MultiColour Mechanism (MCM)

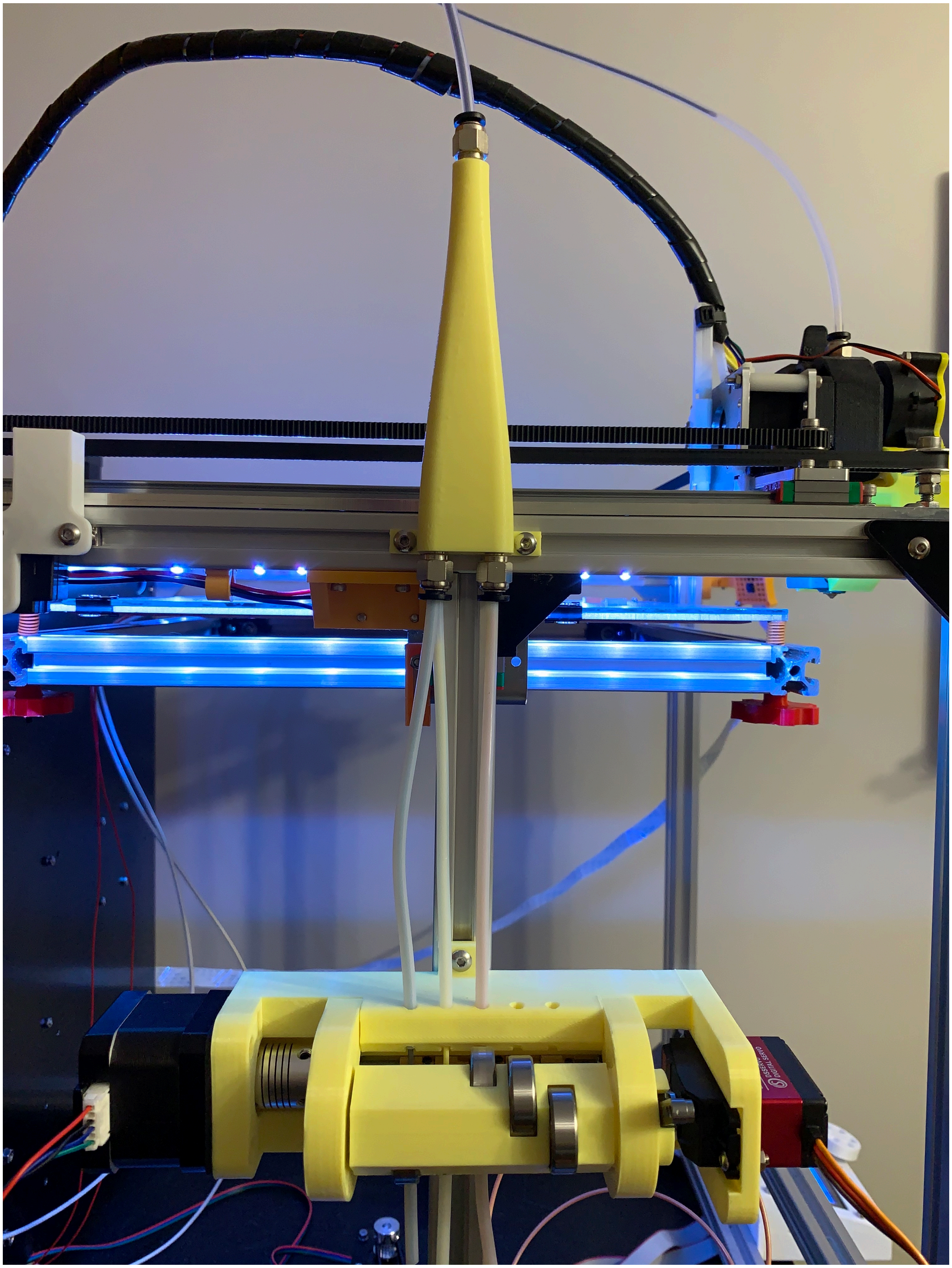


Table of Contents

System Requirements and Preconditions

In theory, this MCM could be adapted to almost any printer; I've designed it to work best with the Seckit SK-Go. Below are some considerations for how this MCM was designed and how it drops in to work with the SK-Go.

- Your mainboard needs at least 2 extruder slots (SKR 1.3 or 1.4 are ideal).
- Your firmware should be Marlin, and you should be comfortable making firmware changes (for the purposes of this guide). More on firmware changes later.
- Your mainboard needs a servo control pin, but the servo may need external power depending on the size and voltage requirements.
- You should slice models using PrusaSlicer. The colour switching is done using gcode, and PrusaSlicer seems to be the best at handling multicolour prints.

There is one component of this project where I am still unsure. This is the assembly of the main body to the idler mounts. Currently I have printed small tubes out of TPU, and they seem to do very well as springs. All of the springs I own are far too rigid for this project, but I believe some softer springs would be the better way to go. If I run into issues with the TPU springs, I'll look at other options.

The only servo I own is a 20kg, 180º servo. This means that, with the current idler design, I can only use 3 filaments. However, I believe that this system could be adaptable to use so many more. I'm going to start by buying an identical servo with a range of 270º, and I'm going to experiment with the "circumferential arc length distance" (I have no idea if this is a word or not) between the bearings on the idler to see if I can fit more filaments into the same range of motion of the servo.

