

# DIY Space Mouse for Fusion 360 Using Magnets

By salimbenbouz in CircuitsSensors

69,782

599

100

Featured

Download

Favorite



By salimbenbouz

[Links](#)

[Follow](#)

More by  
the author:



About: Designer and Maker of things 🎨 Electronics and 3D printing Tinkerer [More About salimbenbouz »](#)

[Fusion 360 Projects »](#)

This project originated from my curiosity about CAD input devices. I realized that I was struggling to remember navigation combos in Fusion 360 whenever I switched workstations. Having never used a Space mouse before, I decided to explore the possibility of creating a similar device using only the parts I already had on hand.

To make the 3-axis joystick, I used a low-resolution magnetometer and neodymium magnets mounted on a spring system directly inspired from the real 3D connexion Space Mouse. Currently, the device emulates the mouse and keyboard to execute the orbit and pan functions within Fusion 360. There are also two shortcut buttons — one for the home view and another for fitting to the screen.

To address the lack of a native option to assign a keyboard shortcut for views in Fusion 360, we will go over a workaround that involves a custom python add-in.

[Add Tip](#)

[Ask Question](#)

[Comment](#)

[Download](#)

# Supplies

## Parts

- 1 × QT Py RP2040 [Adafruit](#)
- 1 × TLV493D Triple-Axis Magnetometer [Adafruit](#)
- 1 × STEMMA QT / Qwiic JST SH 4-pin Cable - 100mm Long [Adafruit](#)
- 2 × Tactile Button switch (6mm) [Adafruit](#)
- 3 × Adafruit LED Sequins - Emerald Green (optional) [Adafruit](#)
- 4 × Rubber Bumper Feet [Adafruit](#)

## Hardware

Round Magnets [Amazon](#)

6 × (6 × 2mm)

Springs [Amazon](#)

3 × Compression 23/64" × 1-3/8"

3 × Extension 13/64" × 13/16"

M2.5 Heat-Set Inserts [McMasterCarr](#)

24 × 3.4 mm

M2.5 Socket Head Screws [Amazon](#)

10 × 3mm

3 × 5mm

7 × 8mm

4 × 12mm

## 3D printing

PLA filament [Amazon](#)

0.177cal Steel BB's [Amazon](#)

## Finishing (optional)

Assorted Grits Sandpaper [Amazon](#)

Glazing and Spot Putty [Amazon](#)

Filler Primer [Amazon](#)

Spray Paint [Amazon](#)

Flat Matte Clear Coat [Amazon](#)

## Tools

3D Printer [Amazon](#)

Soldering Station/Iron. [Amazon](#)

Helping hands [Amazon](#)

Precision Screwdriver Set [Amazon](#)

Wire stripper [Amazon](#)

Sanding block

Flux, Solder

Hook up wire

Flush cutters

Hobby knife

Tweezers and pliers

*Affiliate links may be included in the parts list. I may receive a small commission at no additional cost to you.*

Add Tip

Ask Question

Comment

Download

## Step 1: Watch the Build Video

The video will give you an overview of the design and build process.

