Motorized Camera Slider MK3

Created by Ruiz Brothers

https://learn.adafruit.com/motorized-camera-slider-mk3

Last updated on 2024-03-08 02:19:44 PM EST

Table of Contents

Overview 5

- Motorized Camera Slider
- The Upgrade
- Prerequisite Guides
- Electronic Components
- Hardware
- Fasteners & Joining

Software 8

- Uploading Sketches
- Advanced Controller Motor Shield Sketch
- Simple Controller Motor Shield Sketch

Circuit Diagram 12

3D Printing 12

- What If I Don't Have A 3D Printer?
- 3D Printed Parts
- Tolerances
- Dry Fit Testing
- Design Source
- Slice Settings

Stepper Motor Assembly 15

- Pre-fasten Motor Mount
- Secure Stepper Motor to Mount
- Install Motor Mount Support
- Secure Motor Mount Support
- Install Rail Mount to Motor Mount
- Secure Rail Mount
- Motor Subassembly

Bearing Mount Assembly 17

- Install Mount Support
- Install Rail Mount
- Bearing Mount Subassembly

Tripod Mount Assembly 18

- Install Tripod Screw
- Install T-Nuts to Aluminum Extrusions
- Tripod Mount Pre-Install
- Secure Tripod Mount

Extrusion Assembly 19

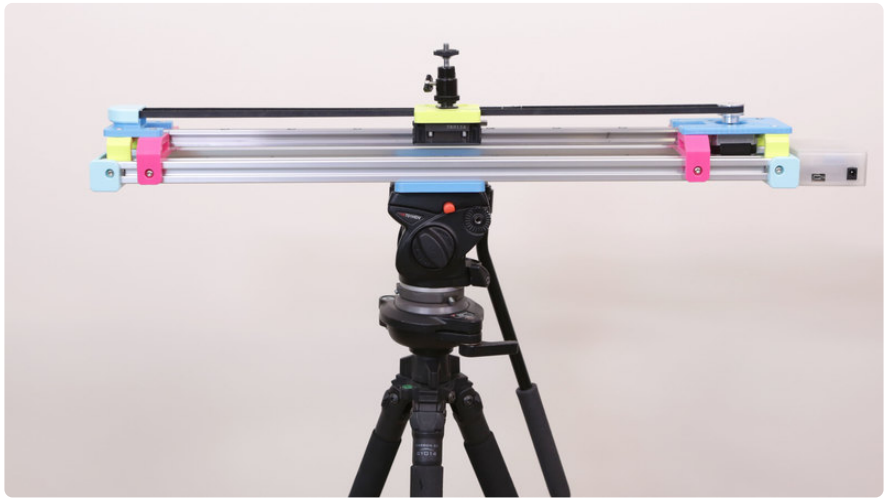
- Install Side T-Nuts
- Install Bearing Mount
- Insert Side Screws
- Install Bearing Bar Mount
- Secure Bar Mount
- More T-Nuts
- Install Motor Mount Assembly

Case Assembly 22
• Attach Case to Mount
• Secure Case to Mount
• Thread Motor Wires
• Install Metro
• Wiring Motor to Shield
• Motor Shield Wiring
• Install Motor Shield
• Secure Case Assembly
Camera Mount Assembly 25
• Install D-Ring
• Install Camera Mount
• Secure Camera Mount
Rail Assembly 26
• Install Slide Rail
• Secure Rail to Motor Assembly
Belt Assembly 28
• Install Bearing to Mount
• Install Belt to Pulley
• Install Belt to Camera Mount
• Install Belt to Bearing
• Install Bearing Cover
• Belt Tensioner
Final Assembly 30
• Motor SubAssembly
• Bearing SubAssembly
• Tripod Mount SubAssembly
Powering 31
• 12V Power
• Wall Adapter
• Portability

Overview

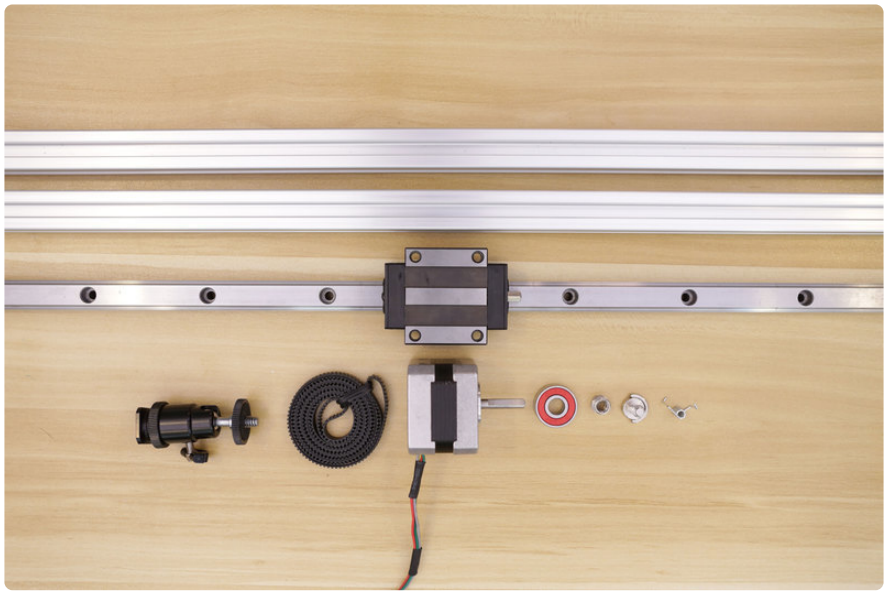
Motorized Camera Slider

This project builds on top of the original bluetooth motorized camera slider (https://adafru.it/Chn). Most of the electronics and hardware components are repurposed in this upgrade and uses the exact same code and software.



The Upgrade

The majority of the update is mechanical. The slider performs much smoother and operates with less friction in the timing belt. The entire slider assembly can now be mounted to a tripod! Two 2020 aluminum extrusions support the linear bearing rail allowing the assembly to be mounted to a tripod. Redesigned mounts reduces the amount of friction caused by the timing belt, allowing much smoother motion.



Prerequisite Guides

I suggest walking through the following guides to get a better understanding of the electronics.

- Adafruit Guide to Excellent Soldering (https://adafru.it/drl)
- Arduino Motor Shield V2 for Arduino (https://adafru.it/doc)
- Adafruit Bluefruit LE SPI Guide (https://adafru.it/iqb)
- Bluetooth Controlled Motorized Camera Slider (MK2) (https://adafru.it/Chn)

Electronic Components

The Adafruit Metro, Motor Shield, BLE Module and Stepper Motor are the main electronic components used in this project.

1 x Adafruit METRO 328 https://www.adafruit.com/product/2488
with Headers - ATmega328

---

1 x Adafruit Motor Shield https://www.adafruit.com/product/1438
Adafruit Motor/Stepper/Servo Shield for Arduino v2 Kit - v2.3

---

1 x BLE Module https://www.adafruit.com/product/2633
Adafruit Bluefruit LE SPI Friend - Bluetooth Low Energy

Hardware

There's a lot of hardware components such as the sliding rail, linear bearing, ball bearing, timing belt, extrusions, etc.

1 x Linear Bearing Supported Slide Rail https://www.adafruit.com/product/1861
Linear Bearing Supported Slide Rail - 15mm wide - 500mm long

---

1 x Linear Bearing Pillow Block https://www.adafruit.com/product/1866
15mm Diameter - Wider Version

---

2 x Slotted Aluminum Extrusion https://www.adafruit.com/product/1221
Slotted Aluminum Extrusion - 20mm x 20mm - 610mm long

6mm Belt - 36 Tooth - 5mm Bore

1 x Aluminum GT2 Timing Pulley — https://www.adafruit.com/product/1253
6mm Belt - 36 Tooth - 5mm Bore

---

1 x Ball Bearing — https://www.adafruit.com/product/1178
Radial Ball Bearing 608ZZ - Set of 4

---

1 x Timing Belt GT2 — https://www.adafruit.com/product/1184
Profile - 2mm pitch - 6mm wide 1164mm long

Fasteners & Joining

Handful of machine screws, hex nuts, t-nuts, etc.

