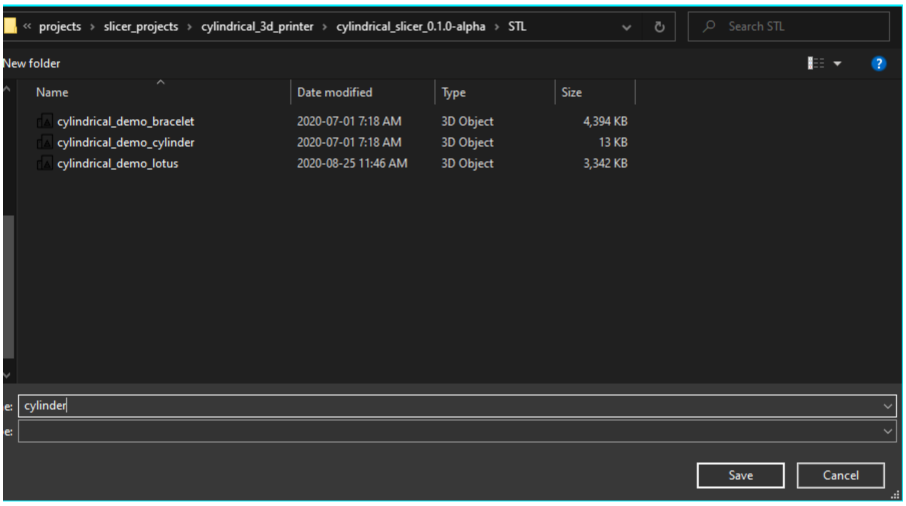
You should see that both the location of the model on the cylinder change, as well as the position of the unwrapped layer in the layer viewer. In terms of model orientation, the rotation mechanics of this slicer are not that great and I suggest exporting the model in the proper orientation before uploading to slicer. You will learn how to orientate your model in the **Using Your Own Models** section. However, if you do choose to use it, you can only rotate around one of the axis and get expected results. You can't rotate around a combination of multiple axis.



At the layer viewer tab you can see the unwrapped view of each layer at different radii. You can click on the layer text box and type in the desired layer you want to see. Press enter when you finish typing.

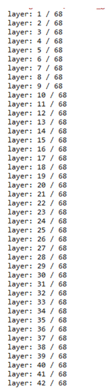


Since it is just a rectangle the layer viewer is not all that interesting. Let's start preparing the Gcode file. Select the **Prepare Gcode** button at the bottom left corner and a dialogue box should open. Type in **cylinder** and then save. ".gcode" a will be added on to that file name automatically when the program finishes writing the file.



You will see a little loading animation on the screen after you press save.

If you want information head to the python console and you should see how many layers there are left to slice. Please be patient. It may take a while.



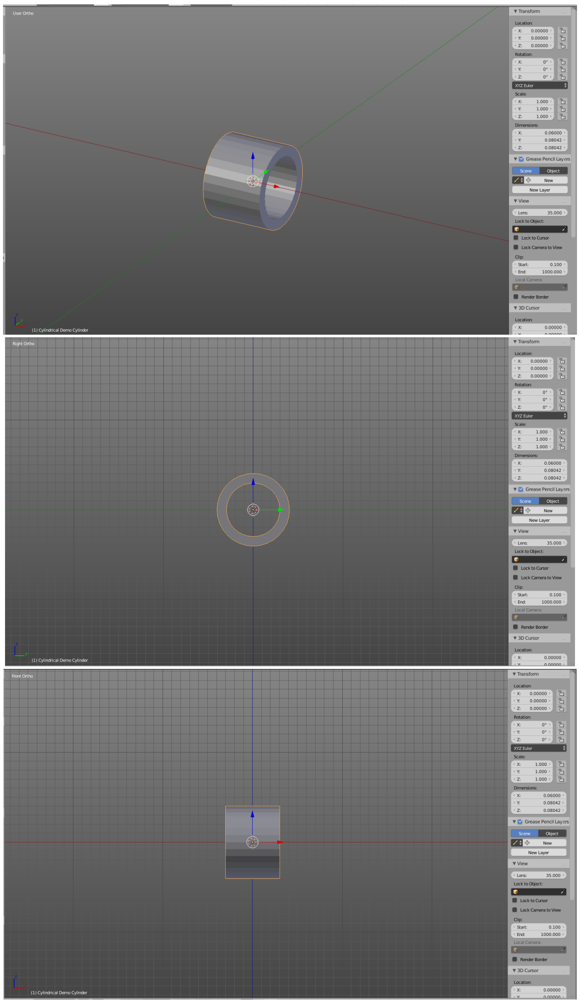
Finally, when everything is finished the loading animation should disappear and the Gcode file should appear to wherever you decided to save it to.