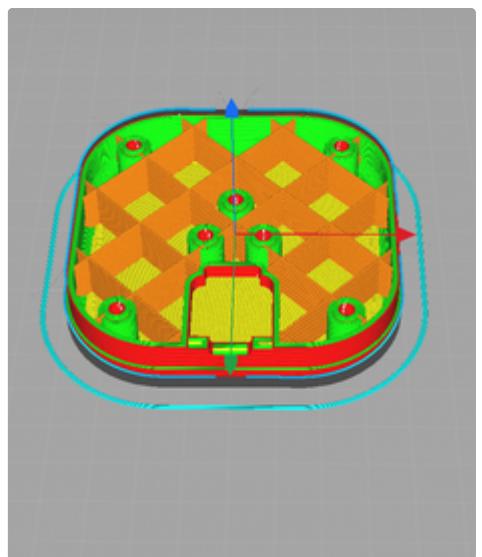
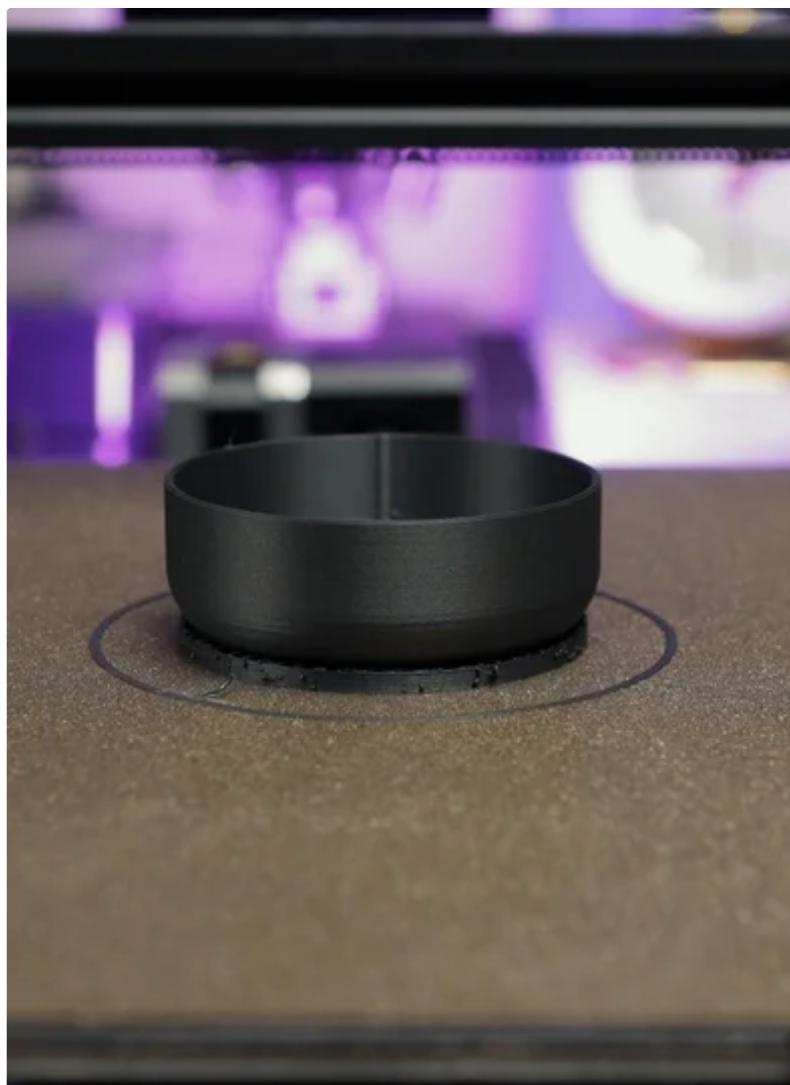
Add Tip

Ask Question

Comment

Download

## Step 2: 3D Printing



You can grab the included STLs in the page or from [Github](#)

**⚠ Few things to keep in mind before printing**

I have designed the holes and fit of the pieces to accommodate tolerances from my 3D printing setup. You might want to verify the heated inserts holes and the fit, particularly between the **Base\_top.stl** and the **Base\_bottom.stl** before proceeding.

For the **Base\_bottom.stl**, I printed it using a 5% grid infill with 3-4 wall counts. Around layer 64, I paused the print to pack in the steel BB's and add some weight. This step is essential to counteract the tension from the springs prevent the the device from sliding around.

I suggest to go for 100% infill solid parts from the knob and the three stem parts.

I included 1 model for the buttons but you obviously need to print it twice.

#### Space Mouse \_1

 <b>Base_bottom.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Base_middle_plate.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Base_top.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Buttons.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Knob.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Stem_bottom.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Stem_middle.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>
 <b>Stem_top.stl</b>	<a href="#">View in 3D</a>	<a href="#">Download</a>

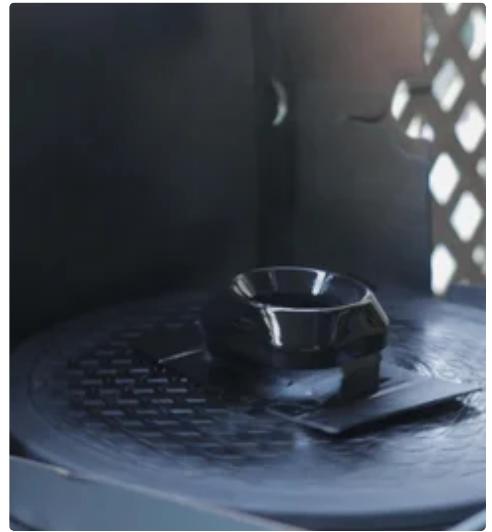
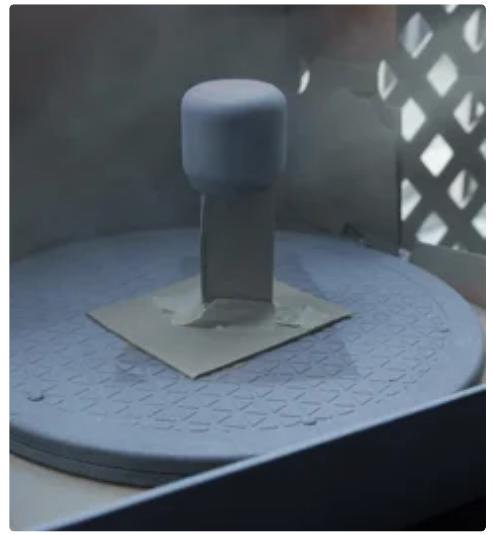
Add Tip