I'm currently using a brass brush which I've mounted to the SK-Go extrusion; this helps wipe the nozzle between prints, but can get gunked up quickly. I haven't included this in the assembly guide because every printer has a different home point and bed location based on the user's build. Designing a simple mount is easy enough for each person to do if they want.

List of Materials & Costs

Item	Quantity	Possible Source	Approx Cost
M4 x 16mm Bolts	8	Anywhere	-
M4 x 12mm Bolts	2	Anywhere	-
M4 Nuts	10	Anywhere	-
M3 x 12mm Bolts	2	Anywhere	-
M3 Nuts	2	Anywhere	-
608ZZ Bearings	8	Anywhere	$10
PTFE Connector - M6 thread - Through-fitting for 4mm tube (see link for details)	6	PTFE Connector	$10
PTFE Tubing	As much as you want/need	Anywhere	$10
NEMA17 Stepper Motor and relevant cables	1	Anywhere	$20
Servo Motor - High torque strength (20kg) - Large control angle (270º) - You may need a buck converter to power the servo externally from your 3D printer mainboard (6V+)	1	Servo Motor	$20
Extruder Gears	5	Extruder Gears	$10
5mm Rod Stock - (length) x (quantity) - 12.9mm x 5 - 105mm x 1	180mm		
5mm to 5mm flex coupler	1	Coupler	$5
		Total:	**$85**

If you're lazy and impatient like me, you probably want fast delivery of these parts all from the same supplier, so you'll have to spend around $80-$100. If you know different suppliers that are cheaper and maybe a bit slower, I guarantee that you could find all the parts above at much cheaper than I did. You could probably get under $50.

Printing the Parts

A plate of all parts is available on Thingiverse, which shows the correct print orientation. Total print time is approximately 10 hours, at around 160g of filament.

Recommended print settings:

- PLA
- 3 bottom layers
- 3 top layers
- 3 perimeters
- 15% infill, grid pattern
- 0.4mm nozzle
- 0.2mm layer height
- 0.8mm (2 line) brim

Supports are not needed for most parts; the Idler will need supports underneath the round edge (see screenshot) because most FDM printers cannot create a perfect round bottom unless your cooling settings are perfect. The SK-Go can handle support distances of 0.04mm, which come off easily and leave a near-perfect round bottom.

I recommend printing at a layer height of 0.2mm for best resolution, but 0.24mm is fine for most parts. However, the Bowden Splitter should be printed at 0.2mm so the interior walls are as smooth as possible and don’t cause any additional friction for the filament in motion.

Firmware Modifications

Frankly I've made so many preferential changes to my firmware that it doesn't really make sense for me to share my config files. Instead I'll explain the necessary changes below.

I prefer using a direct-drive (DD) extruder on my SK-Go because I find it gives a better print quality. In order to keep using the DD extruder with my MCM, I've made some simple firmware changes. If you don't care about the DD extruder and you're fine with a Bowden extruder, you probably don't need some of these firmware changes; you could get by with only one stepper motor acting as the extruder and the filament switcher.

In order to use the DD extruder with the filament switcher, I've enabled Mixing Extruders. This will allow me to move both extruders at the same time. This is important for ensuring that when the filament has passed from the MCM, through the reverse Bowden, once it reaches the DD, that the DD properly grips the filament.

```
#define MIXING_EXTRUDER
#if ENABLED(MIXING_EXTRUDER)
  #define MIXING_STEPPERS 2        // Number of steppers in your mixing extruder
  #define MIXING_VIRTUAL_TOOLS 16  // Use the Virtual Tool method with M163 and M164
  #define DIRECT_MIXING_IN_G1    // Allow ABCDHI mix factors in G1 movement commands
  #define GRADIENT_MIX           // Support for gradient mixing with M166 and LCD
  #if ENABLED(GRADIENT_MIX)
    #define GRADIENT_VTOOL       // Add M166 T to use a V-tool index as a Gradient alias
  #endif
#endif
```

I've discovered that, even though I have Distinct E Factors enabled, this does not work for the Mixing Extruders. Mixing Extruders applies the E0 steps/mm to both extruders.

```
#define DISTINCT_E_FACTORS

#define DEFAULT_AXIS_STEPS_PER_UNIT   { 100, 100, 400, 413, 100 }
```

In order to get around this, you can set a gcode mix factor that is equivalent to the ratios of steps/mm between the two extruders. For example, my Bondtech BMG DD extruder has a 3:1 ratio, with 413 steps/mm. The MCM extruder has 100 steps/mm. You can do some math to find that 413+100=513, then 413/513=80.5%. Therefore, the mix ratio is approximately 80:20 (I'm currently using 79:21 just to err on the side of caution when both extruders are engaged). This is what the gcode looks like to set the mix ratio:

```
M163 S0 P0.79 ; 79% on the DD nozzle extruder
M163 S1 P0.21 ; 21% on the MCM extruder
M164 ; set and activate the mix factor
```

The MCM uses a servo angle and some mathematical expressions in PrusaSlicer to choose which filament should be activated. Servos need to be activated in your firmware.

```
#define NUM_SERVOS 1
#define SERVO_DELAY { 100 }
//#define DEACTIVATE_SERVOS_AFTER_MOVE
#define EDITABLE_SERVO_ANGLES
```

The Idler of the MCM currently has bearings positioned every 60º around its circumference (spaced apart lengthwise by the size of the extruder gears), but I'm experimenting with the arc length and resolution of the bearing positions to see if I can put more bearings within the same servo control angle. In any case, since each colour is separated by a 60º servo angle, the math works out using the expressions below.