MAKE SURE TO CHECK THE GCODE AND MAKE SURE EVERYTHING LOOKS OKAY BEFORE PRINTING!!!

Now that the Gcode is ready, let's start printing.

Using Your Own Models

When determining a model that is compatible with this type of 3D printer it is important to understand that the model has to be able to fit around the cylinder that is acting as the new build surface. The inner diameter of the 3D object being printed should match the diameter of the cylinder otherwise you may get undesired results. You can modify the length and diameter parameters of the cylinder in the configuration.py file to match your setup. Models you provide do not have to be cylindrically symmetric. In fact, the model does not have to fully encompass the cylinder. Models with curved bases that match the curvature of the apparatus and partially wrap around the cylinder can be used. When exporting your own STL make sure your model is centered at the origin and is orientated so that its axis is along the x-axis. Also make sure the mesh has clean topology! I suggest before you upload your model to the slicer I recommend you to first put it in blender and use the remove doubles feature with a merge distance of 0.1mm. Example:



Printing With The Cylindrical 3D Printer

Once the cylinder mechanism is correctly installed on the build plate and the print head is properly level with the cylinder you can start printing. You should upload the Gcode file to your printer and start printing **cylinder.gcode**.

WATCH YOUR PRINTER AT ALL TIMES WHILE PRINTING!!!

Hopefully, if all went well you now have officially printed a full model on your new cylindrical 3D printer! There are two other demo models (**cylindrical_demo_bracelet.stl** and **cylindrical_demo_lotus.stl**) you can try running through the slicing software and try printing (The slicing is really slow for these models and will take a while to prepare the Gcode. I am working on it to speed it up). The slicing software is still very early in development.

Troubleshooting Slicer

If you get an error in the slicer, there is a likely chance that the problem is from the program not being able to construct a closed loop on the current layer it was trying to slice. The problem usually originates from the 3D model that is uploaded. Check if the model has double vertices, loose vertices, holes, polygons that are partially attached to the model that look like "flaps", etc. If the mesh topology is not clean the slicer will not work. I recommend that you use low poly 3D models for this slicer. Also make sure the model is positioned and orientated correctly on the cylinder. The slices are incremented radially and if one of the cylindrical slices does not intersect with model than an error will be produced. Do keep in mind that this slicer is still very early in development and will have bugs. So if you are sure the model is fine, it very well could be a problem with the software. I cannot guarantee it will work for all other external models you throw at it.