Ask Question

Comment

Download

### Step 3: Finishing (optional)



If you are not familiar with post-processing techniques for FDM prints, I have some instructions in this [project](#) to guide you.

Add Tip

Ask Question

Comment

Download

## Step 4: Heated Inserts



Once the clear coat has had ample time to dry, I proceeded to install the heated inserts into the designated 24 holes.

Although it may seem excessive for most cases, I prefer incorporating them into my 3D prints. I would rather do this than deal with stripped threads after few adjustments.

Add Tip

Ask Question

Comment

Download

## Step 5: Assembly \_1

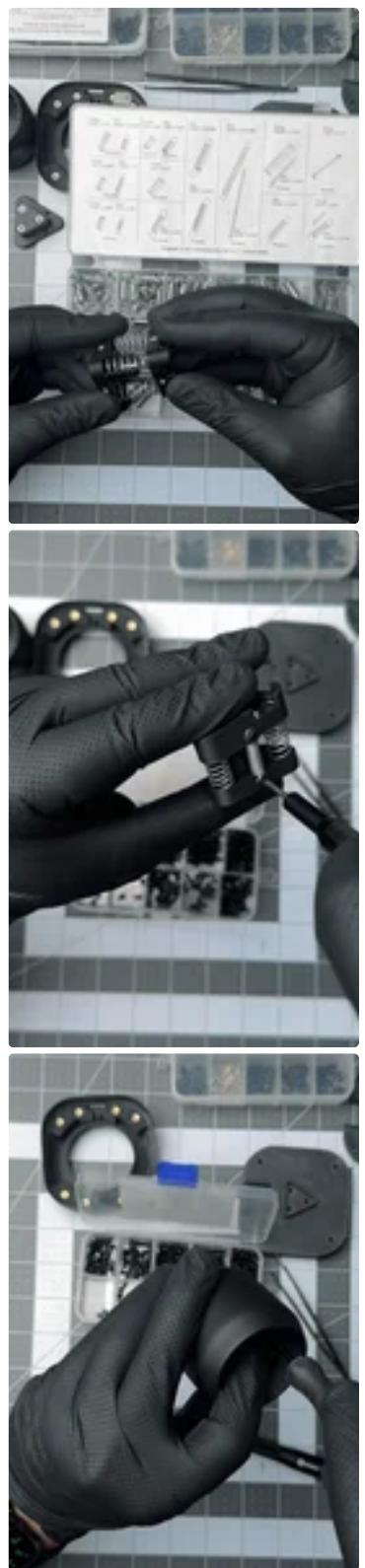


Following that, I proceeded to install six mini round magnets (6x2mm) into the designated slot on **Stem\_top.stl**.

Please note that depending on the tolerances of your 3D printer and the magnet manufacturer, you may need to apply some glue to secure them in place. In my case, I was able friction fit them.

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

## Step 6: Assembly \_2



Now, we can assemble the spring mechanism and attach it to the knob from the inside.

Begin by inserting the 3 compression springs into the appropriate holes on both **Stem\_top.stl** and **Stem\_middle.stl**. With both pieces compressed together, securely hold them in place while fastening the extension springs to the three sides using 6 x 3mm M2.5 socket head screws.

Once this step is complete, you can insert the entire contraption inside the knob and secure it from the inside using 3 x 5mm M2.5 screws. The **Stem\_middle.stl** features three larger holes, allowing you to pass through the springs and screw in the entire thing to the knob from inside.