1 x M3 Screws — https://www.albanycountyfasteners.com/Phillips-Pan-Head-Machine-Screw-M3-x-5-p/1066-1008.htm
M3 x .5 x 4M (for Adafruit Metro) | M3 x .5 x 6M (for Stepper Motor)

---

1 x M4 Screws — https://www.adafruit.com/product/1160
Button Hex Machine Screw - M4 thread - 8mm long - pack of 50 (for various mounts)

---

1 x M4 Screws (Longer) — https://www.albanycountyfasteners.com/Phillips-Pan-Head-Machine-Screw-M4-x-7-p/1066-1009.htm
M4 x .7 x 18M for various mounts

---

10 x M4 Hex Nuts — https://www.albanycountyfasteners.com/Metric-Hex-Jam-Nuts-A2-Stainless-Steel-p/5580000.htm
M4 x .7 (1.95M thick, 7.0M flat)

---

4 x M5 Screws — https://www.albanycountyfasteners.com/Phillips-Pan-Head-Machine-Screw-M4-x-7-p/1066-1010.htm
M5 x .8 x 16M for the Pillow Block

---

1 x T-NUT FOR 20X20 - M4 THREAD — https://www.adafruit.com/product/1158
Aluminum Extrusion Oval T-Nut for 20x20 - M4 Thread - pack of 50

---

1 x Camera and Tripod — https://www.adafruit.com/product/2392
3/8" to 1/4" Adapter Screw

Cameras / Tripods / Photo / Video

1 x 1/4" Screw with D-Ring
Cameras / Tripods / Photo / Video
https://www.adafruit.com/product/2629

1 x Swivel-Head Pan Tilt (PTZ) Shoe Mount Adapter
https://www.adafruit.com/product/2464

1 x GT2 Timing Belt Torsion Spring 6mm Width Belt Pack of 10
https://www.amazon.com/gp/product/B01E913IJK/

Software

To use the controller motorshield sketch you'll want to make sure you're using the latest version of the Arduino IDE (https://adafru.it/fvm) (1.6.5 at the time of this writing).

If you're totally new to Arduino take a little time to go through some introductory tutorials like how to make a LED blink (https://adafru.it/iqd). This will help you understand how to use the IDE, load a sketch, and upload code.

Next you'll need to make sure the libraries used by the sketch are installed. With the latest Arduino IDE you can use its library manager (https://adafru.it/fCN) to easily install libraries, or check out this guide on how to manually install a library (https://adafru.it/dNR). You'll want to install the following libraries:

- Adafruit Motor Shield V2 Library
- Adafruit Adafruit BluefruitLE nRF51

Search for the libraries in the library manager and they should be easy to find and install:

If you already have one or more of these libraries installed then make sure to update it to the latest version.

Uploading Sketches

To load the sketch make sure the libraries above are installed, and the Arduino is connected to the computer through a USB cable. Make sure under the **Tools -> Board** menu the **Arduino Uno** is selected, and under the **Tools -> Port** menu the serial port for the Arduino is selected (it should say **Arduino Uno**).

Then press the upload button or click the **Sketch -> Upload** item to send the code to the Arduino. Woo-hoo the sketch should be running!

Advanced Controller Motor Shield Sketch

This sketch features several functions for moving the slider. You can set speed and choose linear or easing movements.

```
// Smartphone- or tablet-activated timelapse camera slider.
// Uses the following Adafruit parts:
//
// Arduino Uno R3 or similar (adafruit.com/product/50 or #2488)
// Bluefruit LE SPI Friend (#2633)
// Motor/Stepper/Servo Shield v2 (#1438)
// NEMA-17 Stepper motor (#324)
// Miscellaneous hardware and 3D-printed parts; see guide for full list.
//
// Needs Adafruit_BluefruitLE_nRF51 and Adafruit_MotorShield libs:
// github.com/adafruit
// Use Adafruit Bluefruit LE app for iOS or Android to control timing.
// Buttons 1-4 select interpolation mode (linear vs. various ease in/out).
// Up/down select speed.  Left = home.  Right = start slider.

#include &lt;SPI.h&gt;
#include &lt;Wire.h&gt;
#include &lt;SoftwareSerial.h&gt;
#include &lt;Adafruit_BluefruitLE_SPI.h&gt;
#include &lt;Adafruit_MotorShield.h&gt;

#define LED 13 // Built-in LED on pin 13

// Bluefruit config -----------------------------------------------------
Adafruit_BluefruitLE_SPI ble(8, 7, 6); // CS, IRQ, RST pins

// Stepper motor config -------------------------------------------------
#define STEPPER_STEPS 1100 // Length of slider
#define STEPPER_RPM     20

Adafruit_MotorShield AFMS = Adafruit_MotorShield();

// Stepper motor w/200 steps/revolution (1.8 degree) on port 2 (M3 &amp;amp; M4)
Adafruit_StepperMotor *motor = AFMS.getStepper(200, 2);

// Global stuff ---------------------------------------------------------

// Four motion interpolation modes are supported (corresponding to buttons
// 1-4 in the Bluefruit LE app for iOS &amp;amp; Android): linear is constant speed,
// ease_in_out accelerates/decelerates at the ends (fastest in middle),
// ease_in accelerates (fastest at end) and ease_out decelerates (fastest
// at beginning):
typedef enum interp { linear, ease_in_out, ease_in, ease_out };

// A few different time periods are supported, but not a whole lot.  Since
// there's no display connected, we just blink the onboard LED to indicate
// which setting is active, and there's only so many speed changes one can
// reliably 'read' this way.  Feel free to edit, but keep in mind there are
// upper and lower limits to the time interval -- the stepper can only move
// so fast and if you try to press beyond that it'll just move linearly
// regardless of the selected interpolation mode, and the micros() function
// rolls over about every 70 minutes, so the duration can't exceed that.
// Blink rate is similarly constrained due to BLE comm; about 2 Hz max.
struct {
  uint32_t totalTime;    // Total duration of movement along slider
```

```
  uint32_t LEDflashTime; // LED blink rate indicates selected speed
} speed[] = {
   5 * 60 * 1000000L, 1000000L / 2, //  5 min slide,   2 Hz blink
  10 * 60 * 1000000L, 1000000L,     // 10 min slide,   1 Hz blink
  20 * 60 * 1000000L, 1000000L * 2, // 20 min slide, 1/2 Hz blink
  60 * 60 * 1000000L, 1000000L * 3  // 60 min slide, 1/3 Hz blink
};
#define N_SPEEDS (sizeof(speed) / sizeof(speed[0]))

uint32_t startTime = 0;      // micros() value when slider movement started
interp   mode      = linear; // Selected interpolation mode
uint8_t  speedIdx  = 0;      // Selected speed index (0 to N_SPEEDS-1)
boolean  moving    = false;  // True if motor active &amp;amp; interpolating

// Setup function - runs once at startup ----------------------------------

void setup(void) {
  Serial.begin(57600);
  pinMode(LED, OUTPUT);
  digitalWrite(LED, HIGH);       // LED steady on during init
  if(!ble.begin(false)) for(;;); // BLE init error? LED on forever
  ble.echo(false);
  AFMS.begin();
  motor-&gt;setSpeed(STEPPER_RPM);
  motor-&gt;release();              // Allow manual positioning at start
  digitalWrite(LED, LOW);        // LED off = successful init
}

// Loop function - repeats forever ----------------------------------------

void loop(void) {
  static uint16_t prevSliderPos = 0; // Slider position on prior call
  static uint8_t  led;               // Blink status on/off

  if(moving) { // Motor currently engaged?
    float p, t = (float)(micros() - startTime) /
                 (float)speed[speedIdx].totalTime; // Time 0.0 - 1.0

    switch(mode) { // Cubic easing functions from http://gizma.com/easing/
     case linear:
      p = (float)STEPPER_STEPS * t;
      break;
     case ease_in_out:
      t *= 2.0;
      if(t &lt; 1.0) {
        p  = (float)STEPPER_STEPS * 0.5 * t * t * t;
      } else {
        t -= 2.0;
        p  = (float)STEPPER_STEPS * 0.5 * (t * t * t + 2.0);
      }
      break;
     case ease_in:
      p = (float)STEPPER_STEPS * t * t * t;
      break;
     case ease_out:
      t -= 1.0;
      p  = (float)STEPPER_STEPS * (t * t * t + 1.0);
      break;
    }

    // Move stepper to new position (if required)
    uint16_t sliderPos = (int)(p + 0.5);
    if(sliderPos &gt; prevSliderPos) {
      // Microstepping is used to reduce vibration
      motor-&gt;step(sliderPos - prevSliderPos, FORWARD, MICROSTEP);
      prevSliderPos = sliderPos;
      if(sliderPos &gt;= STEPPER_STEPS) { // At end of motion?
        motor-&gt;release();             // Turn off motor
        moving = false;
```

```
      }
    }
  } else {
    if((micros() - startTime) &gt; (speed[speedIdx].LEDflashTime / 2)) {
      digitalWrite(LED, led++ &amp; 1);
      startTime = micros();
    }
  }

  // Process any pending Bluetooth input
  if(ble.isConnected()) {
    ble.println(F("AT+BLEUARTRX"));     // Request string from BLE module
    ble.readline();                     // Read outcome
    if(!strncmp(ble.buffer, "!B", 2) &amp;&amp; // Controller button command
       checkCRC(255-'!'-'B', 4)      &amp;&amp; // Verify checksum
       (ble.buffer[3] == '1')) {        // Button press? 1=press 0=release
      switch(ble.buffer[2]) {
       case '1': if(!moving) mode = linear;      break;
       case '2': if(!moving) mode = ease_in_out; break;
       case '3': if(!moving) mode = ease_in;     break;
       case '4': if(!moving) mode = ease_out;    break;
       case '5': // Up (faster)
        if((!moving) &amp;&amp; speedIdx) speedIdx--;
        break;
       case '6': // Down (slower)
        if((!moving) &amp;&amp; (speedIdx &lt; (N_SPEEDS-1))) speedIdx++;
        break;
       case '7': // Left (home)
        digitalWrite(LED, LOW);
        motor-&gt;step(prevSliderPos, BACKWARD, DOUBLE);
        motor-&gt;release();
        prevSliderPos = 0;
        moving        = false;
        break;
       case '8': // Right (start/pause)
        if(!moving) { // Don't double-start
          digitalWrite(LED, LOW);
          moving    = true;
          startTime = micros();
        }
        break;
      }
    }
  }
}

boolean checkCRC(uint8_t sum, uint8_t CRCindex) {
  for(uint8_t i=2; i&lt;CRCindex; i++) sum -= (uint8_t)ble.buffer[i];
  return ((uint8_t)ble.buffer[CRCindex] == sum);
}
```