- M280 P0 S{(previous_extruder+1)*60} ; set servo position of previous colour
- M280 P0 S{(next_extruder+1)*60} ; set servo position of next colour
- M280 P0 S0 ; disengage idler

Note that PrusaSlicer and Marlin use E0 as the default extruder and default colour, so it's necessary to add +1 to the mathematical expression. The placeholders "previous_extruder" and "next_extruder" are automatically calculated by PS and appear correctly in the output gcode. However, it is necessary to do some post processing of the gcode, because PS will output a T* (T1, T2, etc) toolchange command each time the colour changes. Simply replace all T* commands with T0 or delete them all. More on gcode later.

The servo angle of 0º is the angle at which none of the bearings engage the filament, so the idler is completely disengaged and the filament can move freely.

Depending on the length of your SK-Go's reverse Bowden, you may need to change the maximum allowable extrusion length:

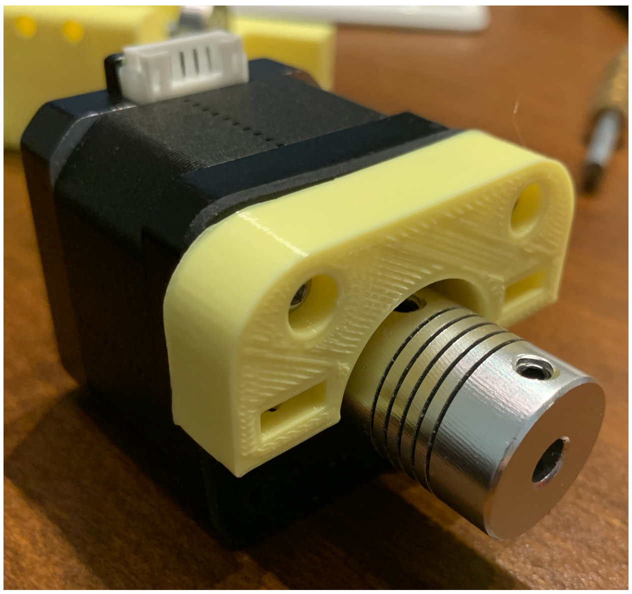
```
#define PREVENT_LENGTHY_EXTRUDE
#define EXTRUDE_MAXLENGTH 1000
```

Finally, make sure your pins for E0, E1, and the servo are set correctly for your mainboard.

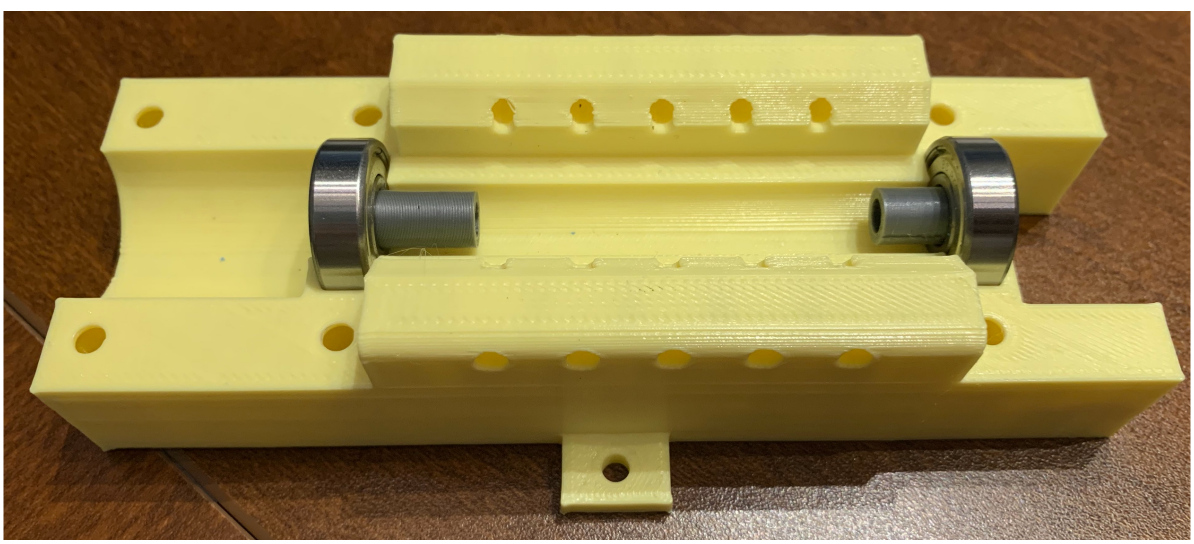
Assembly Guide

Insert the 5mm flex coupler onto the Nema17 motor shaft. The shaft should be about halfway into the coupler. Tighten the grub screws onto the motor shaft.

Attach the Motor Mount printed part onto the motor using two M3x12mm screws. Rectangular holes face away from the motor, see orientation below. I've opted to put a motor damper in between the motor and the printed part just so it runs quieter.