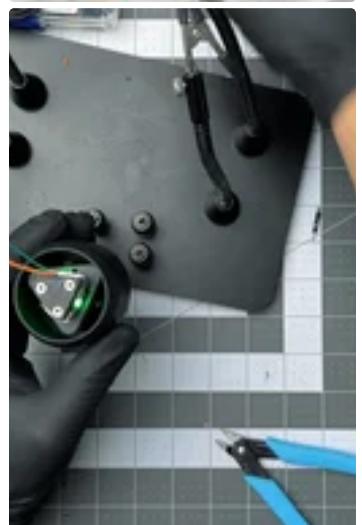
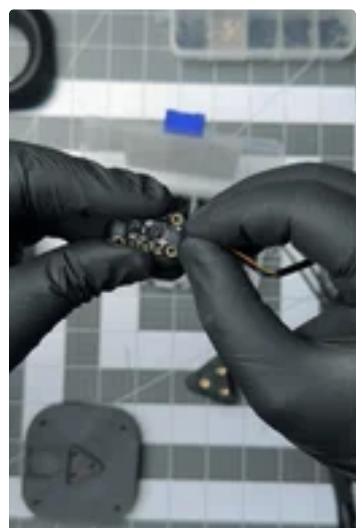
Add Tip

Ask Question

Comment

Download

### Step 7: Wiring \_1



I did not plan on this initially. However, it seemed to be a great opportunity to incorporate some LEDs for added visual interest.

I carefully glued and wired 3 Adafruit LED Sequins onto the sides of **Stem\_bottom.stl**. I took care to ensure that they did not interfere with the four screws or the lip protruding from **Base\_bottom.stl**, allowing for proper clearance.

Next, I plugged the Magnetometer and secured over it **Stem\_bottom.stl** to the inside of the knob using 4 x 8mm M2.5 screws.

#### ⚠ Check the wiring diagram



diy-spacemouse \_wiring diagram.pdf

[Download](#)

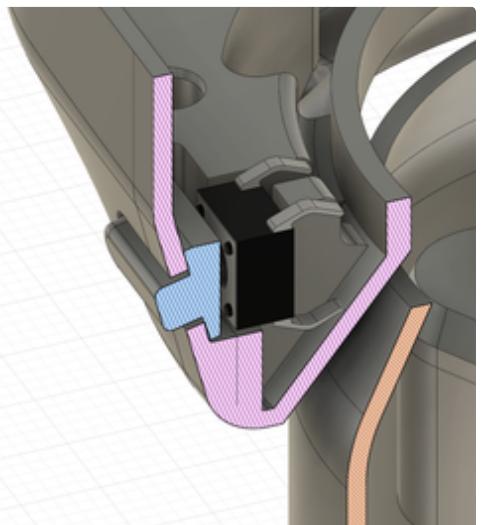
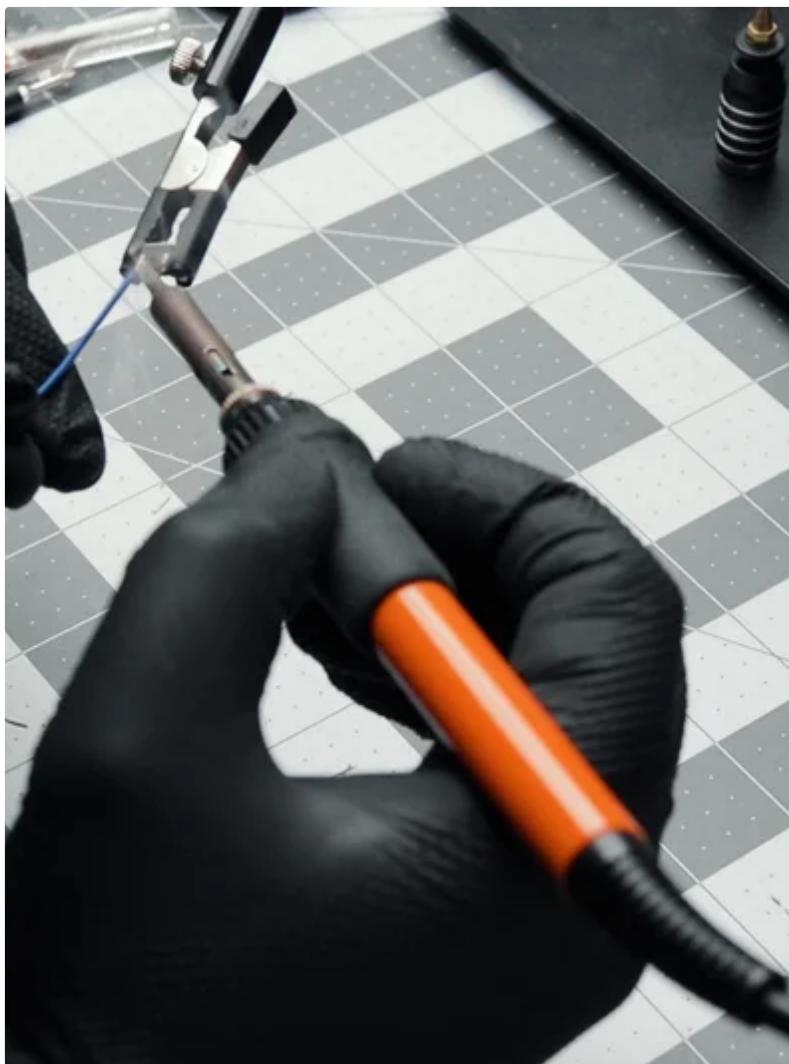
[Add Tip](#)