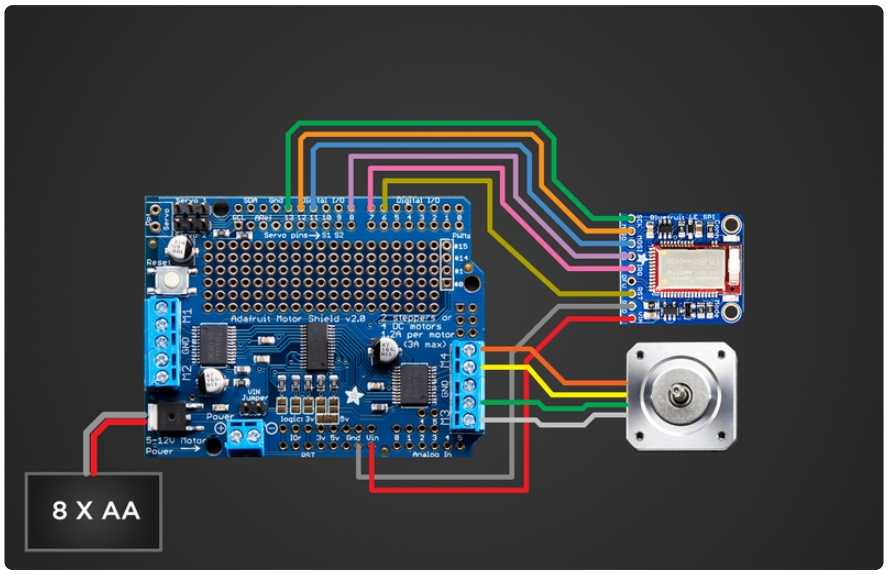
Simple Controller Motor Shield Sketch

This sketch do not feature any easing functions. It moves linearly and do not have any periodic stops. This sketch is ideally for heavier cameras that want to use double steps. The controls are easier to modify and understand.

simple-motorcontrol.zip

https://adafru.it/iqe

Circuit Diagram



To build the camera slider with the Adafruit Motor Shield V2 for Arduino, wire the components together as shown in the diagram above.

- Connect Arduino **VIN** to Bluefruit LE SPI **VIN pin**
- Connect Arduino **GND** to Bluefruit LE SPI **GND pin**
- Connect Arduino **D13** to Bluefruit LE SPI **SCK pin**
- Connect Arduino **D12** to Bluefruit LE SPI **MISO pin**
- Connect Arduino **D11** to Bluefruit LE SPI **MOSI pin**
- Connect Arduino **D8** to Bluefruit LE SPI **CS pin**
- Connect Arduino **D7** to Bluefruit LE SPI **IRQ pin**
- Connect Arduino **D6** to Bluefruit LE SPI **RST pin**

The diagram above shows the wiring for **NEMA-17** size stepper. Connect the four wires from the stepper to motor outputs on **M3** and **M4**.

An 8 x AA battery pack will supply the necessary 12V to the Arduino Uno via a 2.1mm barriel jack.

3D Printing

What If I Don't Have A 3D Printer?

Not to worry! You can use a 3D printing service such as 3DHubs (https://adafru.it/jNb) or MakeXYZ (https://adafru.it/veh) to have a local 3D printer operator 3D print

and ship you parts to you. This is a great way to get your parts 3D printed by local makers. You could also try checking out your local Library or search for a Maker Space.

3D Printed Parts

All of the parts are 3D printed with FDM type 3D printers using various colored filaments. I printed these parts in regular PLA, but it can be printed in ABS or other high-strength material if you need more strength. I recommend printing out the pieces individually, as opposed to printing them all together at once.

Download Fusion Archive

https://adafru.it/B2l

Download STLs from Thingiverse

https://adafru.it/B2m

Download STLs from Pinshape

https://adafru.it/B2n

Download STLs from YouMagine

https://adafru.it/B2o

Tolerances

Tolerances are subject to change depending on filament materials and slice settings. Parts that need to snap fit together might be too loose or tight. Mounting holes might be too large or small. Different types of 3D printers, nozzles, filament and slice settings are all things that can affect the tolerances.

Dry Fit Testing

I highly suggest testing out the 3D printed parts before going through the entire assembly. Here are some things to test.

- Does the screws and hex nuts fit into the mounting holes?
- Does the mount slide into the aluminum extrusion properly?
- Does the cover properly fit snap onto the case?

Most things that are too tight can easily be fixed with a little sanding using a filing tool. If holes are too loose, it might not matter since hex nuts will be used to tighten parts together.

Design Source

I designed the parts in Autodesk Fusion 360 and made the source available to download. It contains full parametric design history so it's easy to remix the design or repurpose it for a new project.