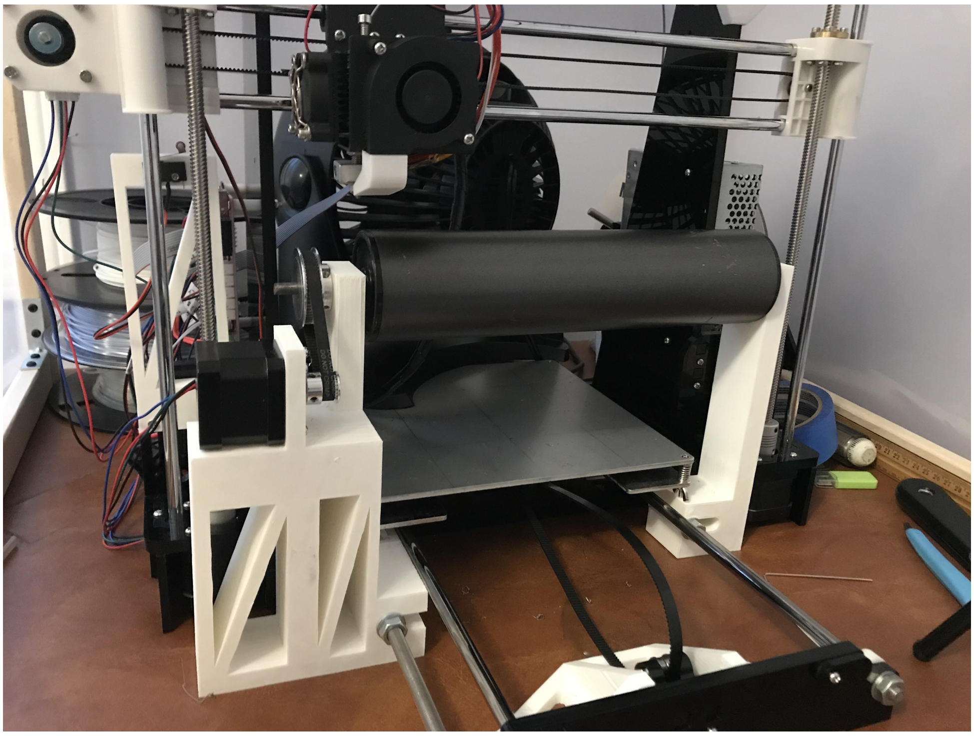
The General Procedure

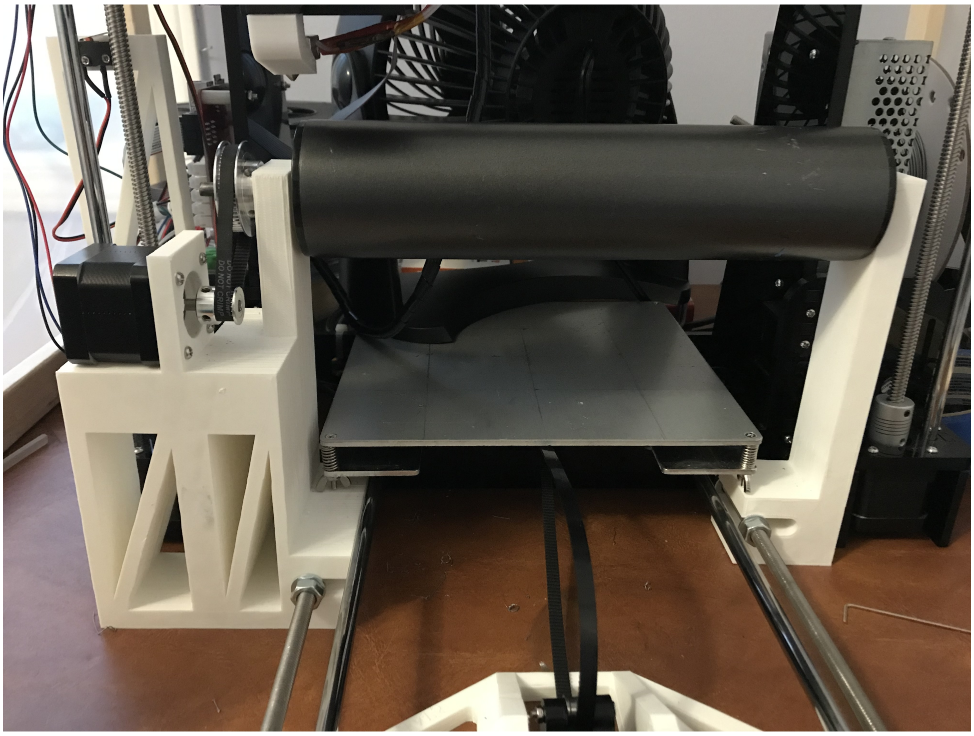
1. Upload STL File
2. Adjust Print Settings
3. Parse Gcode
   - Gcode
4. Print with Gcode