[Ask Question](#)

[Comment](#)

[Download](#)

### Step 8: Wiring \_2



[2 More Images](#)

I soldered wires to both switches and carefully measured the length of the shared wire (ground) that needs to fit into the channel around the opening in the middle. Then, I placed the two printed buttons with the thicker part facing up into the slots, followed by the switch itself. Once that was completed, I positioned the **Base\_middle\_plate.stl** and guided the remaining 5 wires (3 from the buttons and 2 from the LEDs) through the opening from inside. Finally, I secured plate using four 4 x 3mm M2.5 screws.

#### ⚠ Check the wiring diagram



diy-spacemouse \_wiring diagram.pdf

[Download](#)

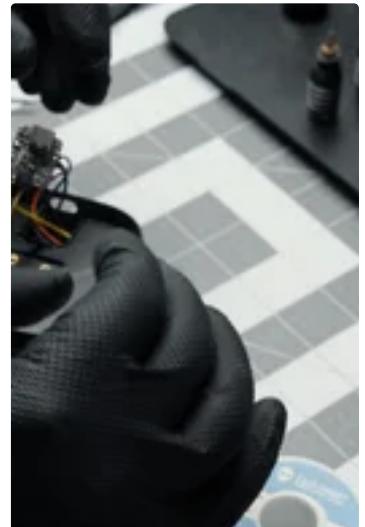
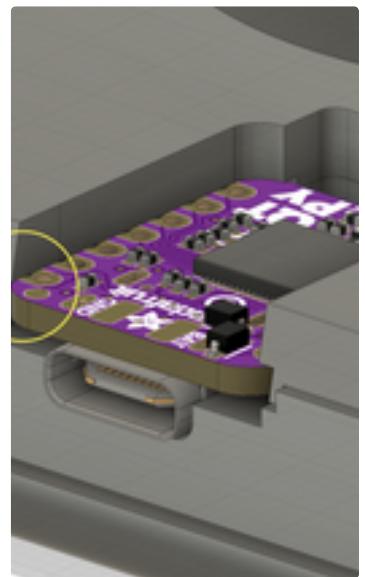
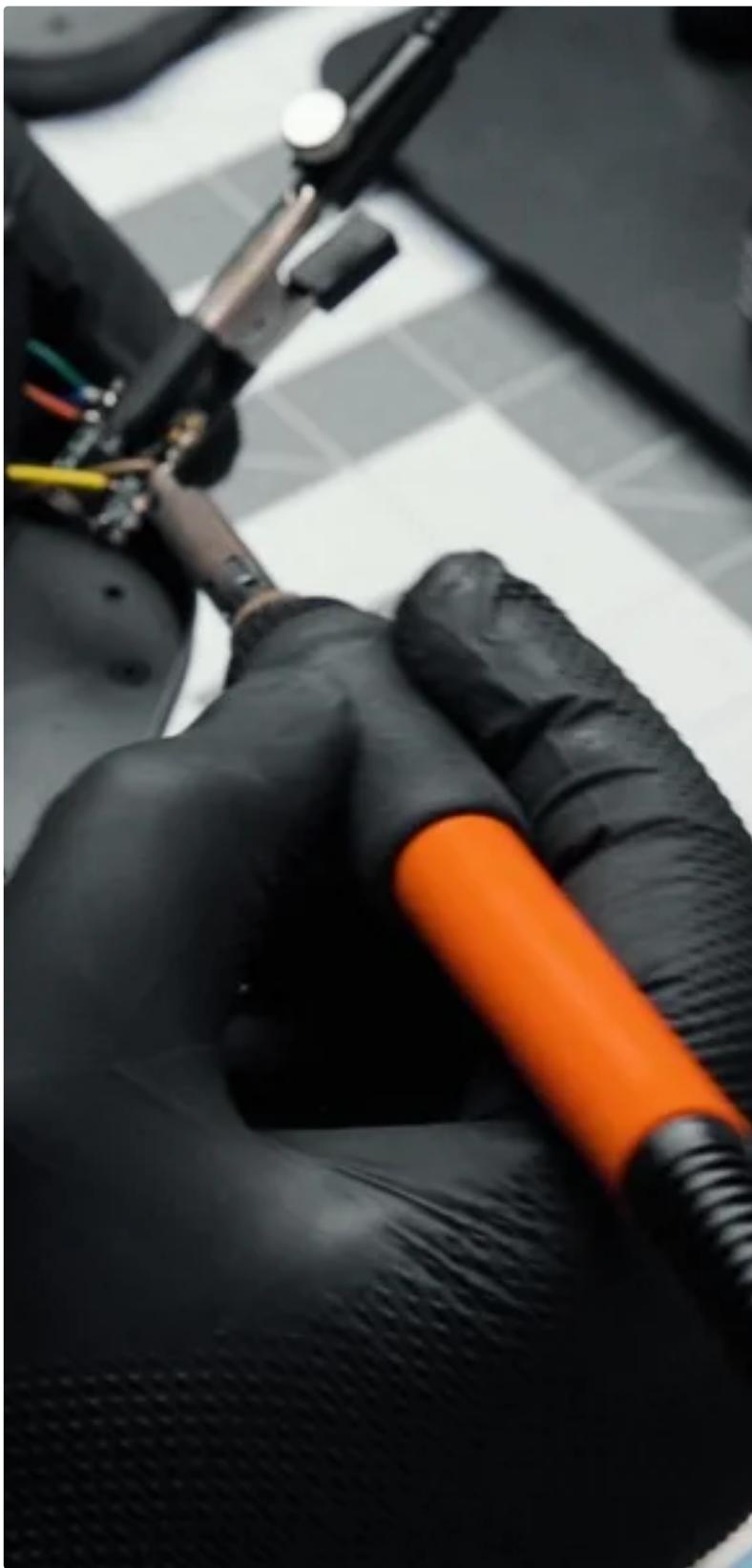
[Add Tip](#)