Slice Settings

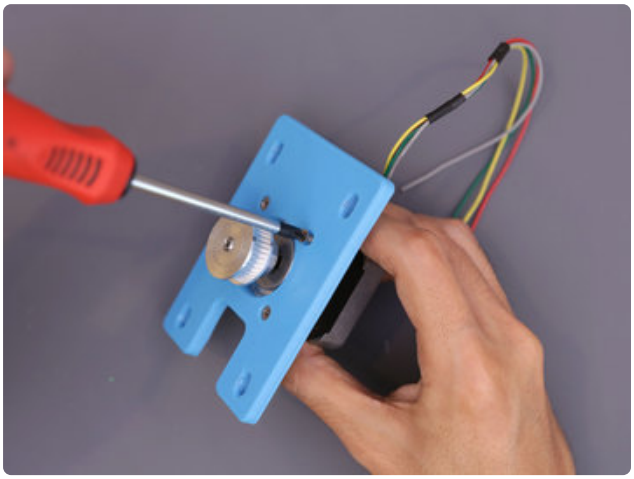
To get the most accurate tolerances for the snap fit features, you'll want to adjust your slice settings accordingly. The parts were tested with PLA filament, other materials that have more shrinkage such as ABS or Nylon may have different tolerances.

- 220C Extruder temperature
- 65C Bed temperature
- 0.2 layer height
- 0.35 line width (0.48 extruder width)
- 2 wall line count
- 25% infill
- 60mm/s default printing speed

Stepper Motor Assembly

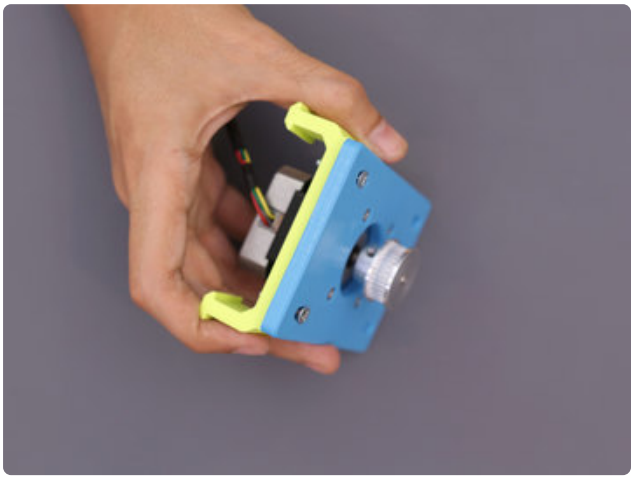
Pre-fasten Motor Mount

Fasten 4x M3 (6mm long) screws into the top of the motor mount – The top is the surface with the chamfered mounting holes. The screws should have a semi-loose tolerance.



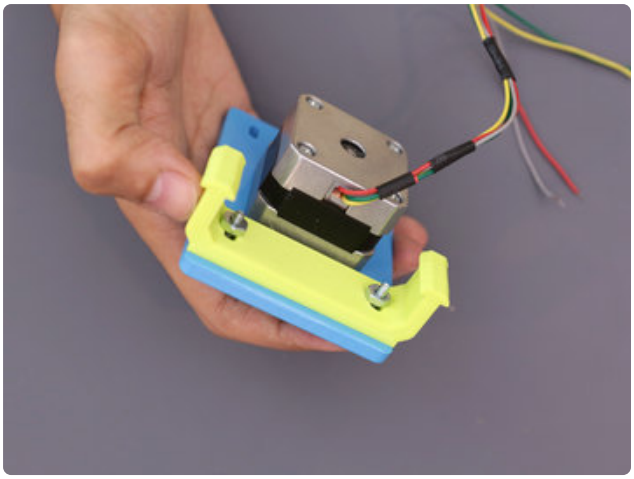
Secure Stepper Motor to Mount

Fit the mount over the stepper motor. Orient the motor so the wires are positioned like it is in the photo. Fasten the 4x M3 screws until fully tightened to secure the mount to the stepper motor.



Install Motor Mount Support

Grab the **slider-motor-support.stl** part and place it under the motor mount. Position the mount so the mounting holes line up to match the photo. Insert and fasten 2x M4 machine screws into the mounting holes.



Secure Motor Mount Support

While holding the support mount in place, insert 2x hex nuts and fasten until fully tightened. The mounting holes are slotted and adjustable for adding more tension to the belt.

Install Rail Mount to Motor Mount

The rail mount features a recessed area for a hex nut. Insert a hex nut and hold it in place. The hex nut is loose, so you'll need to hold it up right so it doesn't fall out while fastening to the motor mount.

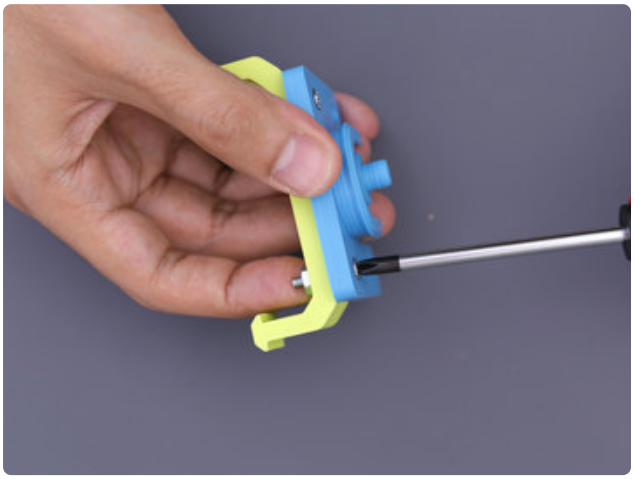
Secure Rail Mount

Place the rail mount onto the motor mount. Insert and fasten an M4 screw through the top of the motor mount. The screw should go through the two mounts with the hex nut securing them together. Repeat this process for the second mounting hole.

Motor Subassembly

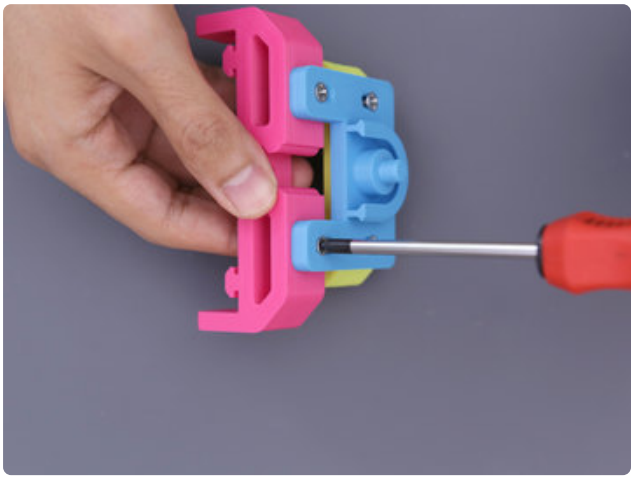
The stepper motor subassembly should look like this. Both the rail mount and the support mount feature slotted holes for adjusting the tension of the belt – This can be done later on the assembly.

Bearing Mount Assembly



Install Mount Support

The bearing mount assembly is similar to the motor assembly. They both need a rail mount and a support mount. Place the support mount under the bearing mount. Insert an M4 screw through the top of the bearing mount. Use an M4 hex nut to secure the two mounts together.



Install Rail Mount

Place the rail mount under the bearing mount. Install hex nuts into the recessed areas of the rail mount. Insert and fasten M4 screws to secure the bearing mount to the rail mount.

Bearing Mount Subassembly

The bearing mount should look like this. Again, the slotted mounting holes are there so you can adjust the belt tension later on in the build.