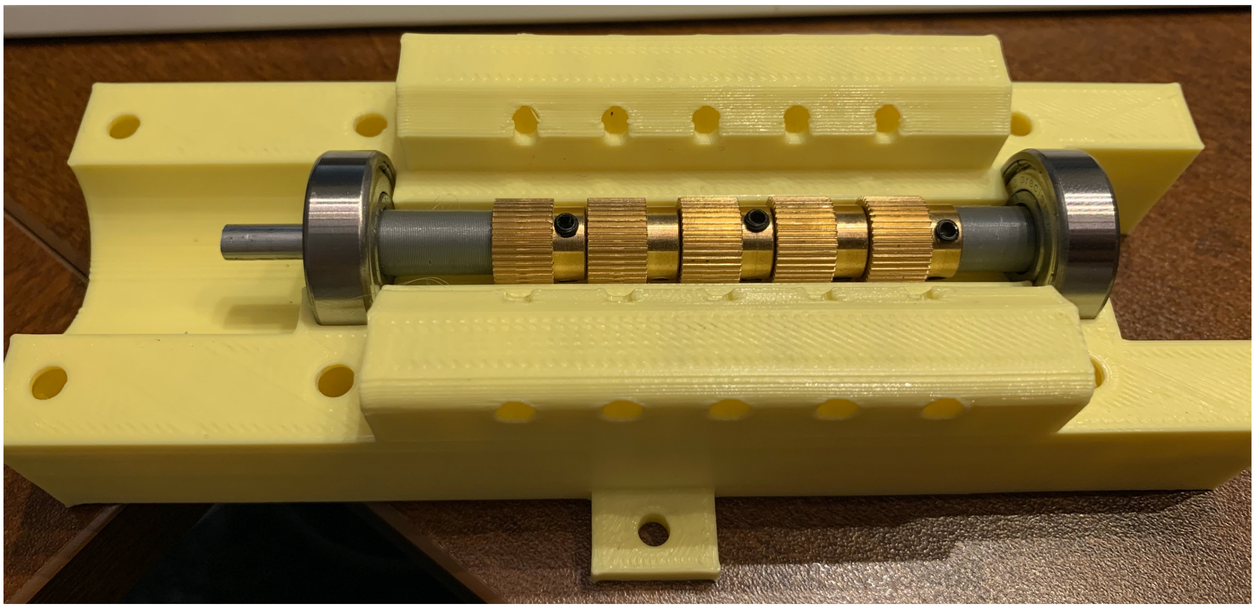


Insert two bearings into the Main Body printed part. Insert the Gear Shaft Spacers into the bearings. Orientation is not important in this step, but must be noted in the next step.

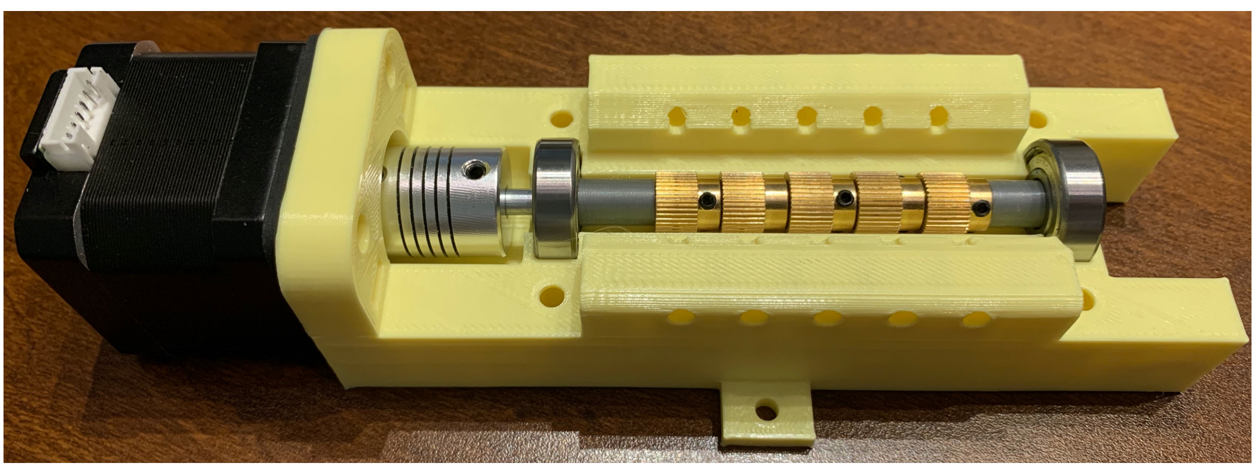


Place the gears in between the bearings. The teeth of the gears need to line up with the filament holes, so the orientation of the Gear Shaft Spacers needs to coincide with the orientation of the gears. Note that the tooth-side of the gears are facing towards the larger Gear Shaft Spacer and the grub-screw-side of the gears are facing towards the smaller Gear Shaft Spacer.

Insert the 5mm x 105mm long rod into the bearings and the gears. Tighten the grub screws so the gears stay in place. You may want to grind/sand down the rod a bit to flatten the edge, so the grub screws have more surface area to grab and so they don’t come loose later.