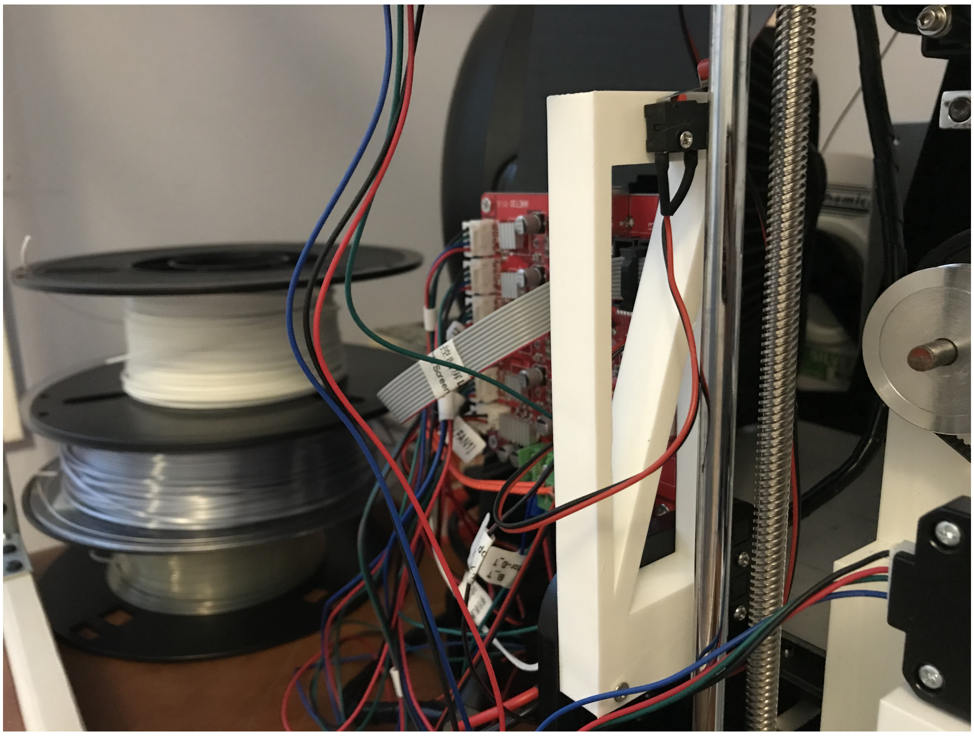
Conclusion

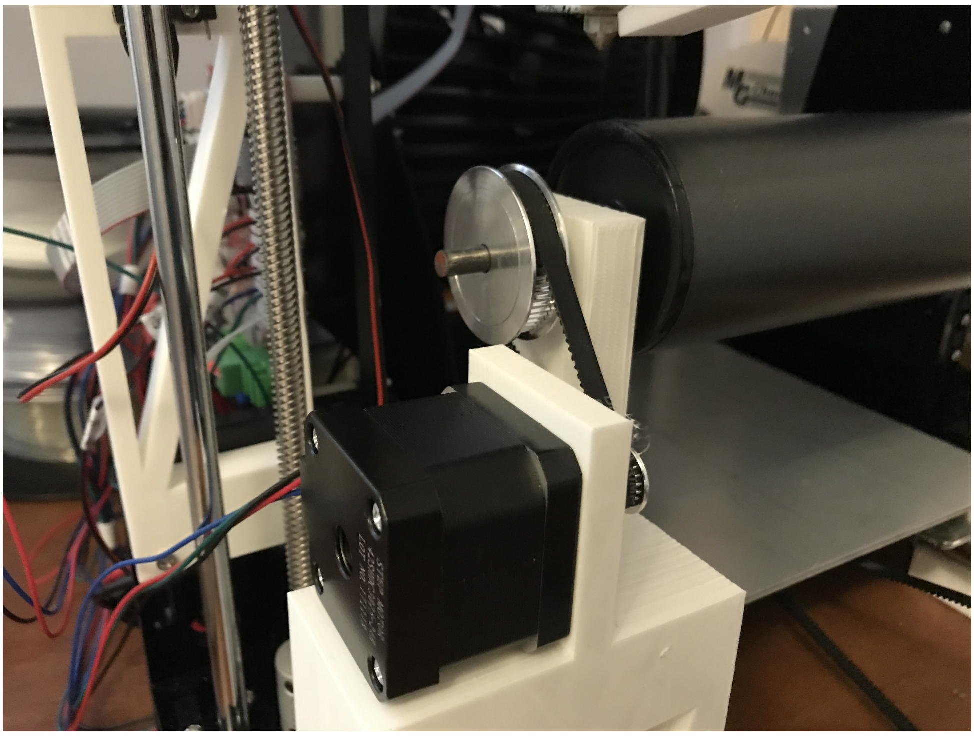
This project is a first generation prototype and can be significantly improved. Longer term opportunity exists to develop an improved version. One of the issues is that there is a loss in print accuracy as the print radius increases due to the fixed amount of steps of the stepper motor but the increasing circumference of the