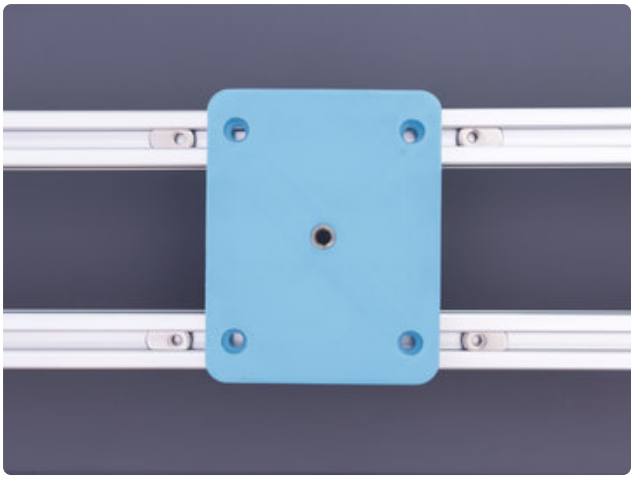
Tripod Mount Assembly

Install Tripod Screw

The tripod mount needs a 3/8" to 1/4" screw adapter so it can be secured to a camera tripod. Insert and fasten the tripod screw adapter to the center of the tripod mount. Use a large flat screwdriver to fully tighten the screw.

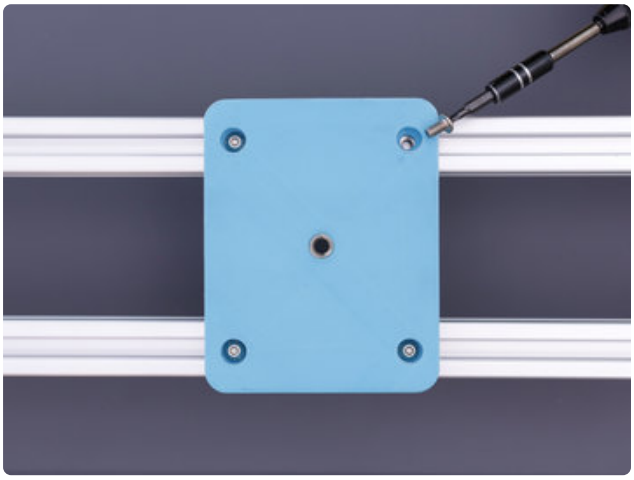
Install T-Nuts to Aluminum Extrusions

The oval T-Nuts are designed to be slotted into the aluminum extrusion. Grab both aluminum extrusion bars and insert two t-nuts. These will be used to secure the tripod mount to the aluminum bars.



Tripod Mount Pre-Install

Position the two aluminum extrusion bars so the tripod mount can be rested over them. Orient the parts so the mounting holes are positioned over the extrusion bars. Slide the T-Nuts so the mounting holes line up with the ones in the Tripod Mount.



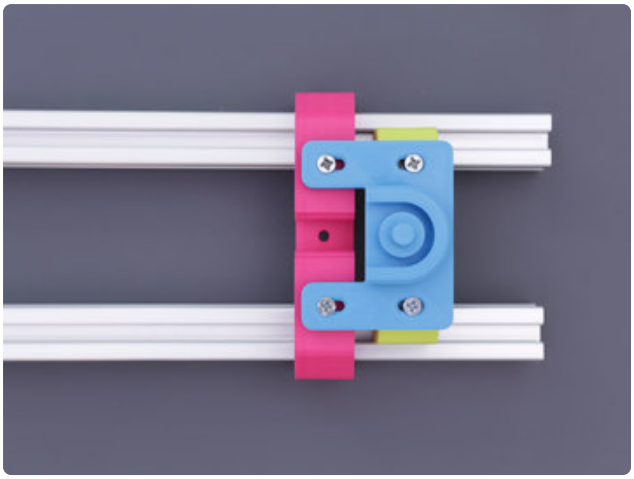
Secure Tripod Mount

Insert and fasten 4x M4 screws to fix the Tripod Mount to the two aluminum extrusion bars. Do not fully tighten the screws, leave them loose. The M4 screw should be threaded through the T-Nuts. The tripod mount should be able to slide along the slotted aluminum extrusions. We will fully tighten the screws later on in the build.

Extrusion Assembly

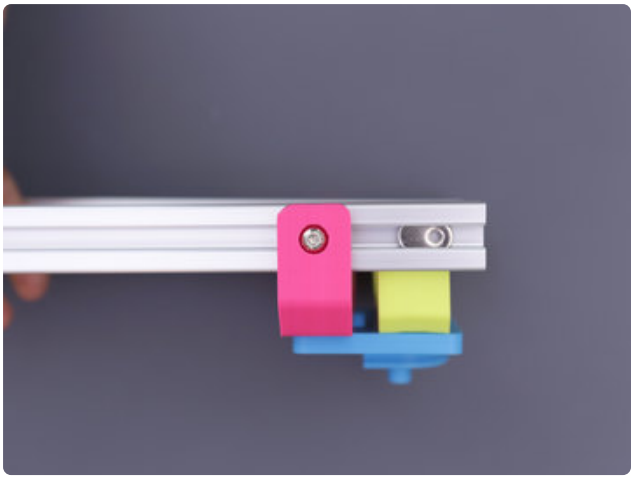
Install Side T-Nuts

Insert two T-Nuts to both of the sides of the two aluminum extrusion bars. Make sure to follow the orientation of the T-Nuts and reference the photo.



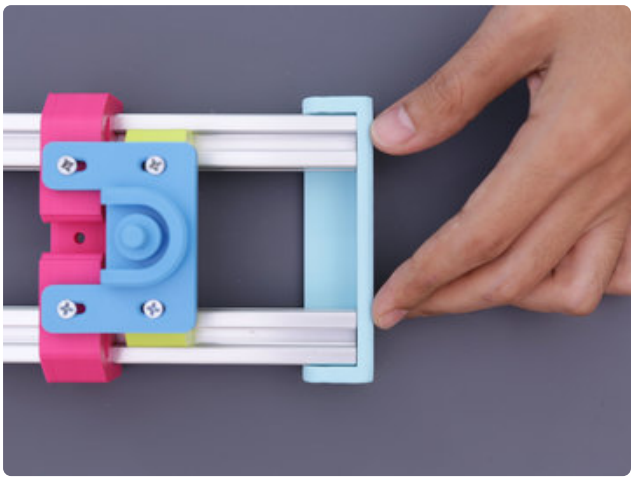
Install Bearing Mount

Grab the bearing mount assembly and slide the T-Shaped nubs through the slotted aluminum extrusions. This should be installed to the opposite side of the tripod mount. The bearing mount assembly should be able to freely slide through the slotted aluminum extrusion.



Insert Side Screws

Slide the bearing mount assembly so the mounting holes line up with the T-Nuts. Insert and fasten M4 screws to secure the mount to the T-Nuts. Leave them loose – The assembly need to be able to slide along the t-slotted aluminum extrusions.



Install Bearing Bar Mount

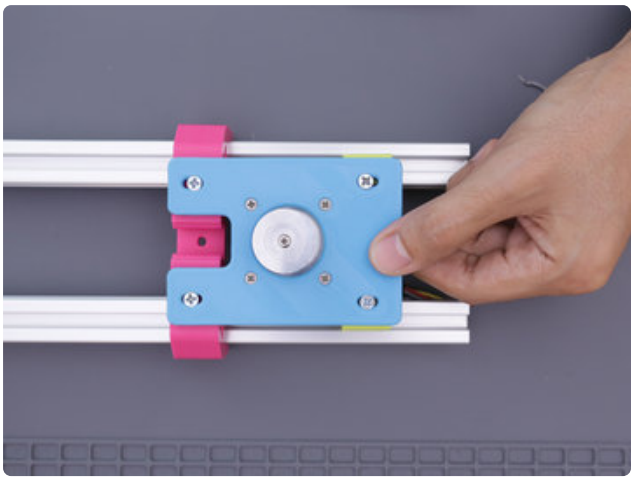
Grab the bearing bar mount and fit it onto the end cap of the two aluminum extrusions. The bar mount features t-shaped nubs that should slot into the aluminum extrusions.

Secure Bar Mount

Slide the remaining T-Nut so it's line up with the mounting hole in the bar mount. Insert and fasten M4 screws to fix the bar mount to the T-Nuts. Again, you'll want to leave them a bit loose so we can adjust them later in the build.