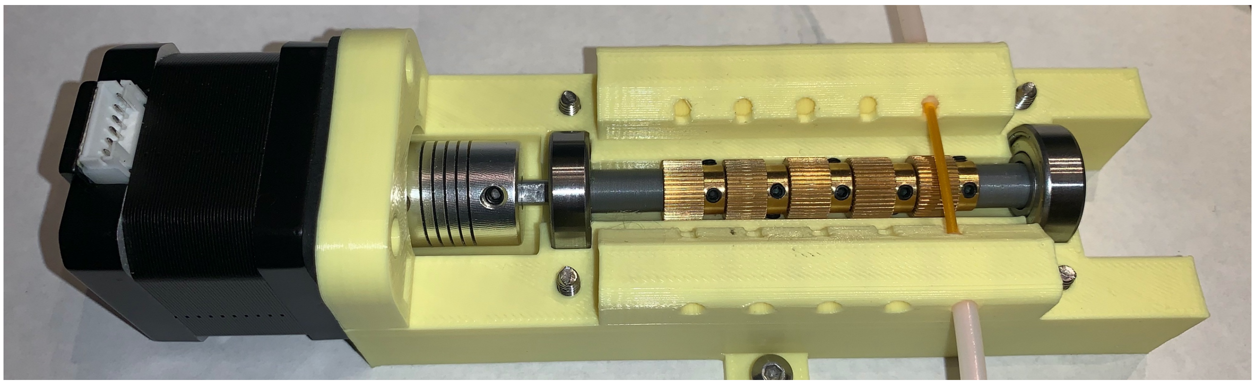


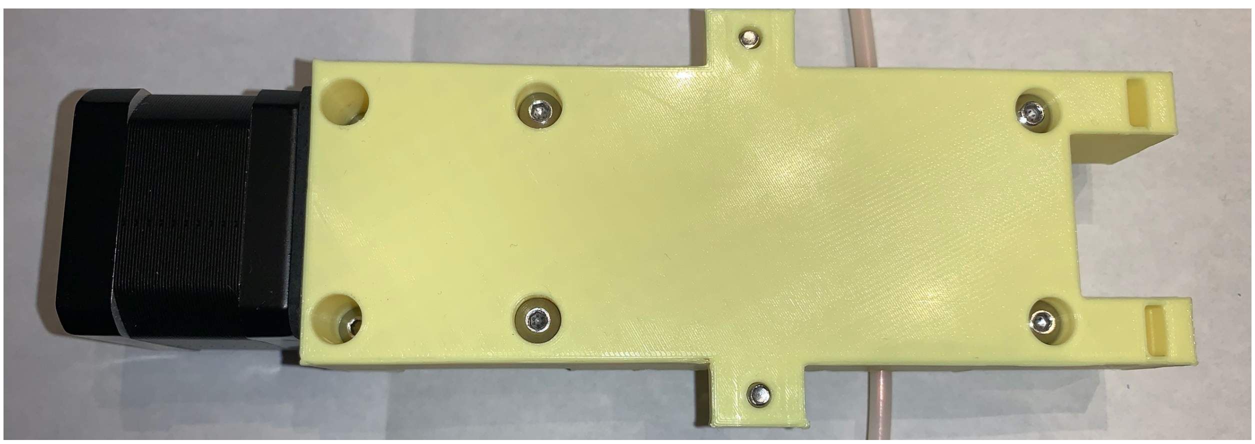
Slide the motor assembly onto the main body assembly, and make sure the 5mm rod enters the coupler properly. Tighten the coupler onto the gear shaft rod.



Insert two M4 nuts into the rectangular holes on the Motor Mount. Insert the two M4x12mm screws into the bottom of the main body (far left holes in the second pic below) and tighten these screws so they engage the nuts in the Motor Mount.

Slide the TPU springs onto four of the M4x16mm screws. Insert them into the centre four holes in the Main Body.