[Ask Question](#)

[Comment](#)

[Download](#)

## Step 9: Wiring \_3



Now, we can complete the wiring by soldering the remaining five wires to the RP2040 on the bottom side, ensuring they are placed as far away from the edges as possible (please refer to the image). After soldering the wires, we can neatly tuck the slack into the opening below, aligning the RP2040 snugly in its designated slot.

### **Check the wiring diagram**



diy-spacemouse \_wiring diagram.pdf

[Download](#)

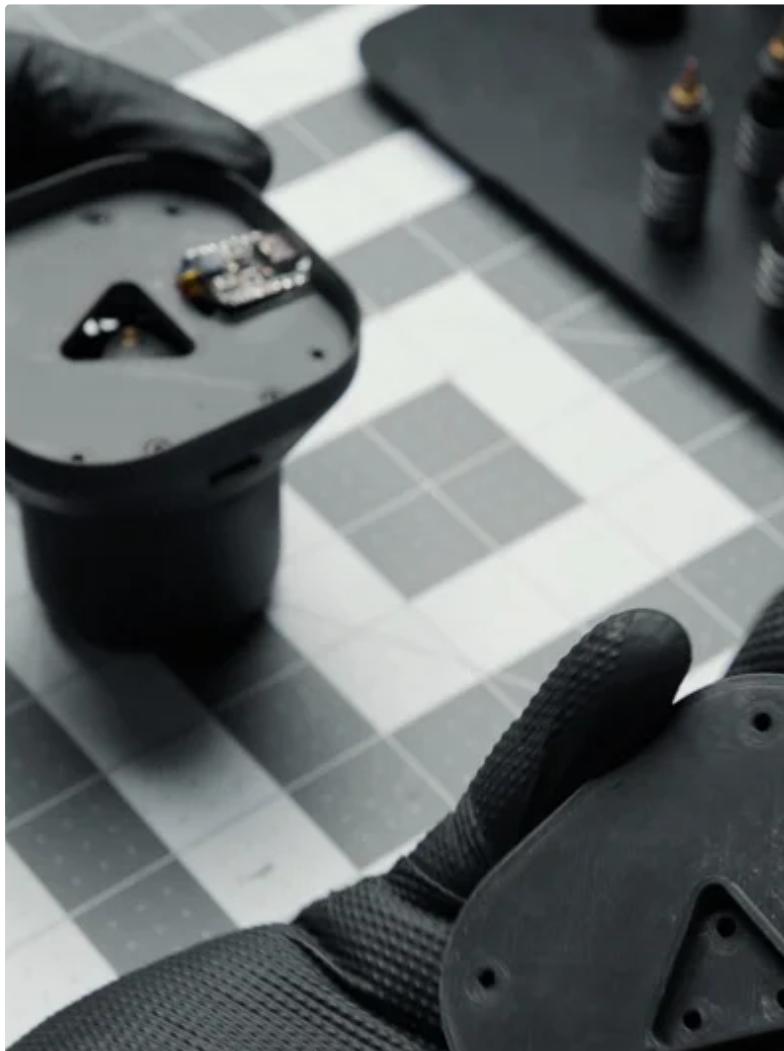
[Add Tip](#)