More T-Nuts

Insert two more T-Nuts on to the other side of the aluminum extrusions. These will secure the Motor Mount assembly to the extrusion bars. Again, make sure the orientation matches the photo.



Install Motor Mount Assembly

Insert and slide the motor mount assembly onto the two slotted aluminum extrusions. Both the rail mount and the support mount should be able to fitted and slide along the slotted aluminum extrusion bars.

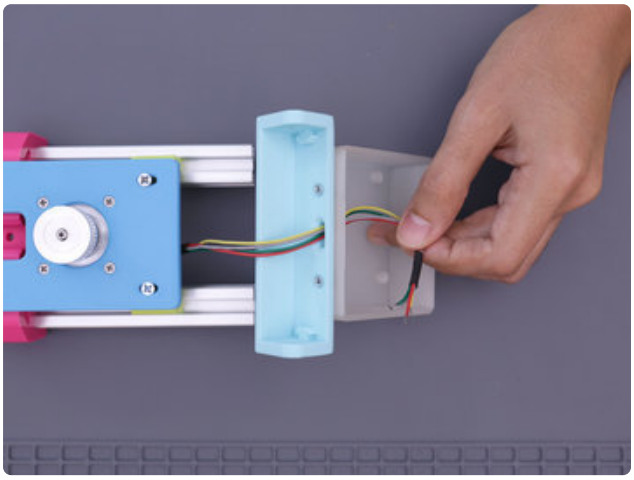
Case Assembly

Attach Case to Mount

Grab the case and motor bar mount parts. Orient the parts so the mounting holes line up with each other. Insert and fasten two M4 x 16mm screws with hex nuts into to join the two parts together.

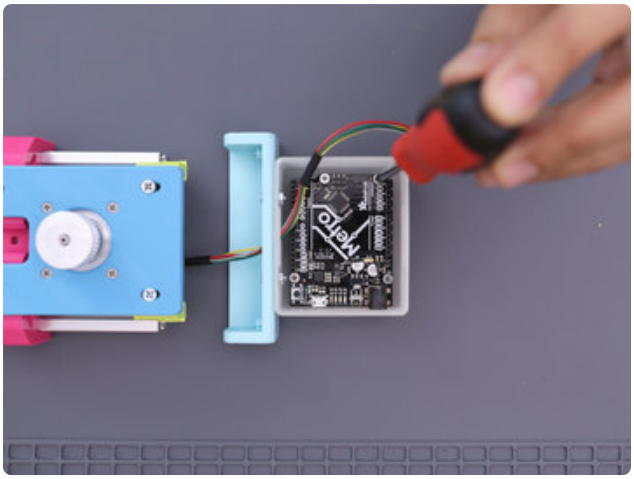
Secure Case to Mount

Hold the hex nuts in place while fastening the screws to secure them tightly.



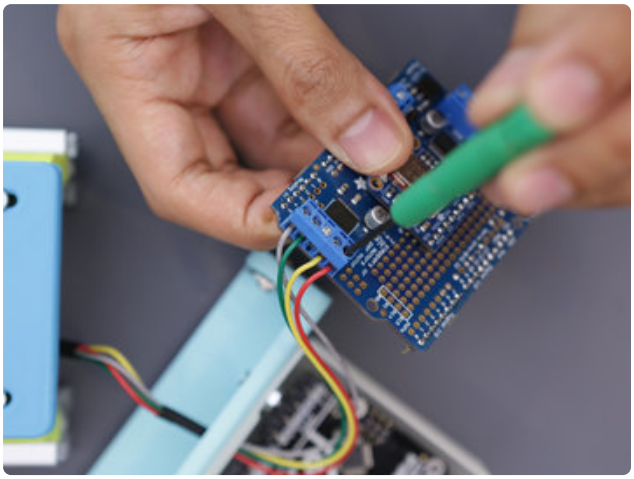
Thread Motor Wires

A piece of heat shrink tubing can help keep the wires from the stepper motor bundled together. Thread the stepper motor wires through the center hole in the case mount assembly.



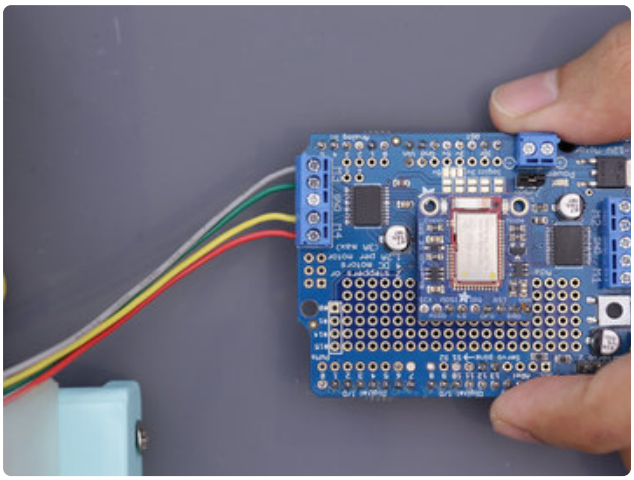
Install Metro

Place the Adafruit Metro into the case. Orient the PCB so the mounting holes line up with the standoffs. Insert and fasten four M3 x 4mm screws to secure the Adafruit Metro to the case.



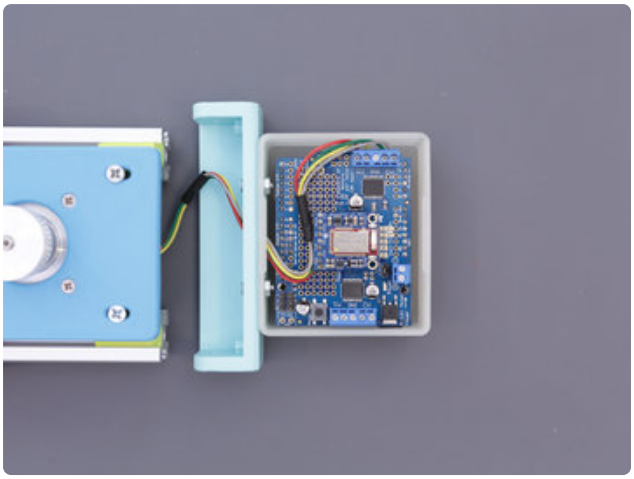
Wiring Motor to Shield

Grab the wires from the stepper motor and the Adafruit Motor Shield. Insert the pre-tinned wires into the screw block terminals with the M3 and M4 labels. Use a small screw driver to fasten the screw block terminals tight.



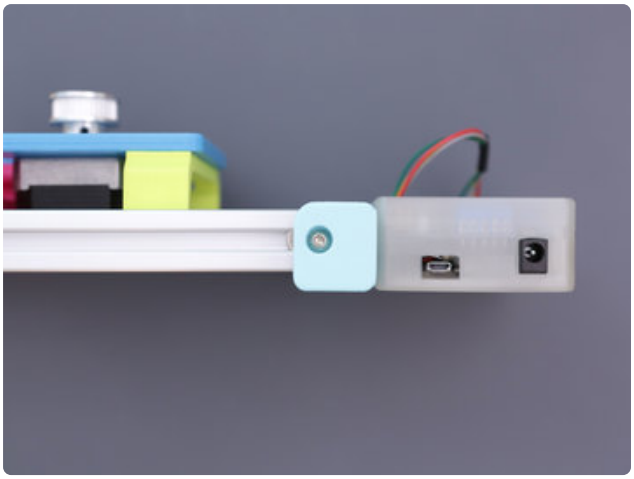
Motor Shield Wiring

Follow the photo to reference the correct wiring. Double check your work to ensure the wiring is installed correctly.



Install Motor Shield

Position the motor shield over the Adafruit Metro so the header pins are lined up and matching. Carefully press the motor shield down on top of the Adafruit Metro and fully seat the header pins.