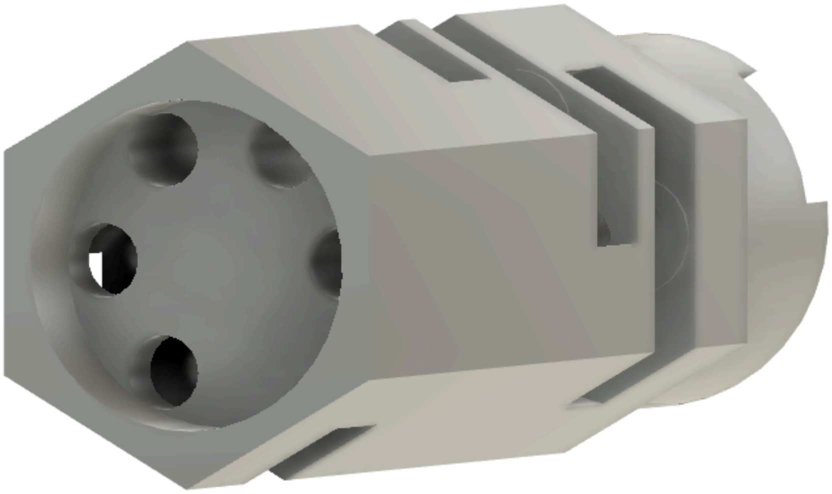


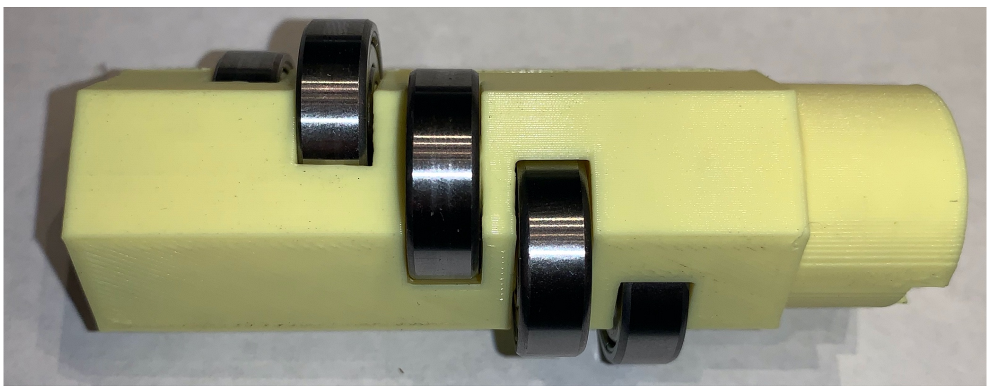


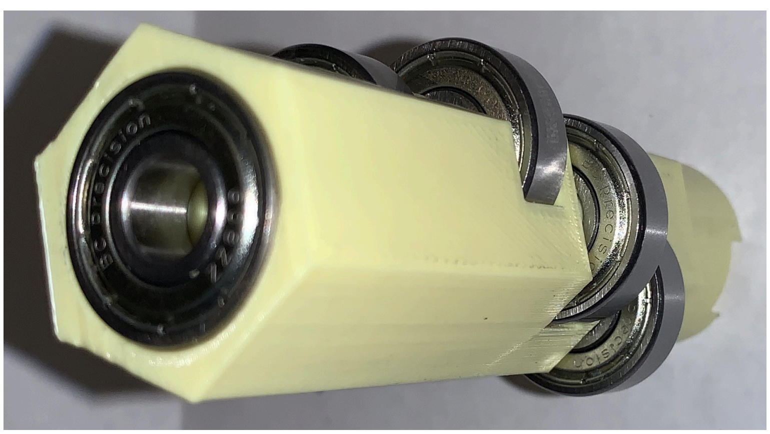
Insert the five Idler Bearing Spacers into five of the bearings.

There are five bearing slots in the Idler. Insert the first bearing into the position furthest from the side with five holes. In the second picture below, start by inserting the first bearing into the slot furthest to the right. Insert one of the 5mm x 12.9mm long rods into the hole of the Idler; push it all the way back so it engages the first bearing.

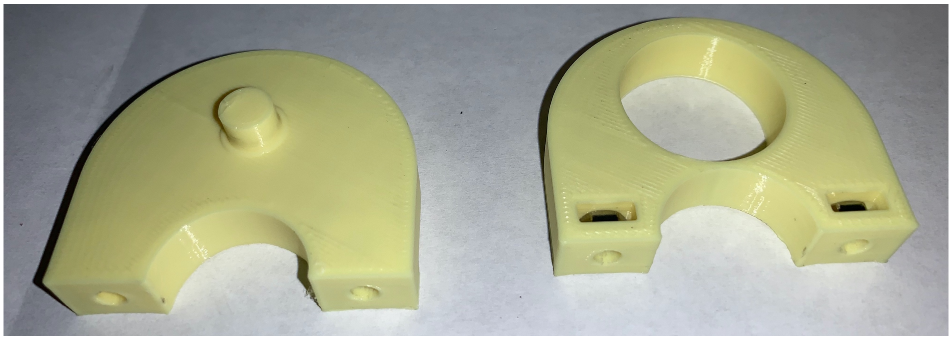
Repeat for all bearings, moving from right to left (if you're following the pics below). Be careful not to insert a rod or bearing out of order; I have not found a way to undo this other than cutting apart the whole idler and reprinting it. Once the final bearing and the final rod are in place, insert the sixth bearing into the end of the idler, sealing the five holes (left side of pic 3 below).