[Ask Question](#)

[Comment](#)

[Download](#)

## Step 10: Assembly \_3



The assembly was finalized by positioning the **Base\_bottom.stl** and securely attaching it to the stem using 3 x 8mm M2.5 screws then to the rest of the body using 4 x 12mm M2.5 screws.

To enhance stability, I affixed 4 rubber feet to the bottom.

Now, we can proceed to upload the code and conduct a test run of the device.

Add Tip

Ask Question

Comment

Download

## Step 11: Code \_1

A screenshot of the Arduino IDE's Serial Monitor window. The window title is "Serial Monitor x". The text area shows the following output:

```
40 Serial.begin(9600);
41 Wire.begin();
42
43 // init mag sensor
44 mag.begin(Wire);
45 mag.setAccessMode(mag,MASTERCONTROLLEDMODE);
46 mag.disableTemp();
```

Message [Enter to send message to 'Adafruit QT Py RP2040' on '/dev/cu.usbmodem14101']

0...06  
-0.12  
-7.72

The window is set against a dark OS X-style desktop background with various application icons at the bottom.

First thing is to get the Adafruit QT Py RP2040 working in the Arduino IDE. You can follow [this guide](#) if you never done this before.

Once the board is working properly you can open the Arduino sketch located [here](#) and install the required libraries in your environment.

```
#include <TinyUSB_Mouse_and_Keyboard.h>
#include <OneButton.h>
#include <Tlv493d.h>
#include <SimpleKalmanFilter.h>
```

In order for the sketch to compile you will also need to change the USB stack to from Arduino to [TinyUSB](#). You can read more on how to do that [here](#).

Once the board flashed successfully you can unplug, open the serial console and plug it back in.

By default the code prints out to the console x,y,z after initialization.

You can test the knob to see if it moves the mouse cursor and if it's in the right orientation. Incase it's reversed you can invert the values in the code if you don't want to open it and flip the sensor.

Depending on how centred the knob in your build and how the everything fits together, you might need to adjust the **magRange** and **sensitivity** values. You can move the knob to the extremities take a look at the x and y values printed on serial console and adjust the ceiling value for **magRange**.

The "dead zone" or center position is represented by **xyThreshold** you can adjust this for tighter tolerances to your preference.

```
int calSamples = 300;
int sensitivity = 8;
int magRange = 3;
int outRange = 127; // Max allowed in HID report
float xyThreshold = 0.4; // Center threshold
```