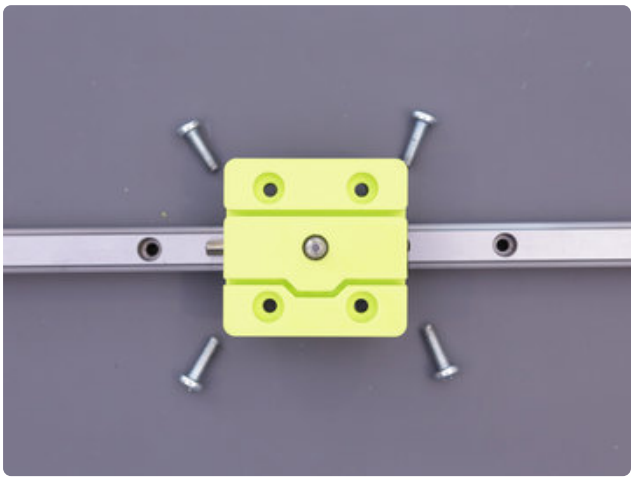
Secure Case Assembly

Install the motor bar mount to the two aluminum extrusions. The t-shaped nubs should slot into the sides of the aluminum extrusion. Slide the oval T-Nuts so the mounting holes line up with the bar mount. Insert and fasten M4 screws to secure the bar mount to the aluminum extrusion.

Camera Mount Assembly

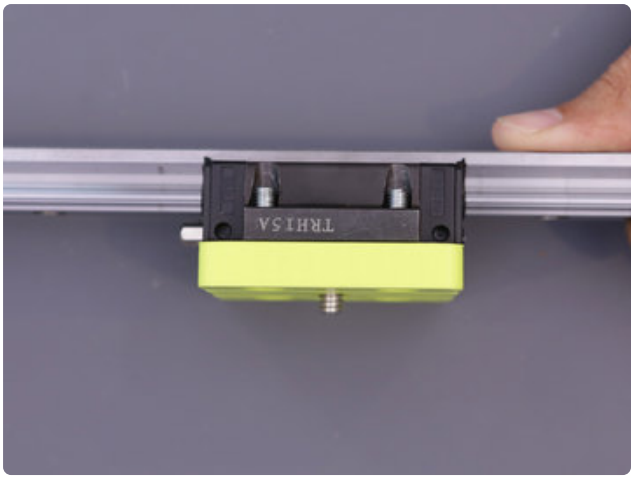
Install D-Ring

Grab the D-Ring tripod screw and insert it into the camera mount. The screw will be very loose and fall out easily. Hold the screw in place while installing it to the Pillow Block.



Install Camera Mount

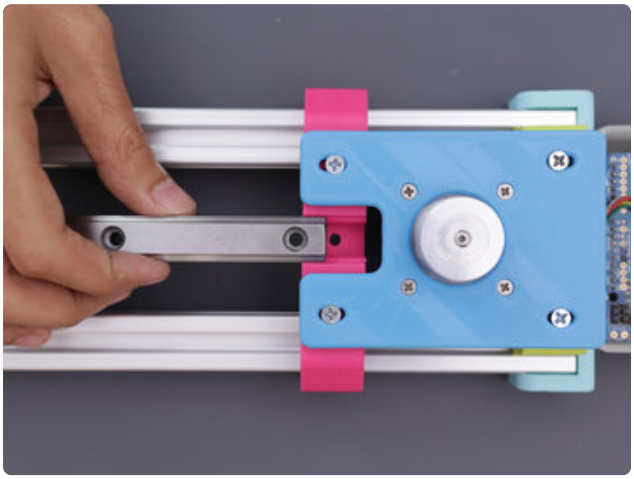
While holding the tripod screw, place the camera mount on top of the linear bearing pillow block. Orient the mount so the mounting holes line up. M5 x 16mm long screws will secure the camera mount to the linear bearing pillow block.

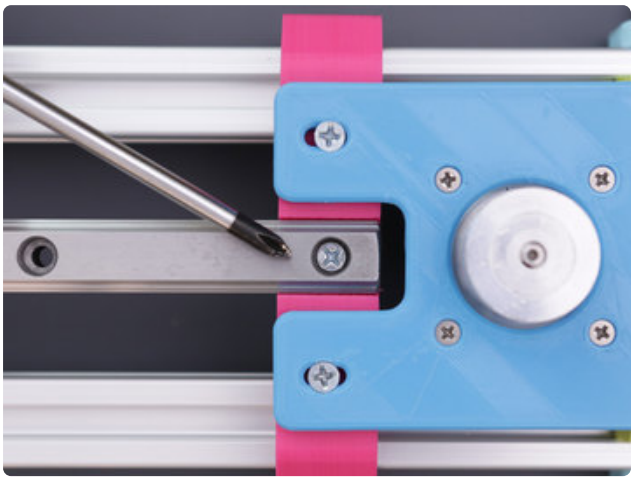


Secure Camera Mount

Insert and fasten the M5 screws into the mounting holes to secure the camera mount to the pillow block. Fully tighten the secures to join the two parts together. The mounting holes in the pillow block is threaded, so it doesn't need hex nuts.

Rail Assembly



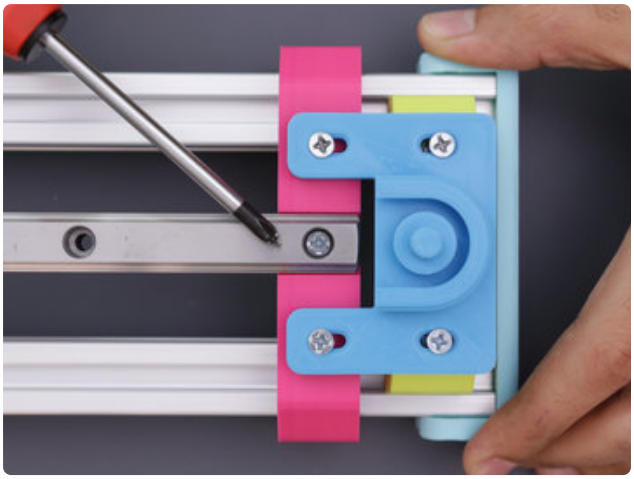


Install Slide Rail

Grab the linear rail and slide it into the motor mount assembly. The rail should slot into the motor rail mount loosely. The bearing assembly should be able to slide out of the extrusion bars to allow space for the linear rail. Line up the mounting hole on the rail with the mount. Insert an M4 x 18mm long screw into the mounting hole.

Secure Rail to Motor Assembly

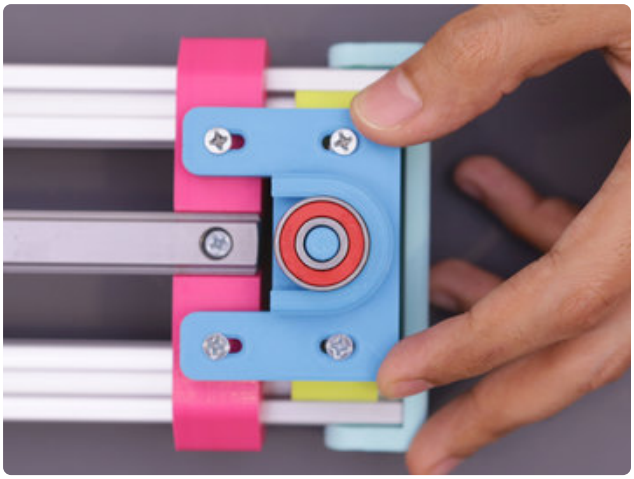
Flip the entire assembly over and place a hex nut onto the threading of the M4 screw. Fasten the screw and hex nut to tightly secure the parts together.