model. A way to improve this is through more involved calibration, modification of the firmware, or using a better quality stepper motor. The stability of the apparatus can also be improved and a better technique to remove models off the cylinder more easily would be beneficial. Feedback is much appreciated and I wish you the best!

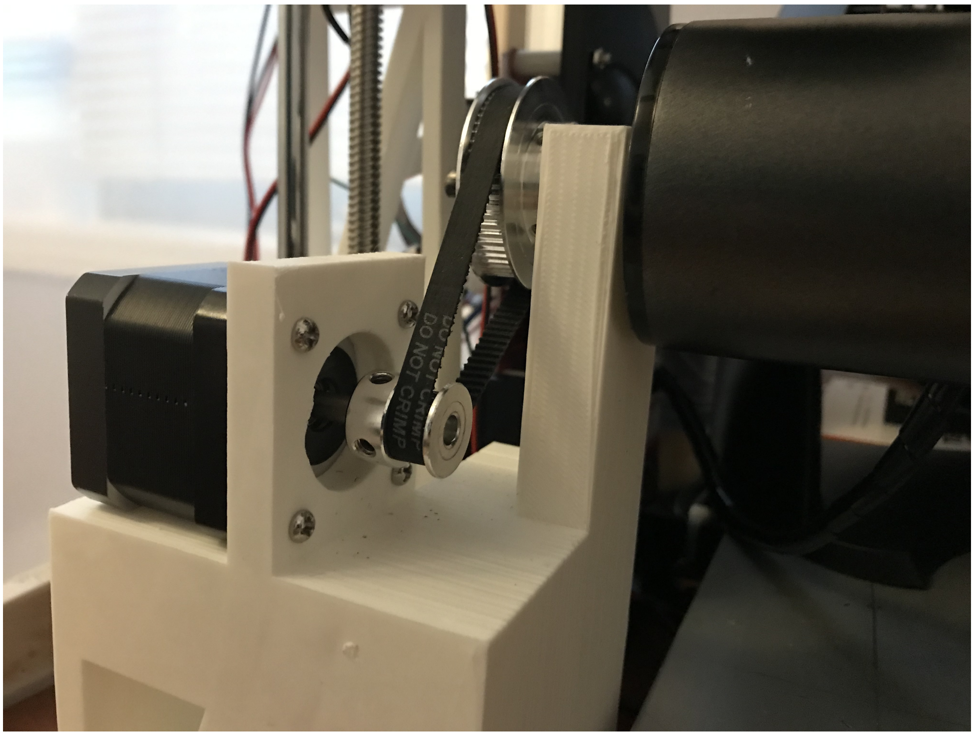
Apparatus Reference Images










DO NOT CRIMP