The shortcut buttons are sent from these 2 functions. You can define new combos if you like using this [reference](#).

```
// go to home view in Fusion 360 by pressing (CMD + SHIFT + H) shortcut assigned to the custom Add-in command
void goHome()
{
    Keyboard.press(KEY_LEFT_GUI);
    Keyboard.press(KEY_LEFT_SHIFT);
    Keyboard.write('h');

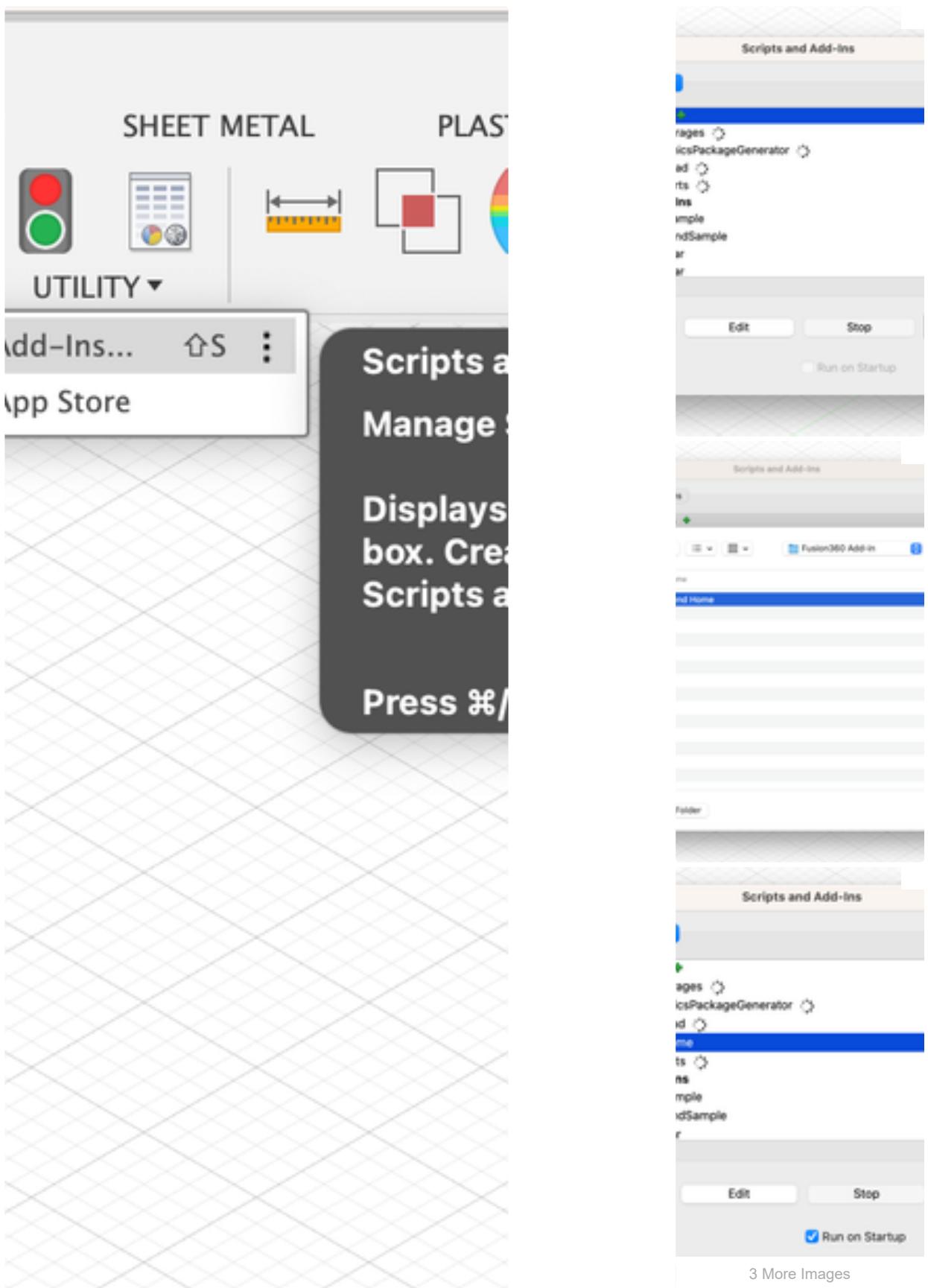
    delay(10);
    Keyboard.releaseAll();
    Serial.println("pressed home");
}

// fit to view by pressing the middle mouse button twice
void fitToScreen()
{
    Mouse.press(MOUSE_MIDDLE);
    Mouse.release(MOUSE_MIDDLE);
    Mouse.press(MOUSE_MIDDLE);
    Mouse.release(MOUSE_MIDDLE);

    Serial.println("pressed fit");
}
```

Add Tip    Ask Question    Comment    Download

## Step 12: Code \_2



To get **homeview** shortcut working you'll need to install the custom Add-in for Fusion.

In the toolbar, go to UTILITIES > Scripts and Add-Ins.

Click on the Add-Ins tab. Under My Add-Ins click on the green plus sign and copy the **Send home** folder located [here](#) to that location and click open.

Make sure **Run on Startup** is active and click **Run**.

You should see a new command > **Send To Home View** appear under Add-Ins.

Click on the 3 dots icon > Change Keyboard Shortcut...

Assign the same keyboard shortcut sent by the button press from the device: ***cmd+shift+h***

Add Tip

Ask Question

Comment

Download

## Step 13: Done



Thank you for sticking around until the end!

This was a lot of fun to make and I enjoyed the process. There is certainly a lot of improvements to be made. I'd love to hear your suggestions in the comments below.

I have a lot of ideas and I plan on making more projects like these [Follow me](#) for more