Secure Rail to Bearing Assembly

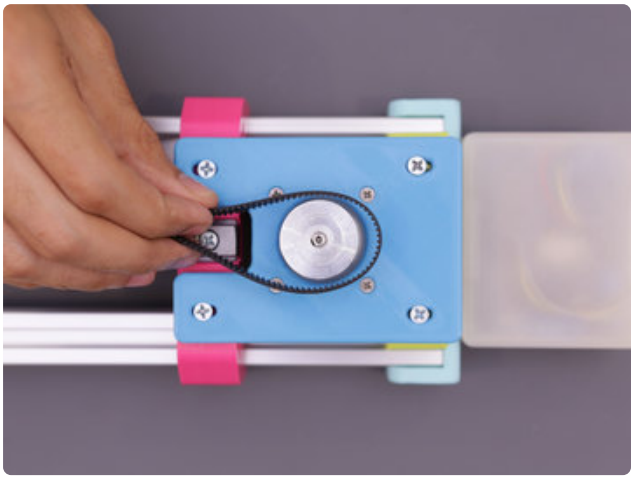
Following a similar process, slide the rail onto the bearing mount assembly. Line up the mounting holes and insert another M4 x 18mm long screw. Use a hex nut to secure the rail to the rest of the assembly.

Belt Assembly



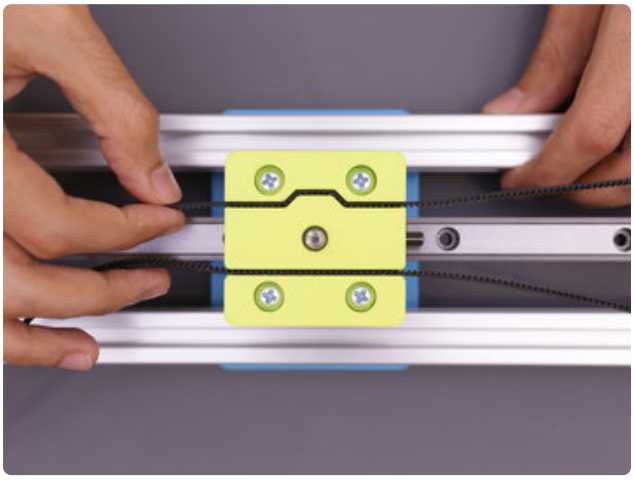
Install Bearing to Mount

Place the radial ball bearing over the post in the bearing mount.



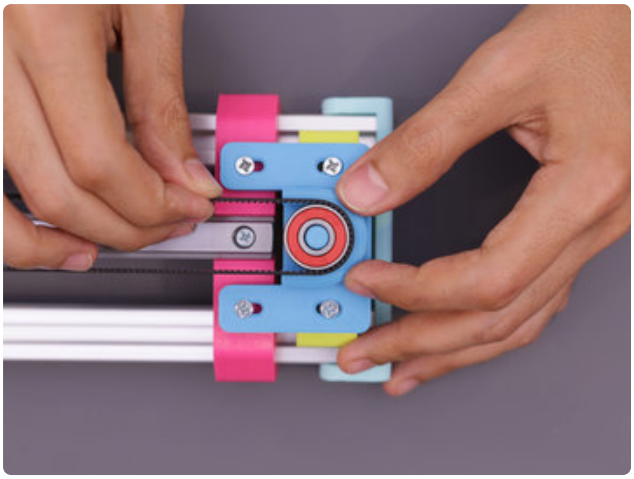
Install Belt to Pulley

Grab the timing belt and place it over the pulley of the stepper motor. The teeth in the belt should grab onto the teeth of the timing pulley.



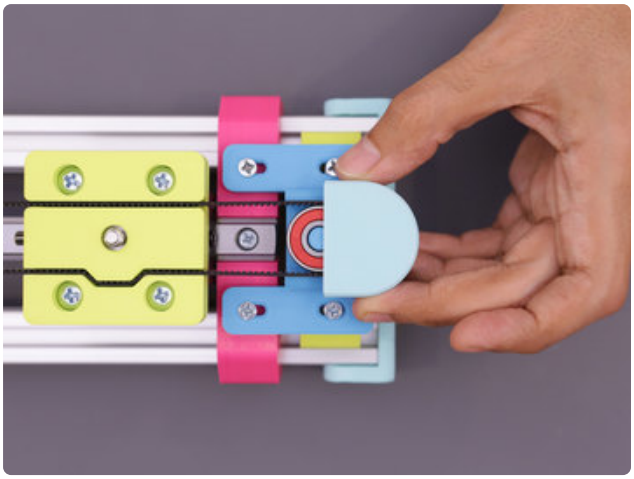
Install Belt to Camera Mount

Work the timing belt into the notches of the camera mount. Press the belt into the slotted features of the camera mount to secure it in place.



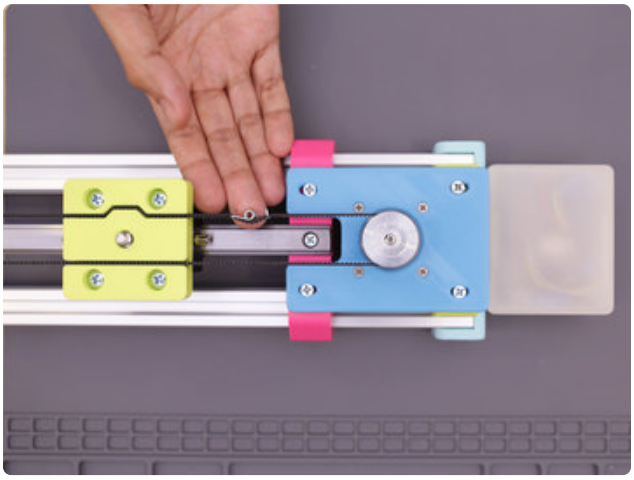
Install Belt to Bearing

Grab the other end of the timing belt and work it on top of the radial ball bearing. Pull the belt and press it down so it grabs onto the ball bearing.



Install Bearing Cover

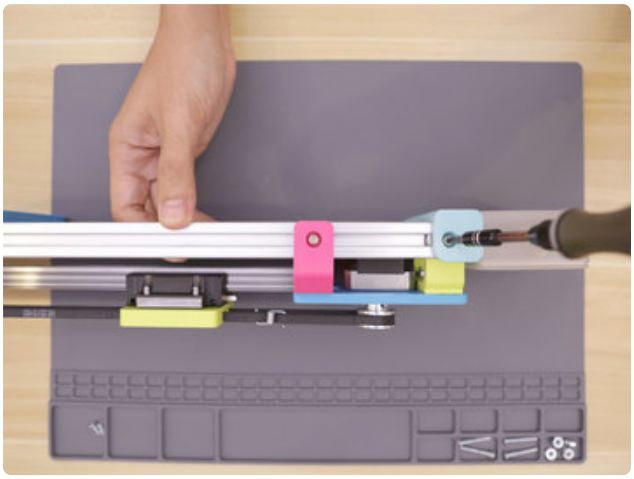
The bearing cover slides into the grove on the edge of the bearing mount. This rests in place and doesn't need any further securing. The tolerances will be loose or tight, depending on the printed part. That should not effect the performance.



Belt Tensioner

You can optionally install a belt tensioner (https://adafru.it/B2p) to tighten the timing belt.

Final Assembly



Motor SubAssembly

Ensure the timing belt is tight. Proceed to tighten all of the screws in the motor assembly.

Bearing SubAssembly

Hold the entire assembly while tightening all of the screws in the bearing assembly.

Tripod Mount SubAssembly

Slide the tripod mount so it's in the center of the sliding rail. Then, proceed to tighten all of the screws.

Powering

12V Power

To properly power the stepper motor, you need at least 12V. A USB battery pack typically is only 5V and will not provide enough juice to push the weight of a camera (be it a lightweight GoPro or Mobile Phone). I highly suggest using either a 12V power wall adapter or 12V battery pack.

Wall Adapter

I mostly use the camera slider indoors. Having it plugged into the wall is nice because I don't have to worry about batteries getting low and dying. This 12V DC 1000mA power adapter (http://adafru.it/798) works great with this project.

Portability

If you plan to use the slider outdoors, I suggest using an 8 x AA battery power pack (http://adafru.it/875). This features an on/off switch and 2.1mm barrel jack that plugs directly into the Adafruit Metro.