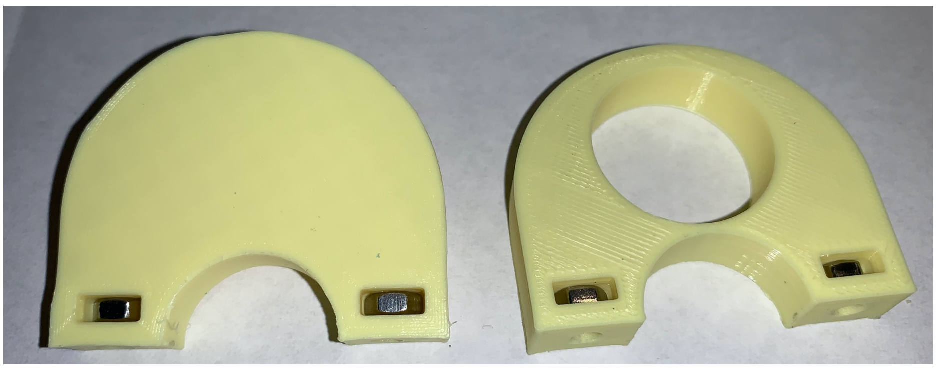




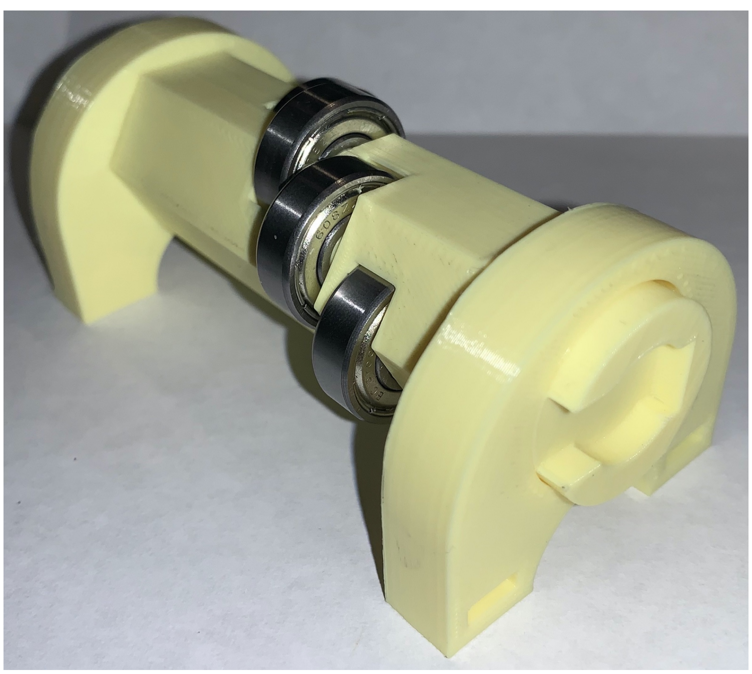


The Idler Pin and the Idler Hole both have rectangular holes. Insert four M4 nuts into these holes.

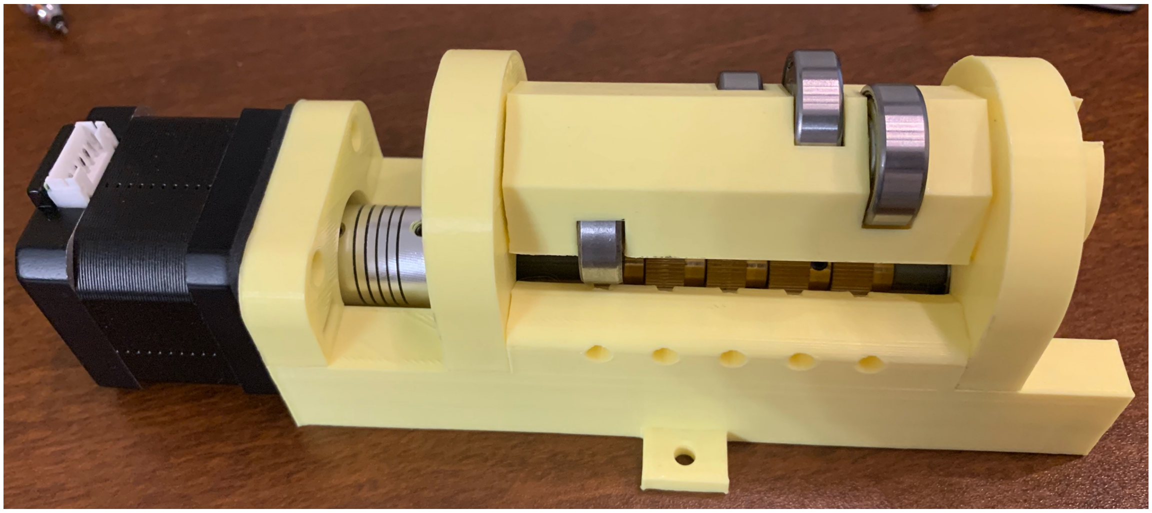




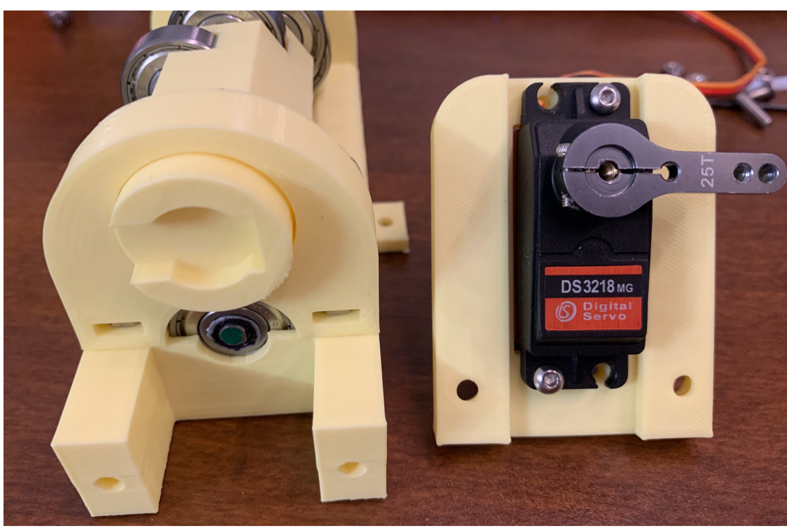
Mount the idler to the pin and hole. Make sure the rectangular holes face away from each other.

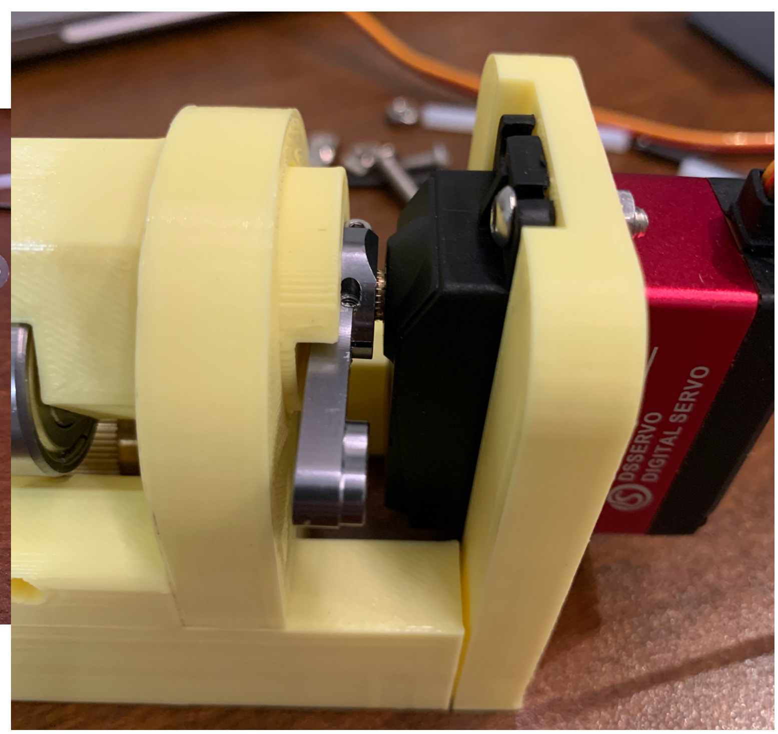


Mount the Idler assembly to the Main Body assembly. Tighten the M4x16mm screws until the nuts on the idler mounts are engaged, but do not over tighten them. There needs to be some flex for the assembly to self-adjust tension on the filament. This tension can be manually adjusted later.



Mount the servo to the Servo Mount. Plug your servo into your printer's mainboard. Set the servo angle to 0º. Tighten the servo horn onto the servo facing in the appropriate direction of rotation. The pictures below show the 0º starting position I use. Make sure the servo horn lines up to the idler position. Insert two M4 nuts into the bottom of the Main Body. Insert two M4x16mm screws into the Servo Mount and tighten this to the Main Body.





This marks the end of the official assembly. Optionally, you can mount the entire MCM to your printer or another area. I've used some t-nuts to mount it directly to the SK-Go's extrusion.

The remaining steps are to:

- Insert the PTFE fittings into the Bowden splitter
- Cut lengths of PTFE tube to suit your printer and spool locations
- Plug the whole thing into your printer and see if it works

This is currently designed to use a 20kg servo in order to overcome any friction. You should probably still sand down some areas if there is too much friction, and you can probably still get away with a weaker servo.

I'm still fairly new to CAD, and this whole project is still extremely Beta, so I can't really provide much support and customization. Hopefully this guide is sufficient, and I can try to answer some questions in the Seckit Facebook group. Frankly, if I can learn enough CAD to design this project, then I'm sure you can too, and hopefully you can make any necessary adaptations to suit your own printer. Best of luck!

GCode

As noted earlier, most of the colour-changing commands are handled by gcode. Below is the gcode script I use to change colours; you can read more about these commands [here on the Marlin Gcode page](). You probably don't need all these M400 commands, but I found that excluding them sometimes means the stepper motors will start moving before the servo has moved to the correct position. Another important note, for some reason if you include an identical feed rate gcode (eg F600) in two sequential G1 commands, the second G1 command is completely omitted from the output gcode file. Not sure why. To avoid this, assign different feed rates to each line of G1 code, or do not include an F command in the second G1 command if it should already be the same feed rate as the previous move anyway.

```
; tool change start
G1 E-1 F2400 ; retract before moving
G1 X60 Y-20 4000; park toolhead off of print bed

G1 E6 F600 ; purge nozzle

;cooling and tip shaping moves
G1 E-36 F1600 ; filament is pulled rapidly out of the nozzle into the cooling tube (a piece of
PTFE tube up inside the hotend heatsink)
G1 E-25 ; shape the tip of the filament while it is still soft
G1 E25
G1 E-25
G1 E25

G1 E-75 ; retract DD Nozzle Extruder completely until the filament is ejected
M400

M280 P0 S{(previous_extruder+1)*60} ; set servo position of previous colour
M163 S0 P0
M163 S1 P1; choose MCM Extruder
M164
M400

G1 E-195 F1000 ; retract filament {previous_extruder+1} on MCM Extruder (note earlier on
distinct E factors not working. Gcode says E-195 but it's actually retracting about 800mm)
M400

M280 P0 S{(next_extruder+1)*60} ; set servo position of next colour
M400

G1 E193 ; forward filament {next_extruder+1} on MCM Extruder
M400

; select both extruders simultaneously
M163 S0 P0.79 ; DD nozzle extruder 79%
M163 S1 P0.21 ; MCM extruder 21%
M164
```

```
M400

G1 E60 ; forward filament to engage Nozzle Extruder
M400

M163 S0 P1.0
M163 S1 P0.0 ; disengage MCM Extruder
M164
M280 P0 S0 ; disengage idler
M400

G1 E100 F400; purge nozzle to avoid colour bleed
M400

G1 X10 Y-22 F4000; wipe sequence across nozzle brush
M400
; tool change end
```

As mentioned earlier, some post-processing of the output gcode file is necessary:

- Remove T* commands
- PrusaSlicer puts a toolchange gcode sequence at the start of the file, immediately after the print start gcode sequence. Make sure you manually remove this toolchange script in the gcode file. There’s probably a better way to address this, but my workaround seems fine as long as the initial colour is already loaded in the nozzle.